Politechnika Warszawska

WYDZIAŁ ELEKTRONIKI
I TECHNIK INFORMACYJNYCH

Instytut Automatyki i Informatyki Stosowanej

Praca dyplomowa magisterska

na kierunku Informatyka
w specjalności Systemy Informacyjno-Decyzyjne

Analiza wydźwięku zdania na podstawie drzewa rozbioru

Piotr Poniedziałek
Numer albumu 263345

promotor
dr inż. Mariusz Kamola

WARSZAWA 2019

Streszczenie

Analiza wydźwięku na podstawie drzewa rozbioru zdania

Przedstawiona praca magisterska opisuje rozważania dotyczące analizy wydźwięku dla języka polskiego. Sentyment badany jest dla opinii książek, napisanych przez użytkowników w sieci na popularnym serwisie internetowym *lubimyczytac.pl*. Głównym celem pracy jest wykorzystanie do badań informacji uzyskanych z transformacji zdania na drzewo rozbioru, opierając się na gramatyce języka polskiego. Zabieg ten jest zastosowany do ekstrakcji cech tekstu. Uzyskane wyniki porównane są z inną popularną metodą reprezentacji tekstu - *bag of words*. Określenie nacechowania emocjonalnego opinii realizowane jest poprzez zastosowanie jednej z metod nadzorowanego uczenia maszynowego - naiwnego klasyfikatora Bayesa. Pierwsza część pracy skupia się na przedstawieniu niezbędnej teorii dotyczącej analizy wydźwięku, drzew rozbioru oraz klasyfikacji tekstu. Każdy z tematów zobrazowany jest odpowiednimi przykładami. W dalszej części dokument przedstawia szczegóły dotyczące projektu, w ramach którego zaimplementowane zostały dwie aplikacje, będące podstawą przeprowadzonych badań. Ponadto, praca zawiera opis narzędzi służących do analizy i przetwarzania języka naturalnego, które znalazły zastosowanie w projekcie badawczym oraz tych, które okazały się nieudanym eksperymentem. Całość zakończona jest podsumowaniem, które ukazuje możliwości rozwoju projektu w przyszłości.

Słowa kluczowe: analiza wydźwięku, klasyfikacja tekstu, drzewo rozbioru, *bag of words*, Bayes

Abstract

Sentiment analysis using sentence's syntax tree

Presented master's thesis describes sentiment analysis of text in Polish language. Study focuses on opinions of books, written online by users on popular among polish readers community website - *lubimyczytac.pl*. Main goal of thesis is to make use of output data collected from transformation of sentences to syntax trees, based on Polish grammar. The purpose of this step is to enhance feature extraction process applied to analyzed text. Results obtained using this method are compared with more common and popular approach to expressing vectorized text representation - *bag of words*. Sentiment analysis is carried out based on one of the supervised machine learning methods - naive Bayes classifier. First part of document focuses on introduction of essential theory concerning sentiment analysis, syntax trees and text classification. Each of listed topics is illustrated with examples. Following part of thesis describes details about implementation of two applications, which are base for sentiment verification, as a part of main project. Moreover, document contains description and short summary not only of tools that were successfully used during analysis, but also for experimental and failed attempts. Thesis ends with summary containing ideas for future project development.
Keywords: sentiment analysis, text classification, syntax tree, *bag of words*, Bayes

załącznik do zarządzenia nr 28/2016 r.
Rektora PW

Politechnika Warszawska

„załącznik nr 3 do zarządzenia nr 24/2016 Rektora PW

.....
miejscowość i data

.....
imię i nazwisko studenta

.....
numer albumu

.....
kierunek studiów

OŚWIADCZENIE

Świadomy/-a odpowiedzialności karnej za składanie fałszywych zeznań oświadczam, że niniejsza praca dyplomowa została napisana przeze mnie samodzielnie, pod opieką kierującego pracą dyplomową.

Jednocześnie oświadczam, że:

- niniejsza praca dyplomowa nie narusza praw autorskich w rozumieniu ustawy z dnia 4 lutego 1994 roku o prawie autorskim i prawach pokrewnych (Dz.U. z 2006 r. Nr 90, poz. 631 z późn. zm.) oraz dóbr osobistych chronionych prawem cywilnym,
- niniejsza praca dyplomowa nie zawiera danych i informacji, które uzyskałem/-am w sposób niedozwolony,
- niniejsza praca dyplomowa nie była wcześniej podstawą żadnej innej urzędowej procedury związanej z nadawaniem dyplomów lub tytułów zawodowych,
- wszystkie informacje umieszczone w niniejszej pracy, uzyskane ze źródeł pisanych i elektronicznych, zostały udokumentowane w wykazie literatury odpowiednimi odnośnikami,
- znam regulacje prawne Politechniki Warszawskiej w sprawie zarządzania prawami autorskimi i prawami pokrewnymi, prawami własności przemysłowej oraz zasadami komercjalizacji.

Oświadczam, że treść pracy dyplomowej w wersji drukowanej, treść pracy dyplomowej zawartej na nośniku elektronicznym (płycie kompaktowej) oraz treść pracy dyplomowej w module APD systemu USOS są identyczne.

.....
czytelny podpis studenta"

Spis treści

1. Wstęp

1.1. Problematyka tematu

W dzisiejszych czasach wyrażenie opinii na zadany temat lub przedmiot dyskusji, która dotrze do wielu odbiorców na całym świecie, jest zadaniem trywialnym. Rozwój technologiczny, a w szczególności internetu, spowodował, że wyrażenie swojego zdania o ulubionej książce lub restauracji wymaga dostępu do sieci oraz kilku ruchów z wykorzystaniem myszki i klawiatury komputerowej. Powszechność internetu oraz łatwość z jaką można korzystać z jego możliwości, doprowadziły do masowej wymiany zdań i komunikatów między ludźmi pochodzącaych z dowolnego miejsca na ziemi. Mnogość utrwalonych w sieci opinii wyrażonych przez użytkowników sprawił, że internet stał się źródłem bogatym w cenne informacje dotyczące wybranych zagadnień, czy przedmiotów.

Taką wiedzę warto wykorzystać - mając świadomość jakie są opinie użytkowników na temat wybranego modelu telewizora, producent może się do nich ustosunkować, by ostatecznie produkt odpowiednio dostować do potrzeb odbiorców. Im większy zbiór danych, tym więcej informacji zwrotnej od klienta, a co za tym idzie - łatwiejsza praca nad doskonaleniem produktu. Działa to również w drugą stronę - zdanie jednego użytkownika może mieć wpływ na decyzję drugiego. W przypadku, gdy opinii jest wiele, ręczna weryfikacja tego, co autor miał na myśli, może być uciążliwa. Z tego powodu istnieje potrzeba wprowadzenia automatyzacji do procesu analizy tekstu. Najważniejsze pytanie - w jaki sposób rozpoznać, czy opinia autora jest pozytywna bądź negatywna? Jak odczytać zawarte w tekście emocje?

Odpowiedzi na te pytania stara się ustalić analiza wydźwięku. Jej zadaniem jest określenie w sposób automatyczny (jeśli to możliwe) nacechowania emocjonalnego danego tekstu. Metod oraz rodzajów analizy jest wiele, kilka z nich zostało omówionych w dalszej części pracy.

1.2. Cel i zakres pracy

Celem pracy jest weryfikacja, czy wykorzystanie metod rozkładu zdań na różnego rodzaju struktury drzewiaste poprawia jakość analizy wydźwięku. W zakres pracy wchodzi również porównanie tychże metod to klasycznych podejść wykorzystywanych powszechnie w badaniach wydźwięku. Dodatkowo, zaprezentowane zostanie zestawienie popularnych narzędzi służących do analizy tekstu. Praca skupia się na badaniu sentymentu dla języka polskiego, w szczególności będą to opinie internautów dotyczące książek, pochodzące z popularnego w sieci portalu internetowego *lubimyczytac.pl* [1].

[1] http://lubimyczytac.pl

Rodział pierwszy jest krótkim wstępem do pracy, zawiera wprowadzenie do tematyki analizy wydźwięku. Przedstawia cel oraz zakres omawianej pracy.

Rodział drugi zagłębia się w analizę sentymentu, dokładnie opisuje metody i problemy związane z badaniem wydźwięku. Ponadto przedstawia zastosowania oraz możliwości wykorzystania analizy w celach komercyjnych.

Rodział trzeci wprowadza pojęcie drzewa rozbioru. Ukazuje teoretyczne rozważania dotyczące tej tematyki oraz definiuje powiązaną z nią terminologię. Sekcja zawiera także opis dwóch najważniejszych rodzajów drzew rozbioru, które stanowią podstawę badań opisanych w pracy.

Rodział czwarty to część pracy, której celem jest przedstawienie istoty zadania klasyfikacji tekstu. W zwięzły sposób opisuje poszczególne etapy procesu, podkreślając możliwe problemy i sposoby ich rozwiązania. Ponadto ukazuje różne metody reprezentacji tekstu, które umożliwiają poprawne działanie klasyfikatora. Opisano także wybrany rodzaj klasyfikacji wykorzystującej nadzorowane uczenie maszynowe - naiwny klasyfikator Bayesa, który jest podstawą analizy w pracy.

W rodziale piątym przedstawiono dokładny opis prac nad projektem magisterskim. Zawarto w nim informacje dotyczące wybranego zbioru danych oraz wykorzystanych narzędzi (łącznie z nieudanymi eksperymentami). Kolejne sekcje szczegółowo opisują sposoby analizy oraz warianty przeprowadzonych badań, których wyniki ukazane zostały w formie wykresów oraz tabel. Dodatkowo, rozdział zawiera krótkie wprowadzenie dotyczące szczegółów implementacyjnych aplikacji stworzonych na potrzeby projektu.

Rodział szósty jest zakończeniem dokumentu - stanowi podsumowanie wykonanych w ramach projektu prac, podkreśla to, co zostało wykonane poprawnie oraz to, co mogło być wykonane lepiej. Zakończenie przedstawia wnioski wyciągnięte z przeprowadzonych badań oraz ukazuje możliwości rozwoju projektu w przyszłości.

2. Analiza wydźwięku

Analiza wydźwięku, znana też jako analiza sentymentu, jest szczególnym rodzajem analizy tekstu, która skupia się na wydobyciu z treści emocji autora oraz poznaniu jego subiektywnej oceny [9]. Łączy w sobie elementy pochodzące z lingwistyki komputerowej oraz przetwarzania języka naturalnego. Pojęcie analizy wydźwięku często stosowane jest zamiennie z eksploracją opinii, której zadaniem jest wykrycie zawartej w tekście opinii wraz z jej podsumowaniem oraz wskazanie elementów oceny [15]. Efektem analizy mogą być m. in.:

— klasyfikacja tekstu do jednej z dwóch kategorii: pozytywna lub negatywna - jest to najprostszy przypadek [10]
— klasyfikacja tekstu do jednej z wielu kategorii: stopniowanie polaryzacji
— zastosowanie ciągłej skali oceny tzn. brak klasyfikacji to jednej z klas, tylko wyznaczenie oceny z określonego przedziału, np. $4, 1/10$
— skupienie się na ocenie poszczególnych cech danego produktu
— wykrycie (ekstrakcja) cech produktu - przydatne w przypadku nowej, nieznanej do tej pory domeny
— weryfikacja, czy tekst jest obiektywny, czy może zawiera subiektywną opinię autora
— wykrywanie nazw własnych (*Named Entity Recognition*)
— wykrycie oraz nazwanie poszczególnych emocji (m. in. radość, miłość, nadzieja) występujących w tekście
— analiza porównawcza kilku produktów.

Wymienione zostały przykładowe rezultaty/cele analizy wydźwięku. Najpopularniejszym jest pierwszy punkt na liście - binarna klasyfikacja tekstu do jednej z klas [5]. Taki rodzaj został też wykorzystany do analizy w projekcie magisterskim, opisanym szerzej w dalszej części pracy.

2.1. Metody analizy

Analizę można przeprowadzić na kilka różnych sposobów [19]. Poniżej znajduje się krótki opis najpopularniejszych podejść.

2.1.1. Analiza ręczna

Jest to najprostsza metoda, najskuteczniejsza, ale zarazem najbardziej prymitywna. Wymaga bezpośredniego zaangażowania człowieka w proces analizy, opiera się na ludzkich zdolnościach analitycznego myślenia oraz rozpoznawaniu emocji drugiej osoby. Ze względu na rozmiary dostępnych zbiorów danych oraz możliwości współczesnych komputerów, wprowadza się automa-

tyzację procesu. Z tego powodu ten rodzaj analizy jest zastępowany przez nowoczesne, bardziej zaawansowane metody.

2.1.2. Słowniki wydźwięku

Metoda ta wykorzystuje słownik tekstowy, zawierający listę słów, które mają przypisaną kategorię [17]. Każde słowo może mieć wydźwięk pozytywny lub negatywny. Analizę słownikową można przeprowadzić na kilka sposobów:

— Weryfikacja, czy tekst zawiera wybrane słowo klucz ze słownika. Na podstawie nacechowania wyrazu, określany jest wydźwięk całej opinii. Dla przykładu w zdaniu:

*Ogólnie bardzo **dobra** książka.*

Jeśli słowem klucz jest wyraz *dobra* o wydźwięku pozytywnym (według słownika), to powyższy przykład zaklasyfikowany zostanie jako zdanie o nacechowaniu również pozytywnym.

— Zliczanie występowania poszczególnych słów. Kategoria, która posiada większą liczebność w danym tekście, jest mu przypisywana. Dla przykładu:

*Książka **dobra**, ale momentami **nudna** i **nieciekawa**.*

Posiadając słownik, w którym *dobra* jest słowem pozytywnym, natomiast *nudna* oraz *nieciekawa* są negatywne, całe zdanie staje się negatywne, ze względu na przeważającą liczbę występujących wyrazów o takim nacechowaniu.

— Wykorzystanie rachunku prawdopodobieństwa. Zamiast jednoznacznie nadawać poszczególnym wyrazom daną kategorię, można określić ją z pewnym prawdopodobieństwem.

Aby móc analizować wydźwięk za pomocą opisywanej metody, wymagane jest posiadanie przygotowanego słownika. Istnieją dwie wiodące ścieżki konstruowania słowników:

— Manualne przygotowanie otagowanej listy słów. Często do tego celu wykorzystuje się społeczność internautów, którzy poprzez udostępniane ankiety, mogą pomóc określić polaryzację podanych słów, co przyspiesza proces konstrukcji słownika. Taki słownik można też z dużą pewnością określić jako wiarygodne źródło informacji.

— Podejście zautomatyzowane - algorytmy, które bazując na niedużym zbiorze początkowym wyrazów, są w stanie słownik rozszerzać i dodawać do niego kolejne elementy z odpowiednią polaryzacją. Przykładami takich algorytmów są zaproponowane przez Hatzivassiloglou i McKeown [6] oraz Turneya [18]. Pierwszy z nich opierał się na obserwacji, że wyrazy połączone spójnikiem *i* mają taki sam wydźwięk, natomiast dla spójnika *ale* - wydźwięk przeciwny. W ten sposób korpus tekstowy zostaje podzielony na dwie niezależne grupy, dla których określana jest polaryzacja. Turney natomiast wykorzystał tzw. współczynnik

orientacji semantycznej. Nacechowanie danego wyrazu wyznaczane jest na podstawie prawdopodobieństwa współwystępowania ze słowem *excellent* oraz *poor*, czyli słów o dwóch skrajnych polaryzacjach.

Oczywiście, chcąc analizować sentyment wykorzystując jedną z opisanych metod, nie ma konieczności tworzenia własnego słownika. Istnieje wiele gotowych pozycji, które można wykorzystać i dostosować do własnych potrzeb. Zdecydowana większość odnosi się do języka angielskiego, aczkolwiek można też znaleźć słowniki dla języka polskiego. Przykładem jest *Słowosieć*, której możliwości zostały szerzej opisane w rodziale 5.

2.1.3. Reguły składniowe i gramatyczne

Metoda wykorzystuje zdefiniowany zestaw reguł i wzorców, na których powinna opierać się klasyfikacja. Podejście to jest bardziej zaawansowane od poprzedniego, nie skupia się tylko i wyłącznie na poszczególnych wyrazach, bierze pod uwagę tekst jako całość (oczywiście nie musi - zależy jak zdefiniowana jest reguła).

Jako przykład mogą posłużyć dwie reguły zdefiniowane w publikacji [22]. Pierwsza z nich skupia się na wykryciu określeń, które odwracają polaryzację nacechowanych emocjonalnie wyrażeń np. z pozytywnego na negatywny (przytoczony dokument wyraża sentyment jako wartość liczbową, ”odwrócenie” wydźwięku realizowane jest poprzez przemnożenie przez -1). Druga przykładowa reguła wyszukuje określone grupy wyrazów np. (przymiotnik, rzeczownik) i na podstawie słów składowych wyliczany jest wydźwięk całej grupy.

W tym przypadku wymagana jest dobra wiedza domenowa w połączeniu z lingwistyką, aby powstałe reguły spełniały swoją rolę. Ciężko jest dobrać wzorce, które byłyby uniwersalne. Z tego powodu metoda ta nie jest elastyczna - w innym kontekście zaproponowany wzór czy reguła może okazać się zupełnie nieprzydatna. Jest to ogólny problem, który nie dotyczy tylko i wyłącznie tej metody, ale w tym przypadku jest on najbardziej wyeksponowany.

2.1.4. Uczenie maszynowe

Metody oparte o uczenie maszynowe zyskują popularność i są coraz częściej wykorzystywane do analizy wydźwięku [12] [3] [16]. Wynika to przede wszystkim z ich uniwersalności i poziomu automatyzacji - z założenia to system przetwarzający dane i je analizujący, powinien się ”nauczyć” obowiązujących w tekście reguł, określić występujące wzorce i być w stanie wykorzystać te informacje w innym kontekście. To tylko teoretyczne założenia, które w praktyce nie są takie proste i oczywiste w implementacji.

Uczenie maszynowe opiera się na konstrukcji klasyfikatora, którego zadaniem jest przypisanie odpowiedniej etykiety (pochodzącej ze skończonego zbioru elementów) dla każdego z elementów pochodzącego ze zbioru wejściowego (jest to przypadek zadania klasyfikacji, w przypadku zadania regresji, wyjściem jest wartość pochodząca z przestrzeni ciągłej) [8]. Inaczej mówiąc - zadaniem uczenia maszynowego jest znalezienie przybliżenia takiej funkcji g:

$$g(x) = y$$

gdzie x jest zbiorem danych wejściowych, natomiast y wyjściem z modelu, np. zbiorem etykiet. Wtedy, dla nowych, niewidzianych przez klasyfikator danych, możliwe jest wyznaczenie klasy bądź liczby z danego przedziału.

Ze względu na sposób identyfikacji funkcji g, uczenie maszynowe można podzielić na dwa główne rodzaje:

— uczenie nadzorowane (*ang. supervised learning*) - klasyfikator uczony jest na specjalnie przygotowanym trenigowym zbiorze danych. Zbiór ten zawiera dane wejściowe, które opatrzone są ”wzorcową” etykietą. Na tej podstawie algorytm uczący dobiera odpowiednie parametry modelu (funkcji g)

— uczenie nienadzorowane (*ang. unsupervised learning*) - klasyfikator nie jest uczony na podstawie dostarczonego zbioru trenigowego, tylko sam jest odpowiedzialny za wykrycie wzorców oraz reguł w zbiorze danych.

Przykładami klasyfikacji z uczeniem nadzorowanym są: *SVM* (*Support-vector machine*) i naiwny klasyfikator bayesowski. Dla uczenia nienadzorowanego jako przykład może posłużyć algorytm centroidów (*k-means clustering*).

2.1.5. Podejście hybrydowe

Istnienie kilku metod analizy nie oznacza, że są zupełnie niezależne i muszą być wykorzystywane osobno. Częstym zabiegem jest łączenie ze sobą różnych metod np. słowników wydźwięku oraz uczenia maszynowego. Celem jest wyciągniecie z tych dwóch podejść ich najlepszych cech - skuteczności uczenia maszynowego oraz szybkości metody słownikowej. Jako przykład może posłużyć publikacja [11].

Autorzy opisują podejście, w którym wykorzystali nadzorowaną metodę uczenia maszynowego, nie posiadając wstępnie zbioru trenigowego, składającego się z oznaczonych etykietami dokumentów. Zbiór ten został utworzony automatycznie z pomocą osób, których zadaniem było wskazanie ”reprezentatywnego” zestawu słów kluczowych powiązanych z daną klasą. Na podstawie tak utworzonego zbioru wyrazów oraz algorytmów grupowania/klastrowania, każdy dokument ze zbioru trenigowego otrzymał odpowiednią kategorię, umożliwiając nadzorowaną metodę uczenia maszynowego.

2.2. Problemy analizy wydźwięku

Analiza wydźwięku nie jest zadaniem prostym. To skomplikowany proces, który wymaga odpowiedniego przygotowania oraz rozpatrzenia wielu potencjalnych problemów. Wyzwania związane z analizą sentymentu (a nawet w ogólności - z przetwarzeniem języka naturalnego) [15] [10]:

— Różnorodność języków. Ciężko jest zdefiniować uniwersalne zasady i reguły, na których oparta zostanie analiza dla języka polskiego i angielskiego. Wspomniane wcześniej słowniki w języku niemieckim, będą się różnić od tych skonstruowanych dla języka japońskiego.

Bogactwo językowe powoduje konieczność rozpatrzenia znacznie większej liczby przypadków.

— Ewolucja języka. Słownictwo się zmienia, pojawiają się nowe trendy językowe. Styl pisania i wyrażania opinii każdego człowieka jest inny, co należy wziąć pod uwagę w trakcie analizy. W ramach danego języka obecne są także różnego rodzaju slangi, które mogą sprawiać problemy w prawidłowej interpretacji (wpływ na to ma wiek, wykształcenie, czy też pochodzenie autora). Zjawisko to jest szczególnie widocznie obecnie, w świecie internetu, w miejscach przeznaczonych do przekazywania komunikatów: fora, grupy dyskusyjne czy portale społecznościowe.

— Błędy językowe. Ludzie popełniają błędy - jest to zjawisko normalne, na które należy być przygotowanym. Ortograficzne, generalnie związane z gramatyką danego języka, ale też nierzadko pojawiają się literówki i błędy interpunkcyjne, co często spowodowane jest pośpiechem i brakiem ostatecznej korekty tekstu. W przypadku takich błędów, interpretacja tekstu może stać się kłopotliwa.

— Negacja. Jak powinno się interpretować różnego rodzaju formy negowania w tekście? Przykładowo:

*Nie jest to **zły** i **bezużyteczny** telewizor, mimo tego, co twierdzi wiele osób.*

W zdaniu znajdują się dwa przymiotniki o wyraźnie negatywnym wydźwięku *zły* i *bezużyteczny*, co może doprowadzić do mylnego wniosku, że całe stwierdzenie jest nacechowane negatywnie. Zastosowanie słowa *nie* w pewien sposób odwraca polaryzację epitetów i sprawia, że całe zdanie ma wydźwięk pozytywny.

W związku z tym, aby analiza miała sens, należy brać pod uwagę wyrazy negujące. Jedną z możliwości jest odwrócenie ”słownikowej” polaryzacji wyrazu, jeśli wcześniej w wyrażeniu pojawiło się słowo negujące. W tym miejscu pojawia sie kolejny problem - ”ile” wcześniej powinna wystąpić negacja? Bezpośrednio przed wyrazem ze słownika? Jak określić odpowiedni zasięg negacji? Jeśli w zdaniu wylistowanych zostało kilka przymiotników jeden za drugim - czy negacja odnosi się do każdego z tych wyrazów (w tym przypadku duże znaczenie ma w jaki sposób słowa są ze sobą połączone tj. rodzaj spójnika lub interpunkcji)? Powyższy przykład zdania dobrze obrazuje wymienione pytania i wątpliwości. Możliwości rozwiązania problemu negacji zostały opisane w [1] [2].

— Kontekst analizy. Duże znaczenie w przypadku analizy wydźwięku ma to, w jakim kontekście jest wykonywana, w szczególności, gdy wykorzystywane są słowniki. Słowa, które posiadają wydźwięk pozytywny w jednej domenie, mogą mieć przeciwny w drugiej. Przykładowo wyraz *przewidywalny* - w przypadku recenzji książki wyraża negatywną opinię, natomiast jako opis zachowania spadochronu - zdecydowanie pozytywną. W związku z tym, częstym zabiegem jest konstrukcja słowników wydźwięku dla każdej analizowanej domeny

osobno.

— ”Zanieczyszczenie” tekstu. Oznacza to nie tylko opisane wyżej błędy w tekście, ale także występowanie znaków, które nie wnoszą wartości emocjonalnej do tekstu. Pod tę kategorię można również zaliczyć wszelkie wyrażenia, o takim samym znaczeniu, ale przedstawione w różnych formach, np. występowanie wielkich liter, czy niejednolite w zakresie dokumentu formaty zapisu daty.

— Sarkazm. Wykrycie ironii w tekście za pomocą zautomatyzowanych metod jest zadaniem niezwykle trudnym. Identyfikacja prawdziwych intencji autora, który używa słów naprowadzających na zgoła inny kierunek, jest często niemożliwe, w szczególności wykorzystując najprostsze metody słownikowe. Przykładowo:

Naprawdę genialna książka! Stanowi idealną podstawę pod mój monitor!

Powyższe zdania zawiera wyraz *genialna* oraz *idealną*, ale nie można powiedzieć, że całość stanowi pozytywną opinię książki (przyjęto założenie, że książka oceniana jest przez pryzmat walorów wynikających z czytania, a nie jako podstawa pod inne przedmioty). Wręcz przeciwnie - bez wyczucia sarkazmu oraz pełnego kontekstu wypowiedzi, nie można prawidłowo określić wydźwięku danego stwierdzenia.

— Stopień granulacji analizy. Jeśli analizowany jest dokument tekstowy, składający się z rodziałów, dalej ze zdań, zdań składowych i ostatecznie z pojedynczych wyrazów, to na którym z tych poziomów powinna koncentrować się analiza? Podejście słownikowe zaczyna od słów, na podstawie których określany jest wydźwięk całego dokumentu. W tym miejscu należy również wspomnieć o *tokenizacji*, czyli rozbiciu tesktu na możliwie najmniejsze cześci składowe, co nie jest zadaniem trywialnym.

2.3. Zastosowania analizy wydźwięku

Jak wspomniano na wstępie pracy, w dobie dzisiejszych rozwiązań technologicznych i rozwoju internetu, analiza wydźwięku staje się narzędziem coraz popularniejszym, którego walory można wykorzystać na wiele różnych sposobów. Przykładowe zastosowania badania sentymentu tekstu pisanego są następujące [14]:

— Agregacja opinii na temat usług i produktów. Biznes otrzymuje automatyczną informację zwrotną od klienta, bez konieczności przeprowadzania dodatkowych badań lub ankiet, które mogą okazać się kosztowne.
— Systemy rekomendacji. Dostosowanie proponowanych produktów użytkownikowi, w zależności od jego preferencji i wystawionych opinii.
— Wykrywanie niechcianych wiadomości przesyłanych za pomocą poczty elektronicznej. Przykładowo, jeśli system analizujący zidentyfikuje e-maila, który posiada negatywny wydźwięk, może taką wiadomość zablokowować.

— Określanie strategii dotyczącej wyświetlania reklam internetowych. Podobny przypadek do systemów rekomendacji - dostosowanie wyświetlanych treści w zależności od wydźwięku wyrażonych w sieci opinii.
— Badanie nastrojów panujących w społeczeństwie. W szczególności przydatne w polityce podczas planowania kampanii wyborczych.

3. Drzewo rozbioru

Każde zdanie lub ogólnie wyrażenie tekstowe, posiada pewną zdefiniowaną strukturę, charakterystyczną dla wybranego języka. Struktura ta ściśle powiązana jest z gramatyką obowiązującą w ramach języka. Wymusza to stosowanie się do określonych reguł i zasad dotyczących składni podczas konstrukcji zdań [7].

Oznacza to, że bazując na danej gramatyce, można analizowany tekst przedstawić w postaci *drzewa składniowego* (*ang. syntax tree*) lub używając ogólnego pojęcia - *drzewa rozbioru* (zdanie ”rozbierane” jest na mniejsze elementy składowe, stąd nazwa). W szczególności, konstrukcja *drzewa składniowego* jest przydatnym zabiegem w przypadku automatycznej analizy i przetwarzania języka naturalnego. Taka reprezentacja ułatwia identyfikację elementów zdania oraz określenie roli i ich wzajemnych relacji w tekście.

3.1. Rodzaje drzew rozbioru

Istnieją dwa główne, najcześciej stosowane rodzaje drzewa rozbioru: *składnikowe (ang. constituency tree)* oraz *zależnościowe (ang. dependency tree)*. Oba skupiają się na przedstawieniu zdania w postaci drzewa, ale ich wewnętrzna reprezentacja identyfikuje różne cechy analizowanego tekstu i jego elementów składowych.

3.1.1. Drzewo składnikowe

Najczęściej konstruowanym typem *drzewa rozbioru* jest *drzewo składnikowe*. Jego głównym celem jest ukazanie wewnętrznej struktury tekstu, rozbijając większe wyrażenia na mniejsze części składowe. Każdy poziom w drzewie przedstawia zbiór węzłów (dzieci), które razem tworzą węzeł nadrzędny (rodzic). Wewnętrzne węzły drzewa należą do zbioru elementów *nieterminalnych*, natomiast liście są przedstawicielami elementów *terminalnych*. Zbiór *terminalny* składa się z konkretnych symboli i wyrazów dla danego języka np. *ja*, *zrobiłem*, *piękna*. Z drugiej strony, zbiór *nieterminalny* zdefiniowany jest przez listę kategorii fraz wyrazowych, których przykładowe wartości przedstawione zostały poniżej:

— *NP (noun phrase)* - fraza nominalna, określana również jako rzeczownikowa
— *VP (verb phrase)* - fraza werbalna, określana również jako czasownikowa
— *PP (prepositional phrase)* - fraza przyimkowa
— *P (preposition)* - przyimek
— *Det (determiner)* - rodzajnik
— *N (noun)* - rzeczownik
— *V (verb)* - czasownik

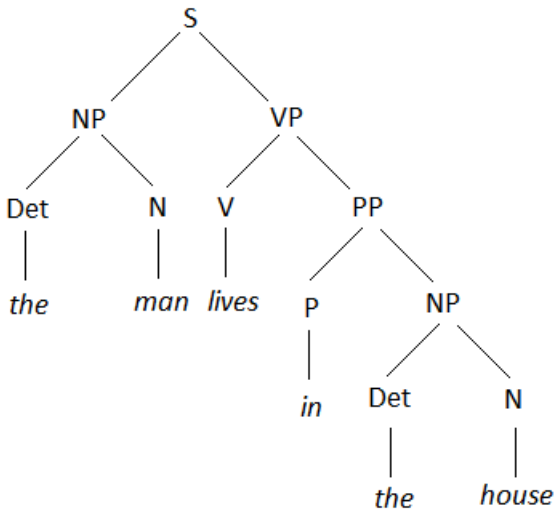
Węzeł *terminalny*, jak sama nazwa wskazuje, nie może zostać rozłożony na części składowe. Dla *nieterminalnych* elementów sytuacja wygląda inaczej - mogą one składać się z innych węzłów, należących do zbioru *terminalnego* jak i *nieterminalnego*. Dokładne zasady rozbioru poszczególnych fraz wyrazowych, określone zostały przez zadaną gramatykę. Przykładowo:

S -> NP VP
NP -> Det N
VP -> V PP
PP -> P NP
P -> 'in'
Det -> 'the'
N -> 'man' | 'house'
V -> 'lives'

jest listą reguł dla pewnej gramatyki, według których frazy mogą być rozkładane na części składowe (*S* oznacza tutaj symbol startowy). Dla zdania (przykład w języku angielskim):

The man lives in the house

Drzewo, skonstruowane na podstawie zdefiniowanej gramatyki, zaprezentowane jest na rysunku 3.1.



Rysunek 3.1. Przykładowe składnikowe drzewo rozbioru

Rysunek 3.1 dobrze ilustruje ideę rozkładania zdania na coraz mniejsze części. Zaczynając od tekstu początkowego (S), przechodząc przez kolejne poziomy drzewa, otrzymuje się poszczególne wyrazy, jako najmniejsze elementy składowe zdania.

3.1.2. Drzewo zależnościowe

Drugim, często wykorzystywanym w lingwistyce komputerowej, rodzajem drzewa rozbioru jest drzewo *zależnościowe*. Tego rodzaju struktura, w odróżnieniu od opisywanego wyżej podejścia, nie wykorzystuje pojęć węzłów terminalnych i nieterminalnych. W drzewie *zależnościowym* każdy z węzłów utożsamiany jest z pojedynczym wyrazem występującym w zdaniu. Ważną rolę odgrywają krawędzie łączące elementy w drzewie - określają one rodzaj relacji (zależności) między słowami (dla drzewa *składnikowego* krawędzie nie wnosiły żadnej wartości, poza ukazaniem związku dziecko - rodzic).

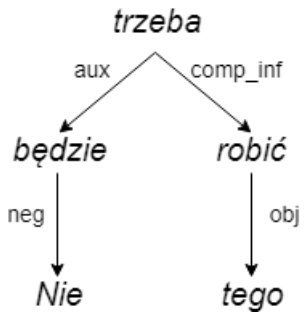
Przykładowo, zależności między wyrazami mogą być przedstawione jako [1]:

— *neg* - znacznik negacji
— *nsubj* - określenie podmiotu dla danej frazy
— *obj* - zależność określająca dopełnienie bliższe
— *aux* - oznaczenie dla czasownika ”pomocniczego” dla głównego w zdaniu

Jako demonstrację rozkładu zdania na drzewo zależnościowe, posłuży tekst:

Nie będzie trzeba tego robić.

Drzewo zależnościowe dla powyższego zdania zaprezentowane zostało na rysunku 3.2.



Rysunek 3.2. Przykładowe zależnościowe drzewo rozbioru

[1] Pełna lista zależności dla języka polskiego jest dostępna pod adresem http://zil.ipipan.waw.pl/PDB/DepRelTypes

3.2. Drzewa rozbioru a analiza wydźwięku

Drzewa rozbioru stanowią ważne ogniwo w procesie automatycznej analizy tekstu. Pozwalają na identyfikację występujących w zdaniu relacji pomiędzy wyrazami oraz na wyodrębienie poszczególnych części składowych. Cechy te umożliwiają badanie tekstu na poziomie jego struktury, czy reguł i zasad obowiązujących dla danego jezyka.

Takie dodatkowe informacje warto wykorzystać do analizy wydźwięku tekstu. Wymienione w poprzednim rodziale metody badania sentymentu, z powodzeniem można połączyć z technikami rozkładu zdań na drzewa rozbioru. Przykładowo, rozkład na drzewo rozbioru mógłby wspomóc proces generowania reguł językowych, na podstawie których tekst otrzymywałby odpowiednią kategorię wydźwięku.

Drzewo rozbioru jest także ciekawym źródłem różnorodnych cech składniowych tekstu, które mogą posłużyć jako zbiór danych treningowych dla klasyfikatora podczas analizy sentymentu metodą uczenia maszynowego. Praca skupia się na takim połączeniu, a efekty, ukazane zostały w dalszej części dokumentu.

4. Klasyfikacja tekstu

W poprzednim rodziale przedstawione zostały wstępne informacje dotyczące metody analizy wydźwięku, za pomocą uczenia maszynowego. Ta część pracy ukazuje temat z szerszej perspektywy i opisuje szczegóły kolejnych etapów procesu. Sekcja skupia się na nadzorowanej klasyfikacji, gdzie tekst przydzielany jest do jednej z definiowanej kategorii. Jako przykład algorytmu, zaprezentowany zostanie naiwny klasyfikator Bayesa.

4.1. Proces klasyfikacji

Proces klasyfikacji składa się z następujących podstawowych kroków:

— Zebranie danych. Bez zgrupowanych danych analiza nie byłaby możliwa. Najczęstszym źródłem jest internet, wiele badań dotyczyło wypowiedzi użytkowników portalu społecznościowego *Twitter* [5]. W przypadku uczenia nadzorowanego, dużym ułatwieniem są gotowe etykiety dla poszczególnych dokumentów, np. liczba gwiazdek, czy wystawiona ocena (wartości te najczęściej kategoryzuje się poprzez wyznaczenie progów liczbowych - jeśli ocena ma wartość powyżej 5 w skali 1 - 10, uważa się ją jako pozytywną, w przeciwnym wypadku - negatywną). Do pobierania danych z sieci zazwyczaj wykorzystuje się różne metody *webscrapingu* - zautomatyzowanego procesu ekstrakcji informacji pochodzących ze stron internetowych.

— Przetworzenie danych. Zebrane dane zazwyczaj nie nadają się do bezpośredniego użycia - należy je wstępnie przygotować. Do tego punktu zalicza się usuwanie z tekstu elementów, które mogą zaburzyć klasyfikację (niepotrzebne znaki, interpunkcja, emotikony).
W tym miejscu należy również wykonać *tokenizację* tekstu, czyli podzielić dokument na wymagane części składowe. *Token* może być, w zależności od wymagań analizy, reprezentowany różnie - najczęściej są to poszczególne wyrazy (znaki interpunkcyjne mogą być także traktowane jako *tokeny*). Przykładowo zdanie:

On powiedział, że lubi matematykę.

Po *tokenizacji* rozbite zostaje na:

['On', 'powiedział', ',', 'że', 'lubi', 'matematykę', '.']

Jeśli klasyfikator powinien wyrazy *Dobry* oraz *dobry* traktować jako jedno słowo, o tym samym znaczeniu, należy ujednolicić tekst, sprowadzając wszystkie elementy do małych liter.

W temacie ujednolicania składowych dokumentu, istotnym zabiegem jest lematyzacja oraz stemming. Lematyzacja polega na wyznaczeniu dla każdego wyrazu opisującej go jednostki słownika morfologicznego - leksemu. Innymi słowy, sprowadzanie słowa tekstowego do jego formy podstawowej. Przykładowe odmiany wyrazów, wraz z ich leksemami, zostały przedstawione w tabeli 4.1.

Słowo	**Możliwe leksemy**
samochodu	*samochód*
widziała	*widzieć*
tego	*to*
mam	*mieć, mama, mamić*

Tabela 4.1. Przykładowa lista słów wraz ich możliwymi leksemami/lematami

Z drugiej strony, *stemming* odnosi się do wydobycia z wyrazu tzw. rdzenia, czyli części, która nie podlega odmianom. Innymi słowy, od wyrażenia ucina się przedrostki/przyrostki, pozostawiając ciąg liter wspólny dla wyrazów tego samego pochodzenia. Przykładowo, dla słów:

robiła, porobili, robialny, urobić

Wspólnym stemem byłby ciąg znaków *robi*.

Zastosowanie lematyzacji oraz stemmingu sprawia, że analiza staje się precyzyjniejsza - semantyka danego wyrażenia zaczyna mieć znaczenie, a nie tylko forma w jakiej zostało słowo przedstawione. Pozornie różnie słowa, po przetworzeniu, są traktowane jednakowo. Naturalnym wydaje się fakt, że wyrazy *przeczytałem* i *przeczytali* niosą za sobą takie samo znaczenie. W przypadku rozdzielenia każdej takiej pary przez klasyfikator, wymagany jest bardzo duży zbiór danych, by słowa miały stosowną liczbę reprezentantów w słowniku.

Etap przetworzenia danych może zawierać usunięcie tzw. *stop words*. Jest to gotowa lista słów dla danego języka, które nie wnoszą zabarwienia emocjonalnego do tekstu. Są to najczęściej różnego rodzaju spójniki, zaimki, przyimki, lub partykuły. Pozbycie się tych wyrazów z tekstu jest krokiem opcjonalnym - jeśli metoda analizy wykorzystuje informacje dotyczące składni i struktury zdania, pozostawienie *stop words* może wpłynąć korzystnie na wyniki.

— Ekstrakcja cech. Klasyfikacja z wykorzystaniem uczenia maszynowego, na podstawie zwykłego dokumentu tekstowego, nie jest możliwa (nawet jeśli tekst został wstępnie przetworzony i oczyszczony). Tekst musi zostać przekształcony do formy zrozumiałej dla klasyfikatora - wektora cech. Cech dla danego tekstu jest nieskończenie wiele, a identyfikacja tych najbardziej istotnych dla analizy nie jest zadaniem prostym [5]. Istnieje wiele metod wyboru

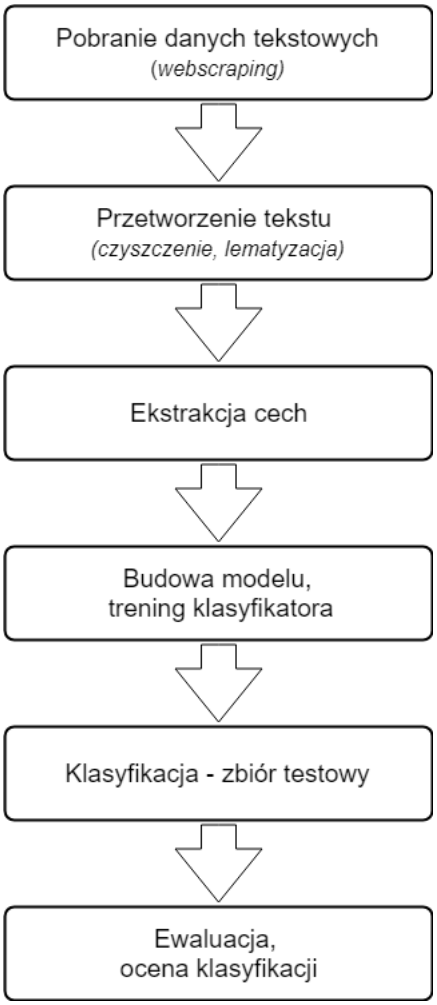
cech, w publikacjach proces ten nazywa się *feature selection/extraction*.

— Budowa modelu. Wykorzystując wyselekcjonowany z zebranych danych zbiór treningowy, przekształcony do postaci wektora cech, możliwa jest budowa modelu.

— Klasyfikacja. Etap, na którym wybrany algorytm uczenia maszynowego, na podstawie zbudowanego modelu, wybiera klasy dla nowych, niewidzianych do tej pory przez klasyfikator danych wejściowych.

— Ocena jakości klasyfikacji. Wyznaczenie miar jakości analizy - weryfikacja, czy klasyfikator w prawidłowy sposób wyznaczył klasy, dla danych ze zbioru testowego. Sprawdzenie jest możliwe, ponieważ zbiór testowy zawiera wzorcowe etykiety, które nie są dostepne i wykorzystywane podczas klasyfikacji - dopiero na ostatnim etapie oceny, prawidłowe klasy są użyte do porównania z tymi wyznaczonymi przez algorytm.

Na rysunku 4.1 przedstawiono kolejne kroki klasyfikacji w postaci diagramu.



Rysunek 4.1. Diagram przedstawiający kolejne etapy procesu klasyfikacji tekstu

4.2. Wektor cech jako reprezentacja tekstu

Jednym z etapów procesu klasyfikacji jest ekstrakcja cech. Należy to rozumieć, jako sprowadzenie tekstu to odpowiedniej reprezentacji, zrozumiałej dla klasyfikatora. Najczęściej jest to tzw. wektor cech, czyli zestaw właściwości charakteryzujących badany tekst, na podstawie którego algorytm uczenia maszynowego, jest w stanie dopasować odpowiednie klasy.

4.2.1. Model n-gramowy

Częstym zabiegem jest wykorzystywanie występowania poszczególnych termów z tekstu, jako jego cechę. W takim przypadku, mowa jest o unigramach, a jako ”najmniejszą” jednostkę analizy przyjmuje się pojedyncze wyrazy [7]. Jeśli do badania wykorzystywane są pary kolejnych słów, tworzą się bigramy. Uogólniając, łącząc ze sobą n wyrazów, otrzymujemy n-gramy. W takim modelu jednostką niekoniecznie musi być wyraz - może to być dowolnie wybrany ciąg znaków. Przykładowo w zdaniu:

Ona wczoraj obejrzała film.

Bigramy, jeśli jako jednostkę przyjmiemy słowa, są następujące:

[’Ona wczoraj’, ’wczoraj obejrzała’, ’obejrzała film’]

W procesie identyfikacji n-gramów w tekście, ważną rolę odgrywa opisana wcześniej *tokenizacja*.

Unigramy są powszechnie stosowane do analizy metodą *bag of words*. Wszystkie unikalne wyrazy, znajdujące się w analizowanym zbiorze dokumentów, są gromadzone w jednym ”worku”, który pełni rolę słownika. Zawiera on wszystkie słowa występujące w badanym korpusie. Następnie, dla każdego elementu ze słownika i dla każdego dokumentu, następuje weryfikacja, czy dany unigram występuje w analizowanym tekście (jest to najprostsza wersja, inną metodą jest zliczanie wystąpień). Jeśli tak, w wektorze cech, na pozycji związanej z danym wyrazem ze słownika, umieszcza się 1. W przeciwnym wypadku - 0. W ten sposób otrzymujemy reprezentację wektorową tekstu, na podstawie skonstruowanego słownika, zawierającego wszystkie słowa z korpusu. Cechą w tym przypadku jest występowanie określonego słowa, w związku z tym rozmiar wektora cech będzie równy liczbie wyrazów znajdujących się w słowniku.

Wraz ze wzrostem liczby dokumentów, rośnie rozmiar konstruowanego słownika, a co za tym idzie - wektora cech. Zbyt duża liczba cech jest zjawiskiem niepożądanym, klasyfikator w takiej sytuacji musi wziąć pod uwagę znaczącą liczbę szczególnych właściwości, często unikalnych dla pojedynczych dokumentów. W związku z tym, w celu ograniczenia słownika, stosuje się zabieg lematyzacji lub stemmingu. W ten sposób, wyrazy odmienione (ogólniej mówiąc - w różnych formach gramatycznych) zajmą to samo miejsce słowniku.

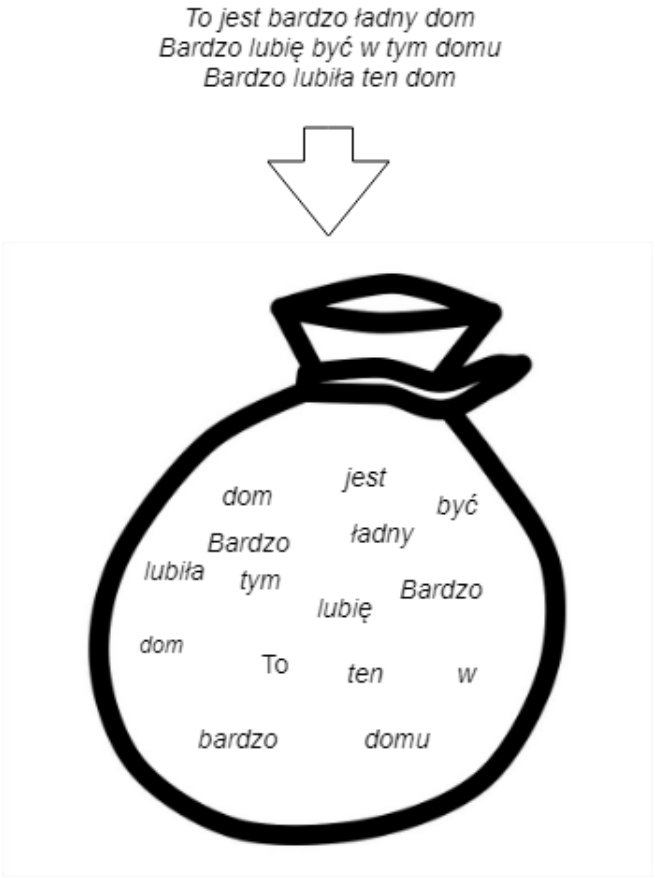
Załóżmy, że analizowany zbiór dokumentów prezentuje się następująco:

To jest bardzo ładny dom.
Bardzo lubię być w tym domu.
Bardzo lubiła ten dom.

Wtedy reprezentacja poszczególnych zdań zdefiniowana jest w tabeli 4.2.

Tekst	***to***	***być***	***bardzo***	***ładny***	***dom***	***lubić***	***w***
To jest bardzo ładny dom.	*1*	*1*	*1*	*1*	*1*	*0*	*0*
Bardzo lubię jeździć na rowerze.	*1*	*1*	*1*	*0*	*1*	*1*	*1*
Bardzo lubiła ten dom.	*1*	*0*	*1*	*0*	*1*	*1*	*0*

Tabela 4.2. Przykład reprezentacji zdań w postaci wektorowej w metodzie *bag of words*



Rysunek 4.2. Ilustracja obrazująca ideę *bag of words*

Bag of words jest popularną metodą reprezentacji tekstu w celu analizy jego wydźwięku. Zdecydowanym minusem takiego podejścia, którego ideę przedstawia rysunek 4.2, jest utrata kontekstu gramatycznego oraz relacji między występującymi w tekście wyrazami. Znaczenie występujących w badanym dokumencie zdań, nie jest w dużej mierze istotne. Przykładowo, analizując zdania:

Gdzieś to mam.

oraz

Mam to gdzieś.

Opierając się na metodzie *bag of words* oba te zdania otrzymałyby jednakową reprezentację, a co za tym idzie - wydźwięk. Nietrudno jest jednak wyczuć znaczącą różnicę w wydźwięku obu tych stwierdzeń, biorąc pod uwagę nie tylko występowanie poszczególnych słów, ale także ich kolejność.

4.2.2. Oznaczanie części mowy

Inną, często wykorzystywaną do konstrukcji wektora cech, właściwością wyrazów w tekście, są znaczniki morfosyntaktyczne, uzyskane za pomocą analizy morfologicznej. Znaczniki zawierają informacje takie jak:

— lemat, czyli forma bazowa słowa
— część mowy
— liczba
— przypadek
— rodzaj

Popularnym narzędziem, umożliwiającym analizę morfologiczną, jest *Morfeusz* [21]. Przykładowo, dla słowa *książkę*, rezultaty uzyskane za pomocą programu są następujące:

lemat -> książka
znaczniki -> subst:sg:acc:f

Gdzie:

— *subst*: rzeczownik
— *sg*: liczba pojedyncza
— *acc*: biernik
— *f*: rodzaj żeński

4.2.3. Cechy składniowe

Jako cechy badanego tekstu można wykorzystać informacje uzyskane z rozbioru zdania na drzewo np. składnikowe lub zależnościowe. W ten sposób można otrzymać precyzyjniejszą reprezentację wyrażeń tekstowych, która, oprócz danych dotyczących poszczególnych wyrazów budujących zbiór, zawiera także informacje na temat zasad konstrukcji zdań w danym języku. Przykładem takiej cechy mogą być pary węzłów, pochodzących ze skonstruowanego drzewa rozbioru [24] [23].

4.3. Naiwny klasyfikator Bayesa

Do klasyfikacji metodą nadzorowanego uczenia maszynowego można wykorzystać różne podejścia: SVM, drzewa decyzyjne, czy naiwny klasyfikator Bayesa. To właśnie ten ostatni, ze względu na swoją prostotę i skuteczność, jest jednym z najpopularniejszych algorytmów. Rodział przedstawia dokładny opis tego rodzaju klasyfikacji [7].

4.3.1. Definicja

Działanie klasyfikatora bayesowskiego wynika z rachunku prawdopodobieństwa. Jego nazwa pochodzi od nazwiska autora, który sformułował twierdzenie, będące podstawą klasyfikacji - twierdzenie Bayesa. Jest to klasyfikator probabilistyczny, tzn. dla każdego elementu (obserwacji) x ze zbioru danych wejściowych X, wyznacza prawdopodobieństwo przynależności do klasy y, należącej do zbioru wyjściowego Y.

Niech wejściem do modelu będzie zbiór dokumentów D, wyjściem - zbiór klas C. Zadanie klasyfikatora można wyrazić w następujący sposób:

$$\overline{c} = argmax\ P(c \mid d) \quad c \in C,\ d \in D \tag{4.1}$$

gdzie $\overline{c}$ oznacza klasę, dla której prawdopodobieństwo warunkowe $P(c \mid d)$ (prawdopodobieństwo *a posteriori* występowania danej klasy c pod warunkiem dokumentu d) przyjmuje wartość największą. Wykorzystując twierdzenie Bayesa, powyższe równanie można przekształcić do postaci:

$$\overline{c} = argmax\ P(c \mid d) = argmax\ \frac{P(d \mid c)P(c)}{P(d)} \tag{4.2}$$

Następnie, można zauważyć, że wartość $P(d)$ jest stała dla każdej klasy c. W związku z tym, mianownik wyrażenia 4.2 można pominąć, otrzymując:

$$\overline{c} = argmax\ P(d \mid c)P(c) \tag{4.3}$$

Wyznaczenie wartości $P(c)$ nie jest zadaniem trudnym (obliczenie występowania prawdopodobieństwa danej klasy dla korpusu). W jaki sposób można obliczyć $P(d \mid c)$? Odpowiedź na to pytanie nie jest oczywista - może zostać uzyskana opierając się na informacjach przedstawionych w rodziale 4.2.

Dokument reprezentowany jest jako zestaw cech $F = \{f_1, f_2, ..., f_n\}$, gdzie n jest rozmiarem wektora (liczbą wybranych cech). Z tego stwierdzenia wynika:

$$\overline{c} = argmax\ P(f_1, f_2, ..., f_n \mid c)P(c) \tag{4.4}$$

Prawdopodobieństwo występowania dokumentu, można zastąpić przez prawdopodobieństwo jednoczesnego występowania cech go reprezentujących (pod warunkiem danej klasy). W tym miejscu uwidacznia się naiwność klasyfikatora - zakłada się, że występowanie cech jest od siebie niezależne. Oznacza to, że 4.4 może zostać zastąpione przez:

$$\overline{c} = argmax\ P(f_1 \mid c)(f_2 \mid c)...(f_n \mid c)P(c) \tag{4.5}$$

Z równania 4.5 wynika, że każdą z cech tekstu, można rozpatrywać osobno, niezależnie od pozostałych (jest to dość mocne założenie, stąd nazwa metody). Obliczenie prawdopodobieństwa występowania danej cechy dla wybranej klasy jest stosunkowo proste, biorąc pod uwagę równanie wyjściowe 4.1.

W trakcie uczenia klasyfikatora, na podstawie zbioru treningowego, wyliczane są wymagane prawdopodobieństwa wybranych cech. Tak skonstruowany model, w postaci wyników obliczeń, wykorzystywany jest do klasyfikacji zbioru testowego.

4.3.2. Rodzaje klasyfikacji bayesowskiej

Ważną kwestią jest określenie rodzajów omawianego klasyfikatora, biorąc pod uwagę rozkład prawdopodobieństwa, na którym opiera się konstrukcja wektora cech [4]:

— Rozkład Bernoulliego o charakterystyce zero - jedynkowej. Wektor cech zawiera jedynie informacje, czy dana cecha występuje lub nie.

— Rozkład wielomianowy. Charakteryzuje się tym, że określa częstość występowania poszczególnych cech np. zliczanie słów w tekście.

— Rozkład Gaussa. Elementami wektora są wartości pochodzące ze zbioru ciągłego.

4.3.3. Wygładzanie Laplace'a

Opisane podejście zawiera pewien problem. W celu wyznaczenia odpowiedniej klasy, klasyfikator "naiwnie" mnoży wszystkie pozostałe prawdopodobieństwa. Co jeśli jeden z czynników okaże się równy 0? Całe wyrażenie przyjmuje taką wartość - brak jednej cechy, która pojawiła się w zbiorze testowym, a nie była obecna w zbiorze trenigowym (mowa jest o prawdopodobieństwie warunkowym, czyli dla danej klasy) w pewien sposób "zeruje" szansę danej klasy.

Jedną z metod rozwiązania tego problemu, jest zastosowanie techniki zwanej wygładzaniem Laplace'a. Polega ona na wprowadzeniu ustalonej, większej od zera wartości α, która dodawana jest do wyrażeń w liczniku i mianowniku przy wyliczaniu prawdopodobieństwa. Przykładowo, stosując metodę *bag of words* z cechami jako zliczanie wystąpień poszczególnych słów, prawdopodobieństwo dla *i-tego* słowa wynosi:

$$P(s_i \mid c) = \frac{(\textit{liczba wystąpień } s_i \textit{ z klasą } c)}{(\textit{liczba wystąpień wszystkich słów z klasą } c)} \tag{4.6}$$

W wersji bez wygładzania. Po dodaniu współczynków:

$$P(s_i \mid c) = \frac{(\textit{liczba wystąpień } s_i \textit{ z klasą } c) + \alpha}{(\textit{liczba wystąpień wszystkich słów z klasą } c) + \alpha|V|} \tag{4.7}$$

Gdzie $|V|$ oznacza rozmiar słownika. W ten sposób niemożliwe jest wyzerowanie licznika wyrażenia, jeśli wybrane słowo testowe nie występuje razem z daną klasą w zbiorze treningowym.

4.3.4. Przykład klasyfikacji

Jako przykład wykorzystania naiwnego klasyfikatora Bayesa o rozkładzie wielomianowym, zaprezentowana zostanie analiza wydźwięku zbioru dokumentów tekstowych, na podstawie zbioru treningowego w postaci tabeli 4.3.

Tekst	***Wydźwięk***
Bardzo dobra książka	*pozytywny*
Ciekawie się ją czytało	*pozytywny*
Nudna jak flaki z olejem	*negatywny*
Niesamowicie nudna książka	*negatywny*

Tabela 4.3. Przykładowy zbiór treningowy

Klasyfikacji poddane zostanie stwierdzenie X:

To jest nudna książka.

Zadaniem jest określenie wydźwięku powyższego zdania, wykorzystując naiwny klasyfikator Bayesa. Do analizy i konstrukcji słownika, wszystkie słowa poddane zostaną lematyzacji, a wielkie litery przekształcone na małe. Słowa, które występują w zbiorze testowym (*to*, *być*), a nie posiadają swoich reprezentantów w zbiorze treningowym, nie są brane pod uwagę podczas klasyfikacji. Wartość współczynnika wygładzania jest równa $\alpha = 1$.

Prawdopodobieństwa występowania dwóch możliwych klas - pozytywna przedstawiona jako p, negatywna jako n - są równe:

$$P(p) = P(n) = \frac{1}{2}$$

Prawdopodobieństwa dla poszczególnych słów testowego zdania, pod warunkiem danej klasy, prezentują się następująco:

$$P(\text{'nudna'} \mid p) = \frac{0+1}{7+13} = \frac{1}{20}$$

$$P(\text{'książka'} \mid p) = \frac{1+1}{7+13} = \frac{1}{10}$$

$$P(\text{'nudna'} \mid n) = \frac{2+1}{8+13} = \frac{3}{21}$$

$$P(\text{'książka'} \mid n) = \frac{1+1}{8+13} = \frac{2}{21}$$

$$P(p \mid X) = P(X \mid p)P(p) = \frac{1}{2} \cdot \frac{1 \cdot 1}{20 \cdot 10} = \frac{1}{400} = 0.0025$$

$$P(n \mid X) = P(X \mid n)P(n) = \frac{1}{2} \cdot \frac{3 \cdot 2}{21 \cdot 21} = \frac{3}{441} \approx 0.0068$$

Z czego wynika:

$$P(n \mid X) > P(p \mid X)$$

Wniosek - wynikiem klasyfikacji jest wydźwięk negatywny dla testowanego zdania.

4.4. Ewaluacja modelu

Istnieje kilka metod weryfikacji jakości klasyfikacji [7]. Najprostszą metodą jest wyznaczenie wartości dokładności *(ang. accuracy)*, tj. stosunku liczby poprawnie zaklasyfikowanych obserwacji do wszystkich obserwacji obecnych w zbiorze testowym. Przykładowo, jeśli zbiór testowy zawiera 50 dokumentów, a 30 z nich otrzymało prawidłową klasę, dokładność w takim przypadku wynosi 60%. Na jakiej podstawie możliwa jest weryfikacja, czy przydzielona klasa jest prawidłowa? Do tego celu, wymagana jest obecność etykiet wzorcowych, dla każdej z obserwacji ze zbioru testowego.

Dokładność nie zawsze jest jednak dobrą miarą jakości modelu. Wyobraźmy sobie zbiór testowy, w którym na 100 dokumentów, 90 ma wydźwięk negatywny, a 10 pozytywny. Wykonując prostą, ”naiwną” klasyfikację, dla której wszystkie obserwacje otrzymują klasę negatywną. W takim przypadku dokładność jest bardzo wysoka, równa 90%, lecz nie można jej uznać za miarodajną i wiarygodną. W związku z tym, istotne jest zbalansowanie zbioru danych, w taki sposób, by występowanie klas było w nim równomierne.

Często dokładność okazuje się niewystarczającą miarą (jak w przypadku powyżej), dlatego wykorzystuje się inne metryki, takie jak precyzja oraz czułość. Opierają się one na następujących miarach ”pomocniczych”:

— *true positive (TP)*: liczba poztywnych obserwacji zaklasyfikowanych jako pozytywne
— *true negative (TN)*: liczba negatywnych obserwacji zaklasyfikowanych jako negatywne
— *false positive (FP)*: liczba negatywnych obserwacji zaklasyfikowanych jako pozytywne
— *false negative (FN)*: liczba pozytywnych obserwacji zaklasyfikowanych jako negatywne

Na tej podstawie:

$$\textit{precyzja} = \frac{TP}{TP + FP}$$

$$\textit{czułość} = \frac{TP}{TP + FN}$$

Precyzja obrazuje, ile spośród obserwacji, które system oznaczył jako pozytywne, zostało zaklasyfikowanych poprawnie. Czułość natomiast sprawdza, ile spośród wszystkich pozytywnych obserwacji ze zbioru testowego, zostało przez system oznaczone jako pozytywne.

Miary te skupiają się na *true positives* - dobrze obrazują jakość detekcji i identyfikacji elementów rzadko występujących w zbiorze danych (z czym nie radziła sobie dokładność). Dla powyższego przykładu, gdzie dokładność była równa 90%, precyzja oraz czułość są równe 0 (brak *TP*), mimo wysokiej dokładności.

Wprowadzono także metrykę, która łączy precyzję (P) oraz czułość (C) zwaną *F-Measure* wyrażoną jako:

$$F = \frac{(\beta^2 + 1)PC}{\beta^2 P + C}$$

Gdzie β określa wagę pomiędzy precyzją i czułością. W szczególności, dla $\beta = 1$, obie te metryki posiadają taką samą wagę.

5. Projekt

Opisany w poniższym rodziale projekt magisterski, stara się powiązać ze sobą trzy przedstawione do tej pory w pracy aspekty: analizę wydźwięku, drzewo rozbioru oraz klasyfikację tekstu (do jednej z kategorii: pozytywna lub negatywna) za pomocą metod nadzorowanego uczenia maszynowego. Skupia się na weryfikacji, czy wykorzystanie niestandardowych cech tekstu, bazujących na jego rozbiciu na drzewo składniowe, poprawia lub pogarsza jakość analizy. Celem nie jest porównanie różnych metod klasyfikacji nadzorowanej - podstawą wszystkich wariantów badań jest klasyfikator bayesowski oparty na rozkładzie Bernoulliego. Analizowany jest przede wszystkim wpływ na wyniki wykorzystanego rodzaju reprezentacji tekstu.

5.1. Zbiór danych

Wymienione cele badań dotyczą korpusów w języku polskim. Podstawą analizy jest wybór odpowiedniego zbioru danych. Czym charakteryzuje się odpowiedni dla omawianego projektu zasób dokumentów tekstowych? Wymagania dla źródła zdefiniowane zostały następująco:

— Tekst wyraża opinię użytkownika na temat pewnego produktu.

— Opinia opatrzona jest wzorcową etykietą, z której w bezpośredni sposób można wywnioskować nacechowanie emocjonalne tekstu.

— Poprawność językowa. Tekst powinien zawierać możliwie mało błędów.

— Odpowiedni rozmiar korpusu. Im większy zbiór tekstów, tym trening modelu staje się efektywniejszy, a co za tym idzie - wzrasta wiarygodność klasyfikacji.

— Krótkie dokumenty, których długość jest ograniczona do kilku słów (w szczególności jednego), powinny stanowić niewielką część całego zbioru. Preferowane są dłuższe teksty, które można podzielić na zdania.

— Łatwość pobrania danych. Jeśli nie ma możliwości zgromadzenia opinii w sposób zautomatyzowany, zbiór staje się nieużyteczny.

Po przeanalizowaniu dostępnych w sieci portali, na których użytkownik posiada możliwość wyrażenia opinii tekstowej wraz z oceną, i uwzględnieniu powyższych wymagań, zdecydowano się na wykorzystanie danych zawartych w serwisie *lubimyczytac.pl*.

Pod tym adresem, dostępny jest jeden z najpopularniejszych portali internetowych do gromadzenia informacji na temat opublikowanych książek, czasopism czy nawet ekranizacji niektórych pozycji. Najważniejszą funkcjonalnością serwisu, jest umożliwienie użytkownikom wyrażania opinii i wystawienia ocen recenzowanych książek w postaci gwiazdek - w skali od 1 do 10. Serwis zgromadził dużą społeczność, więc dostępny zbiór danych jest pokaźnych rozmiarów.

Ważnym aspektem decydującym o wyborze źródła danych, była poprawność gramatyczna zawartych w nim tekstów. Internauci, korzystający z serwisu *lubimyczytac*, rzadko popełniają błędy językowe (oczywiście porównując do innych miejsc w sieci, gdzie użytkownik może wyrazić swoje zdanie słowem pisanym). Teksty zazwyczaj składają się z co najmniej kilku zdań, posiadających poprawną strukturę gramatyczną. Ograniczona liczba literówek i używanych symboli (emotikonek) również wpływa pozytywnie na ocenę portalu. Złożoność niektórych zdań ma także swoje minusy - takie wyrażenia stanowią spore wyzwanie dla wykorzystywanych podczas badań narzędzi do analizy tekstu dla języka polskiego.

Jak wspomniano wyżej, każda z opinii danej pozycji, oprócz komentarza słownego, opatrzona została oceną w postaci wystawionej liczby gwiazdek. Zbiór zawiera także elementy, które posiadały ocenę, ale bez uzasadniającej ją treści. Takie opinie nie wnoszą żadnej wartości podczas badania wydźwięku, w związku z czym, nie zostały uwzględnione w zbiorze uczącym.

Dane do analizy (tekst opinii wraz z oceną) zostały pobrane metodą *webscrapingu*. Więcej o szczegółach implementacyjnych w rozdziale 5.5.

Spośród dostępnych opinii, dla ponad 280000 wszystkich ocenionych książek, wstępnie pobranych zostało 100000. Z tego zbioru wytypowanych zostało 10000 recenzji pozytywnych oraz taka sama (zbiór uczący powinien być zbilansowany, stąd równy podział na klasy) liczba negatywnych. W jaki sposób opinia otrzymywała dane nacechowanie emocjonalne? Zastosowano prostą metodę - jeśli liczba gwiazdek znajduje się w przedziale $[1, 3]$, to wyraża opinię negatywną, natomiast dla wartości z przedziału $[7, 10]$ - pozytywną. W ten sposób ze zbioru wyeliminowano również elementy o wydźwięku neutralnym. Ostatecznie, zbiór wykorzystany w projekcie, składa się z 20000 opinii, każda oznaczona ”wzorcową” klasą. Rysunek 5.1 ukazuje przykładowe dane dostępne na omawianym portalu.

Rysunek 5.1. Przykładowe informacje dla wybranej książki z portalu *lubimyczytac.pl*

5.2. Wykorzystane narzędzia

Automatyczna analiza wydźwięku jest skomplikowanym procesem. Praktycznie niemożliwym jest przeprowadzenie, dających sensowne rezultaty, badań, niewykorzystując i nie opierając się na efektach pracy innych osób, zajmujących się tematyką analizy sentymentu. Obecnie, w sieci dostępnych jest wiele narzędzi i zasobów ułatwiających pracę z tekstem. Poczynając od prostych słowników nacechowania emocjonalnego, kończąc na zaawansowanych parserach i analizatorach morfologicznych zdań.

Znaczną część stanowią narzędzia dla języka angielskiego, ale projekt skupia się na analizie języka polskiego. W związku z tym, przedstawiono zasoby, które z powodzeniem zostały wykorzystane do analizy, ale także te, które, pomimo kilku podejść i prób, nie znalazły zastosowania podczas badań.

5.2.1. Clarin-PL

Clarin-PL jest polską częścią europejskiej infrastruktury naukowej *CLARIN (Common Language Resources and Technology Infrastructure)* [25]. Jej zadaniem jest ułatwienie pracy osobom, instytucjom lub innym podmiotom, zajmujących się szeroko pojętą analizą tekstu. Przede wszystkim udostępnia wiele korpusów tekstowych oraz bogaty zestaw narzędzi, umożliwiających badania nad dokumentami. Usługami udostępnionymi przez *Clarin-PL* są m. in. *Spejd (parser składniowy)*, *Morfeusz (analizator morfologiczny)*, czy *WCRFT2 (tagger)*. Możliwości tego ostatniego zostały wykorzystane w projekcie.

Tagger *WCRFT2* [30] jest to narzędzie, służące do tokenizacji zdań oraz określania znaczników morfosyntaktycznych (których definicja została wprowadzona w rodziale 4). Udostępnia przystępny interfejs programistyczny w postaci usługi *RESTowej* [1]. Na wejściu wymagane jest podanie parametrów w postaci wybranego rodzaju taggera oraz tekstu, który powinien zostać poddany analizie. Na wyjściu narzędzie zwraca plik w formacie *xml*, który zawiera reprezentację analizowanego tekstu w postaci listy tokenów wraz z ich formą bazową.

Jest to wygodne w użyciu narzędzie, na którym, w przypadku konieczności zastosowania lematyzacji można zdecydowanie polegać.

[1] REST (ang. Representational State Transfer) - styl projektowania oprogramowania, opierający się na udostępnianiu usług w postaci zasobów, gdzie komunikacja między stronami opiera się na określonych regułach i ograniczeniach.

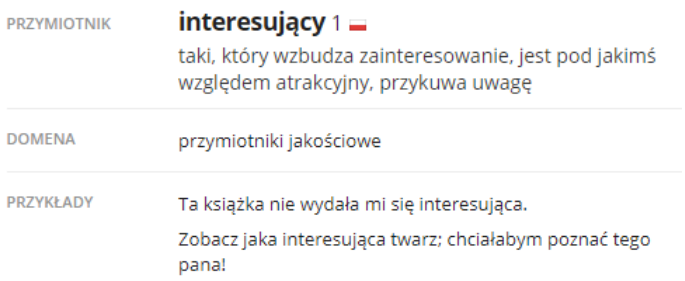
5.2.2. Słowosieć

Następnym serwisem wykorzystanym podczas analizy jest *Słowosieć* [29]. Jest to sieć semantyczna, wzorowana na najpopularniejszym obecnie tego rodzaju rozwiązaniu na świecie - *WordNet Princeton* (z którym jest także w pełni zintegrowana). Inaczej mówiąc, jest to baza danych słów (ponad 190000) dla języka polskiego, połączonych ze sobą różnego rodzaju relacjami. *Słowosieć* określa kilka typów relacji między elementami sieci, są to m. in.:

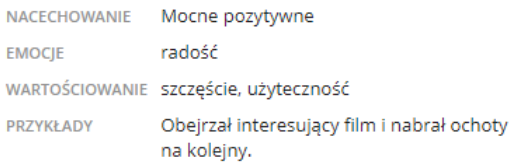
— synonimy
— antonimy
— hiperonimy, czyli wyrazy nadrzędne (o ogólniejszym znaczeniu) dla wybranego słowa
— hiponimy, czyli wyrazy podrzędne (o bardziej szczególnym znaczeniu) dla wybranego słowa.

Dwa ostatnie punkty z listy są ze sobą ściśle związane - jeśli słowo A jest hiperonimem dla słowa B, to B jest hiponimem dla A.

Słowosieć to nie tylko relacje między słowami. Dla ok. 80000 jednostek ze słownika, określone zostało ich nacechowanie emocjonalne, wraz z przykładami zastosowań w zdaniu. Ta własność *Słowosieci* jest szczególnie istotna w kontekście opisywanej pracy, ponieważ bezpośrednio dotyczy analizy wydźwięku. Na rysunkach 5.2 i 5.3 przedstawiono przykład wyniku odpytania serwisu dla słowa *interesujący*.



Rysunek 5.2. Informacje dla słowa *interesujący* pochodzące ze *Słowosieci*



Rysunek 5.3. Nacechowanie emocjonalne słowa *interesujący* pochodzące ze *Słowosieci*

Słowosieć, podobnie do taggera *WCRFT2*, udostępniona jest jako usługa *RESTowa* w ramach projektu *Clarin-PL*. Wynik zwracany jest w postaci pliku w formacie *json*, więc automatyczne wyszukiwanie relacji np. synonimów, nie jest zadaniem skomplikowanym. Ponadto, aby wykorzystać informacje dotyczące nacechowania emocjonalnego poszczególnych słów, nie ma konieczności wysyłania żądań do wystawionej usługi - istnieje możliwość pobrania pliku,

zawierającego wszystkie dostępne jednostki z określonym wydźwiękiem, który pełni rolę słownika.

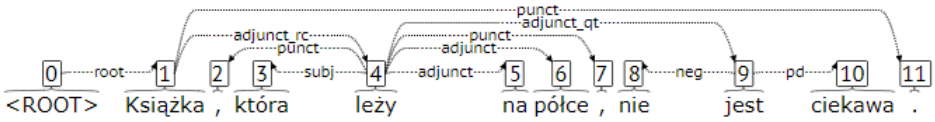
5.2.3. Multiservice NLP

Praca zajmuje się analizą, która stosuje rozbiór zdania na części składowe - w szczególności, wymaga przedstawienia wyrażeń w postaci drzew rozbioru. Istnieje kilka narzędzi umożliwiających zautomatyzowane przekształacanie zdań do żądanej formy, lecz jedno z nich się wyróżnia, ze względu na skuteczność działania oraz wygodę użytkowania - *Multiservice NLP* [13].

Serwis umożliwia m. in. wykrywanie w tekście grup składniowych, nazw własnych, czy parsowanie zależnościowe. Ostatnia wymieniona funkcjonalność, wykorzystana w projekcie, przekształca podany tekst do postaci drzewa zależnościowego. Narzędzie udostępnia interfejs graficzny w formie aplikacji internetowej oraz pobranie wyników analizy w postaci pliku w formacie *json*, więc automatyzacja nie stanowi problemu. Przykładowo, dla zdania:

Książka, która leży na półce, nie jest ciekawa.

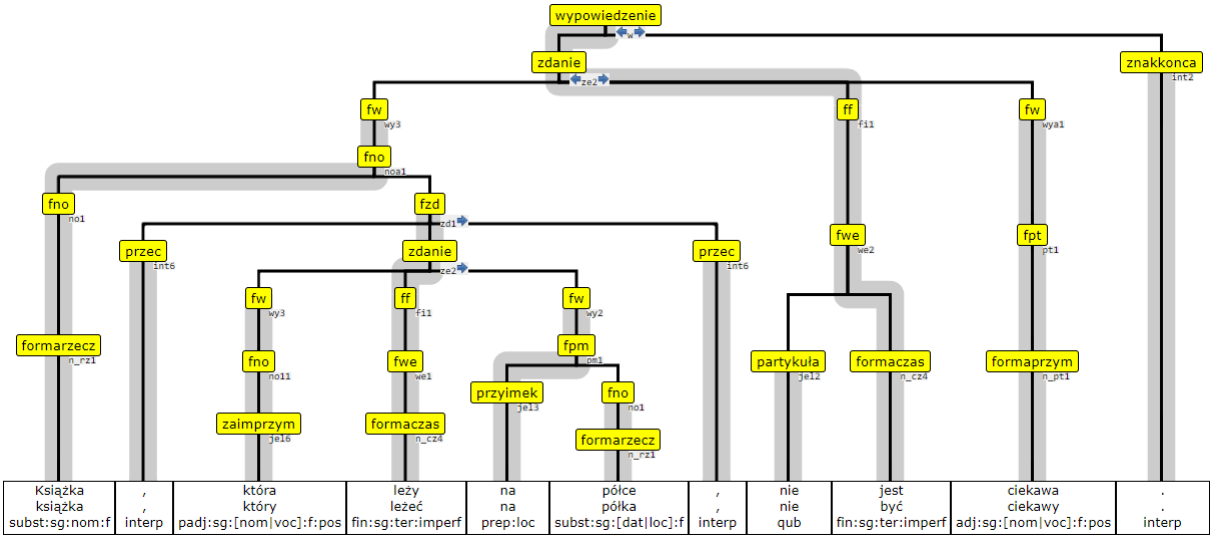
Narzędzie zwraca wynik widoczny na rysunku 5.4.



Rysunek 5.4. Przykład rozbicia zdania na drzewo zależnościowe z wykorzystaniem *Multiservice NLP*

5.2.4. Świgra 2

Ciekawą opcją w przypadku przetwarzania i transformacji zdań jest *Świgra 2* [20]. Narzędzie, podobnie do *Multiservice NLP*, oferuje rozkład podanego tekstu na drzewo, lecz nie zależnościowe, a składnikowe. W przypadku występowania niejednoznaczności, zwracany jest cały las. Dla przykładowego zdania z poprzedniego podrozdziału, rezultat prezentuje się na rysunku 5.5.



Rysunek 5.5. Przykład rozbicia zdania na drzewo składnikowe z wykorzystaniem *Świgra 2*

Świgra 2 jest narzędziem o bardzo dużych możliwościach, które pozwala na przeprowadzenie dogłębnej analizy tekstu, wykorzystując składnikowe drzewo rozbioru. Niestety, ze względów wydajnościowych udostępnionej wersji aplikacji w sieci (procesowanie powyższego zdania zajęło ponad 2 minuty - w przypadku analizy 20000 opinii jest to przeszkoda nie do pokonania), nie zdecydowano się na zastosowanie serwisu w projekcie.

Dokumentacja *Świgry 2* zaznacza, że problemy wydajnościowe w przypadku serwisu mogą wystąpić. Z tego względu, dla analizy większych zbiorów tekstowych, zalecana jest instalacja i uruchomienie narzędzia na dedykowanej do tego celu maszynie. Biorąc pod uwagę związaną z takim krokiem niepewność (działanie aplikacji w innych warunkach może nie ulec poprawie), pozostano przy wyborze *Multiservice NLP*, jako narzędzia do transformacji zdań na drzewa rozbioru.

5.2.5. Biblioteka *NLTK*

NLTK, czyli *Natural Language Toolkit* [26], jest biblioteką dla języka programowania *Python*, która umożliwia analizę tekstu z poziomu kodu własnej aplikacji. Moduł udostępnia gotowe korpusy oraz wiele narzędzi do procesowania języka naturalnego - tokenizacja, lematyzacja, czy parsowanie. Biblioteka wspiera również klasyfikację tekstu z wykorzystaniem uczenia maszynowego, w szczególności dostarcza implementację, wykorzystanego w projekcie, klasyfikatora bayesowskiego.

NLTK jest ciekawą opcją, nie tylko ze względu na możliwości i funkcjonalności modułu, ale także jako źródło wiedzy na temat przetwarzania języka naturalnego. Autorzy biblioteki udostępnili w sieci książkę na ten temat [27], z którą warto się zaznajomić, ponieważ zawiera wiele cennych informacji dotyczących ogólno pojętej analizy tekstu.

5.2.6. Scala Scraper

Gromadzenie danych z wybranego źródła opiera się na metodzie *webscrapingu* - pobraniu zawartości strony internetowej, ekstrakcji z niej wymaganych danych i zdefiniowaniu ustrukturyzowanego modelu. Dostępnych jest wiele narzędzi, pozwalających na parsowanie plików w formacie *html* i udostępniających zestaw funkcjonalności do pracy na utworzonym modelu. Najpopulaniejsze rozwiązania przeznaczone są dla języka *Python*, jednak preferowanym wyborem było narzędzie współpracujące z językiem o silnym i statycznym typowaniu - *Scalą*.

Zdecydowano się na wykorzystanie niedużej biblioteki *Scala Scraper* [28], która spełnia wszystkie postawione wymagania. Jest narzędziem intuicyjnym i wygodnym w użyciu dzięki udostępnionemu interfejsowi programistycznemu. Rysunek 5.6 ilustruje proces *webscrapingu*.

Rysunek 5.6. Przykład procesu webscrapingu dla wybranej opinii

5.3. Metodyka przeprowadzonych badań

5.3.1. Główne metody analizy

Opisana analiza skupia się na dwóch głównych sposobach reprezentacji tekstu - *bag of words* oraz za pomocą *kolokacji syntaktycznych*. Drugi z tych terminów odnosi się do par występujących w tekście. Parowanie tokenów wykonano na dwa sposoby:

— łączenie w pary dwóch kolejno występujących po sobie wyrazów w zdaniu. Innymi słowy, w ten sposób generowane są bigramy, z tokenem w postaci pojedynczego wyrazu. Takie pary w pracy określane są jako *kolokacje liniowe*

— łączenie w pary dwóch sąsiadujących ze sobą węzłów zależnościowego drzewa rozbioru (wygenerowane za pomocą *Multiservice NLP*). Stosując nazewnictwo dla struktur drzewiastych, pierwszym elementem w parze jest węzeł pełniący rolę rodzica, natomiast drugim - węzęł będacy dzieckiem w relacji. Takie pary są w pracy nazywane jako *kolokacje zależnościowe*.

Przeprowadzone badania można podzielić na cztery główne ścieżki, biorąc pod uwagę sposób reprezentacji tekstu oraz wykorzystane cechy dokumentu:

— reprezentacja tekstu jako *bag of words*, poszczególne słowa jako tokeny, wyrazy z pozostawioną formą fleksyjną (brak lematyzacji). Dla ułatwienia identyfikacji, alias przyjęty dla metody: BOW_{raw}

— reprezentacja tekstu jako *bag of words*, poszczególne słowa jako tokeny, wyrazy sprowadzone do formy bazowej (zastosowanie lematyzacji). Dla ułatwienia identyfikacji, alias przyjęty dla metody: BOW_{lem}

— reprezentacja tekstu w postaci *kolokacji liniowych*, wyrazy w formie bazowej. Dla ułatwienia identyfikacji, alias przyjęty dla metody: COL_{lin}

— reprezentacja tekstu w postaci *kolokacji zależnościowych*, wyrazy w formie bazowej. Dla ułatwienia identyfikacji, alias przyjęty dla metody: COL_{dep}.

Wszystkie opcje wykorzystują do analizy naiwny klasyfikator Bayesa o rozkładzie Bernouliiego. Z tego wynika, że wektor cech reprezentujący tekst, dla metod *bag of words* (BOW_{raw} oraz BOW_{lem}) zawiera jedynie informację, czy dane słowo występuje w dokumencie. W przypadku analizy dotyczącej kolokacji (COL_{lin} oraz COL_{dep}) jest podobnie - wektor posiada ”binarną” wiedzę na temat obecności poszczególnych par wyrazów.

Klasyfikator otrzymuje do dyspozycji zbiór 20000 opinii, z których 85% tworzy zbiór treningowy, natomiast pozostałe 15% - testowy.

5.3.2. Warianty głównych ścieżek badawczych

Każda z głównych ścieżek (BOW_{raw}, BOW_{lem}, COL_{lin}, COL_{dep}) została poddana indywidualnej analizie, polegającej na implementacji różnego rodzaju filtrów lub "usprawnień" dla cech. W ten sposób, każda z metod generuje wyniki dla kilku wariantów. Modyfikacje, zaaplikowane dla poszczególnych ścieżek, opisane zostały poniżej.

Dla BOW_{raw}:

— brak filtru, wszystkie wyrazy brane pod uwagę podczas klasyfikacji
— filtrowanie *stop words* [31]
— filtrowanie słów nieposiadających nacechowania emocjonalnego.

Dla BOW_{lem} wykorzystano takie same filtry jak w przypadku BOW_{raw}, z tą różnicą, że dla BOW_{lem} wszystkie słowa są w formie bazowej.

Dla COL_{lin}:

— brak filtru, wszystkie kolokacje brane pod uwagę podczas klasyfikacji
— filtrowanie kolokacji nieposiadających nacechowania emocjonalnego, tzn. oba słowa budujące parę nie posiadają wydźwięku pozytywnego ani negatywnego
— filtrowanie kolokacji nieposiadających nacechowania emocjonalnego, a w ich miejsce dodanie nowych par, w których słowa bez określonego sentymentu, zastępowane są przez zabarwione emocjonalne synonimy (pochodzące ze *Słowosieci*).

Dla COL_{dep} zastosowano podobne warianty do przypadku COL_{lin}, z jednym dodatkiem - dla każdej pary, przedstawiającej relację węzłów w drzewie rozbioru, dodano typ relacji. Przykładowo:

('jest', 'nie', 'neg')

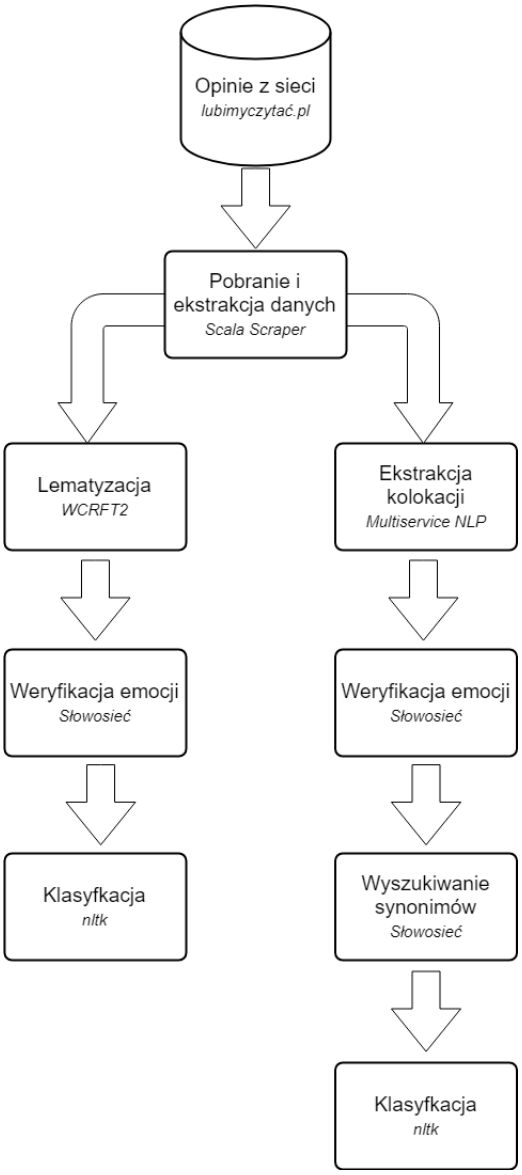
Co oznacza, że między elementami pary (*jest* oraz *nie*), zachodzi relacja negacji (zdefiniowana przez znacznik *neg*).

Dodatkowo, każdy wariant przeanalizowany został pod kątem wpływu rozmiaru wektora cech na wyniki badań. Liczbę słów (BOW_{raw} oraz BOW_{lem}) czy kolokacji (COL_{lin} i COL_{dep}) ograniczono, wykorzystując do analizy najczęściej występujące w całym korpusie elementy. W ten sposób uzyskano wyniki dla każdego wariantu i zdefiniowanego progu ograniczającego rozmiar wektora cech.

5.3.3. Analiza łączona

Analizę łączoną przeprowadzano dobierając w pary metody i warianty, dla których uzyskano najlepsze rezultaty. Polegało to na konstrukcji wektora cech, składającego się z najczęściej występujących elementów pochodzących z reprezentacji wektorowej każdej ze sparowanych metod, zachowując przy tym stały rozmiar wektora. Dla każdej pary modyfikowany był udział procentowy cech poszczególnych metod w wyjściowym wektorze.

5.4. Etapy całego procesu



Rysunek 5.7. Diagram przedstawiający ścieżki procesu analizy

Na rysunku 5.7 przedstawiono kolejne etapy procesu badawczego, wraz określonymi narzędziami, wykorzystanymi w danym kroku. Widoczny jest podział na dwie niezależne ścieżki - z lewej strony ukazana została analiza opierająca się na *bag of words*, natomiast z prawej wykorzystująca kolokacje syntaktyczne.

Obrazuje to w pewien sposób ideę, jaka przyświecała pracy - porównanie klasycznego podejścia do reprezentacji tekstu z inną, bardziej wyszukaną metodą, wykorzystującą cechy składniowe zdania.

5.5. Szczegóły implementacji

Podczas prac nad projektem powstały dwie niezależne od siebie aplikacje, będące podstawą dla procesu analizy. Pierwsza z nich zaimplementowana została w języku *Scala*. Jej listę odpowiedzialności można przedstawić jako:

— pobranie danych i ekstrakcja z nich informacji o opiniach (za pomocą *Scala Scraper*)
— obróbka tekstu (komunikacja z serwisem *WCRFT2*)
— transformacja tekstu na kolokacje (komunikacja z *Multiservice NLP*)
— weryfikacja nacechowania emocjonalnego (czytanie ze słownika wydźwięku w postaci pliku)
— wyszukiwanie synonimów (komunikacja ze *Słowosiecią*)
— zapis przeprocesowanych wyników do bazy danych.

Druga aplikacja, napisana przy użyciu języka *Python*, ma następujące zadania:

— pobranie z bazy danych opinii w przygotowanym formacie
— tokenizacja treści
— zdefiniowanie wektora cech
— trening modelu i klasyfikacja (wykorzystując moduł *NLTK*).

Generalnie, pierwsza z aplikacji odpowiada za pobranie i przygotowanie danych do analizy, natomiast druga, za przeprowadzenie zadania klasyfikacji. Ze względów wydajnościowych, zdecydowano się na wykorzystanie bazy danych, której odpowiedzialnością jest przechowywanie wyników przetwarzania tekstu, dla każdego rodzaju obróbki z osobna. Dzięki takiemu zabiegowi, proces należy wykonać tylko raz, co w przypadku konieczności pobrania i obróbki 20000 opinii tekstowych o zróżnicowanej długości, może przynieść wymierną korzyść w postaci oszczędności czasu. W ten sposób, aplikacja zajmująca się klasyfikacją, ma bezpośredni dostęp do danych w kilku postaciach.

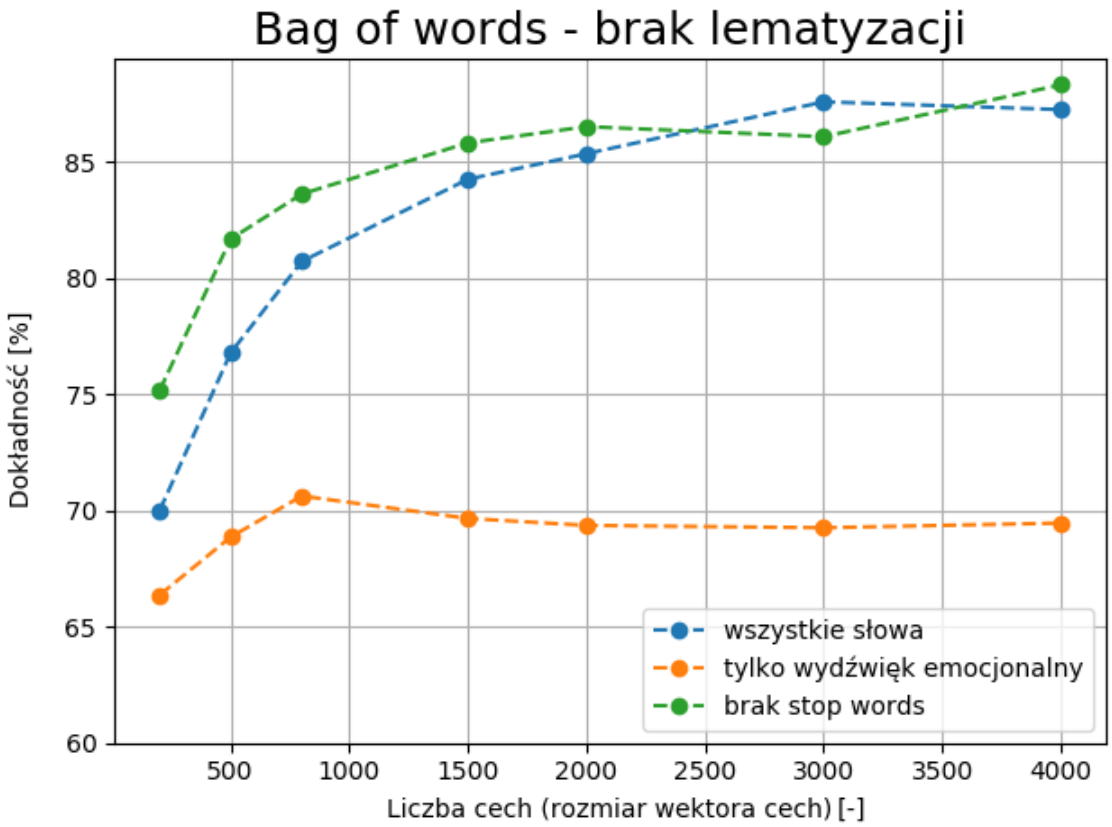
Silnikiem bazy danych wykorzystanym w projekcie jest *PostgreSQL* (rysunek 5.8).

Rysunek 5.8. Schemat komunikacji w systemie

5.6. Wyniki

Rozdział przedstawia wyniki dla podstawowych oraz łączonych metod analizy w postaci wykresów. Sekcja zawiera także spis najbardziej informatywnych cech ukazanych w tabeli.

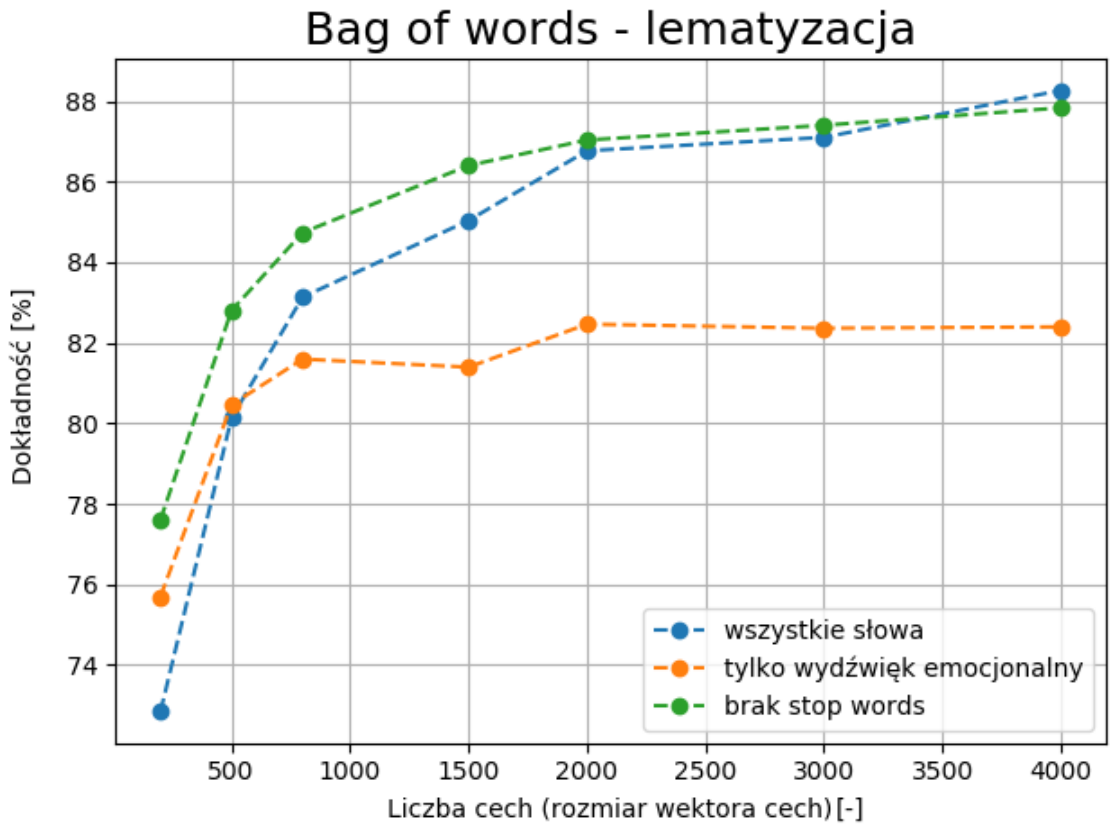
5.6.1. Metody podstawowe



Rysunek 5.9. Wyniki dla metody *bag of words* - wyrazy z pozostawioną formą fleksyjną

Pierwsza metoda - BOW_{raw} - przeanalizowana została w trzech wariantach. Na wykresie (rysunek 5.9), dwa warianty - korzystający ze wszystkich słów oraz filtrujący *stop words* - górują nad trzecim, który używa tylko wyrazów z nacechowaniem emocjonalnym. Tak duża różnica wynika z niedoskonałości ostatniego wariantu. Ze względu na to, że słowa nie są sprowadzane do formy bazowej, trudno jest zweryfikować wydźwięk badanego wyrazu. Słownik nacechowania emocjonalnego zawiera w przeważającej większości wyrażenia w formie bazowej, więc słowa z pozostawioną formą fleksyjną często nie znajdują swojego odpowiednika w słowniku. Efektem tego jest znacząco przerzedzony tekst opinii, a co za tym idzie - niska dokładność klasyfikacji.

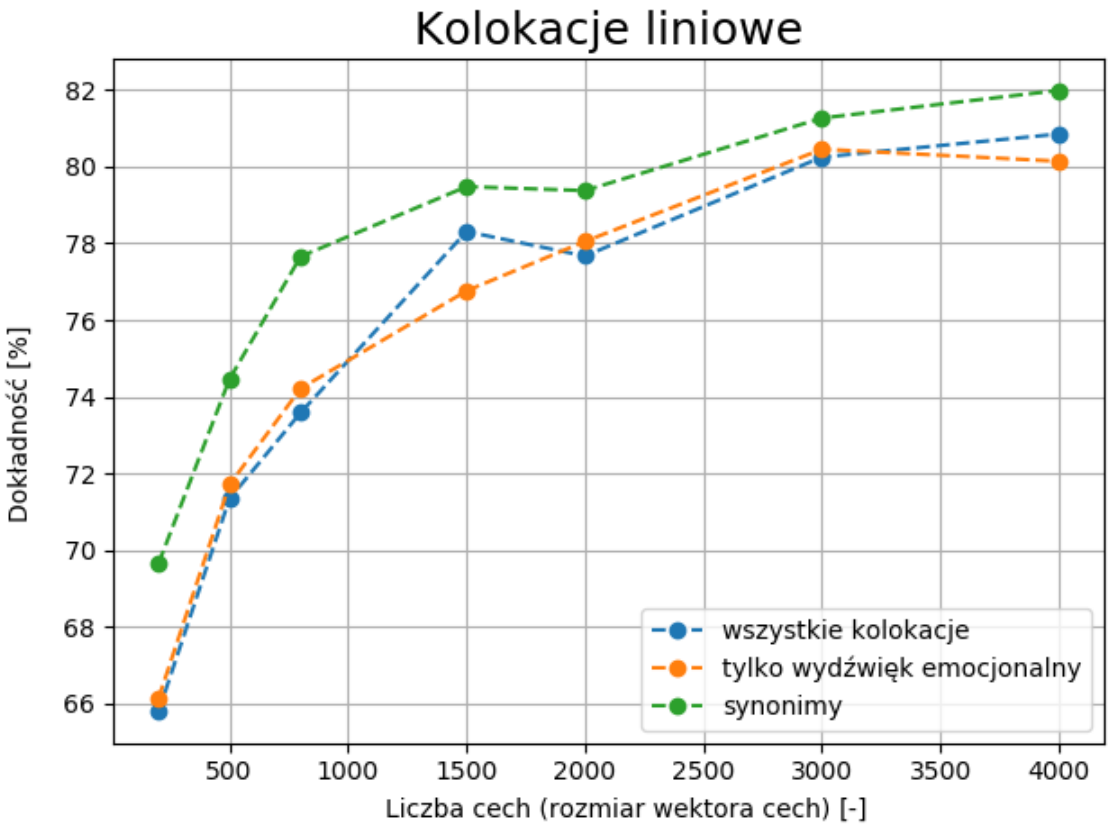
Wyniki dla dwóch pierwszych wariantów są zbliżone. Dla większości rozmiarów wektora cech, lepsze rezultaty uzyskano dla metody z filtrowaniem *stop words*, natomiast drugi wariant w pewien sposób ”goni” lidera. Najlepszy wyniki dla tej metody, niezależnie od wariantu, uzyskano dla filtrowania *stop words* i liczby cech równej 4000. Widoczny jest trend - wraz z wzrostem rozmiaru wektora cech, rośnie dokładność klasyfikacji.



Rysunek 5.10. Wyniki dla metody *bag of words* - wyrazy poddane lematyzacji

Drugą analizowaną metodą jest BOW_{lem}. Badania przeprowadzono dla takich samych wariantów jak dla metody BOW_{raw}. Na rysunku 5.10 widać znaczącą poprawę dokładności dla wariantu niewykorzystującego słów z nacechowaniem emocjonalnym. Ta metoda naprawia błąd popełniony przez poprzednika - weryfikacja wydźwięku odbywa się na wyrazach w formie podstawowej, co powoduje, że trafienia w elementy słownika sa częstsze. To, co łączy obie metody, to podobieństwo między wynikami dla dwóch ”liderów” wśród wariantów. Brak *stop words* daje lepsze rezultaty dla większości przypadków, ale ostatecznie, najlepszą dokładność uzyskano dla wariantu wykorzystującego wszystkie słowa.

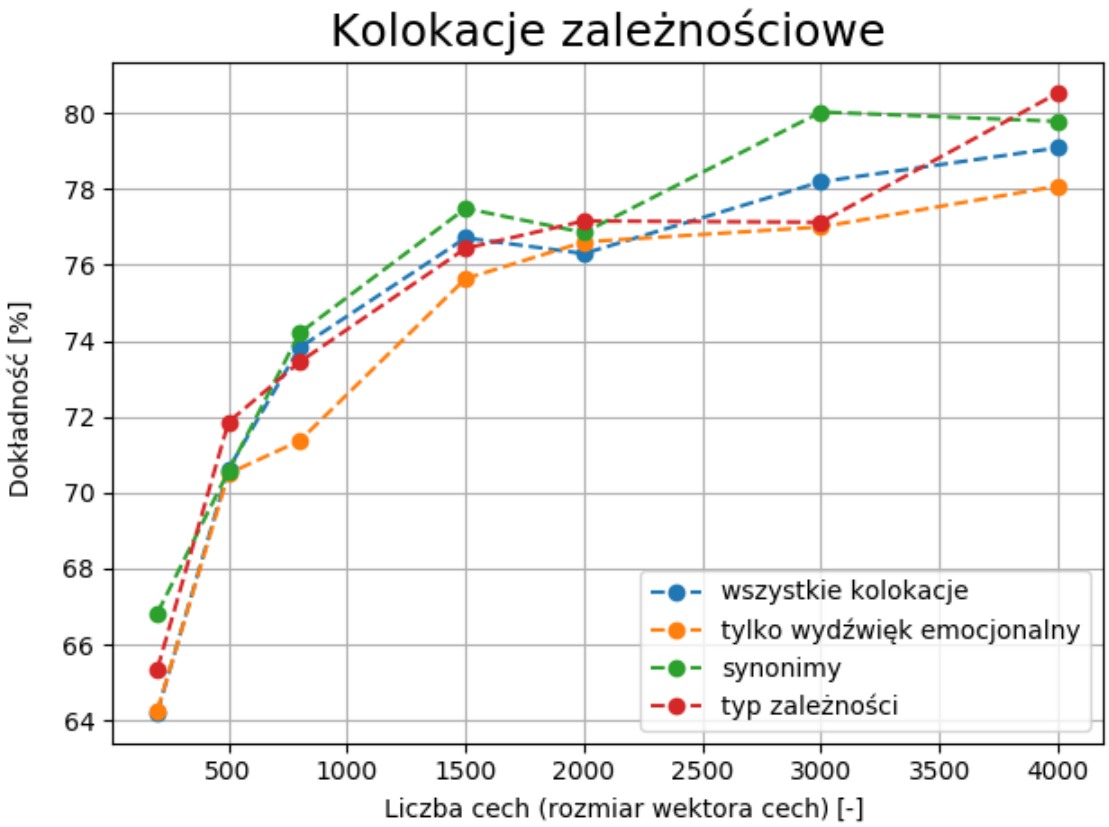
Mimo poprawy wyników dla metody operującej na wyrazach z wydźwiękiem, dokładność wciąż odbiega od najlepszych rezultatów. Skupienie się na słowach (wykorzystując podejście *bag of words*), które wyrażają emocje, nie poprawia jakości analizy sentymentu.



Rysunek 5.11. Wyniki dla metody wykorzystującej *kolokacje liniowie*

Kolejną metodą jest COL_{lin}, opierająca się na kolokacjach liniowych. Wyniki (rysunek 5.11) dla trzech analizowanych wariantów są zbliżone, jednak niewielką przewagę posiada metoda, która pozbywając się z tekstu słów bez wydźwięku, zastępuje je synonimami posiadającymi nacechowanie emocjonalne.

Weryfikacja emocji nie ma bezpośredniego wpływu na jakość klasyfikacji - osiągane rezultaty są praktycznie takie same jak dla wariantu wykorzystującego wszystkie kolokacje do analizy. Dopiero zastąpienie wyrazów ”bez emocji” poprawia dokładność badania.



Rysunek 5.12. Wyniki dla metody wykorzystującej *kolokacje zależnościowe*

Ostatnią z głównych metod badawczych jest COL_{dep} - analiza wykorzystująca kolokacje zależnościowe. Na podstawie wykresu (rysunek 5.12) można wyciągnąć, podobne do poprzedniej metody, wstępne wnioski - pozbycie się kolokacji bez wydźwięku, nie prowadzi do korzystnych rezultatów, natomiast zastąpienie wyrazów budujących te kolokacje synonimami, powoduje poprawę wyników.

Nowym podejściem, nie użytym do tej pory w omawianych wariantach do analizy kolokacji, jest wykorzystanie typu zależności między wyrazami w parze jako cechę tekstu. Ta dodatkowa informacja pozytywnie wpływa na dokładność klasyfikacji - dla 4000 cech uzyskano najlepszy wynik w zakresie analizy kolokacji zależnościowych.

Warianty, dla których uzyskano najlepsze wyniki, zestawione zostały w formie tabeli, zawierającej dokładność, czułość oraz precyzję zarówno dla klasy pozytywnej jak i negatywnej. Dodatkowo, przedstawiono rozmiar wektora cech, dla którego uzyskano najlepszy rezultat.

Metoda	*Wariant*	*L. cech*	*Dokładność*	*Czułość[poz]*	*Precyzja[poz]*	*Czułość[neg]*	*Precyzja[neg]*
BOW_{raw}	brak stop words	4000	**88,3%**	86,6%	89,9%	90%	86,7%
BOW_{lem}	wszystkie słowa	4000	**88,2%**	89,3%	87,7%	87,7%	88,8%
COL_{lin}	synonimy	4000	**81,9%**	87,3%	78,5%	76,7%	86,1%
COL_{dep}	typ zależności	4000	**80,5%**	88%	76,8%	72,8%	85,6%

Tabela 5.1. Najlepsze wyniki spośród wariantów analizy, dla każdej z metod podstawowych

Na podstawie tabeli 5.1 można wyciągnąć następujące wnioski:

— Dokładność dla metod wykorzystujących *bag of words* (BOW_{raw}, BOW_{lem}) jest o kilka punktów procentowych większa od podejść opartych na kolokacjach (COL_{lin}, COL_{dep}). Parowanie słów oraz wykorzystanie informacji pochodzących z drzewa rozbioru, nie przyniosły efektów w postaci poprawy jakości klasyfikacji.

— Dokładność BOW_{raw} oraz BOW_{lem} jest bardzo zbliżona. Różnica równa 0,1% sprawia, że można oba podejścia traktować jako równo efektywne metody analizy wydźwięku.

— Dokładność COL_{lin} oraz COL_{dep} jest zbliżona. Niewielką jednak przewagę (1,4%) posiada COL_{lin}. Oznacza to, że spośród wszystkich analizowanych metod, COL_{dep}, oparte na bardziej zaawansowanym sposobie ekstracji cech tekstu (porównując do pozostałych opisywanych metod), uzyskuje wyniki najsłabsze.

— Dla wszystkich metod i wariantów jeden trend jest wyraźnie widoczny - zwiększenie rozmiaru wektora cech reprezentującego tekst, wpływa pozytywnie na dokładność klasyfikacji. Wszystkie najlepsze rezultaty dla poszczególnych metod, uzyskano dla, zdefiniowanej przed analizą, maksymalnej liczby cech - 4000.

— Podczas, gdy większość wskaźników z powyższej tabeli posiada wartości większe od 85%, precyzja dla klasy pozytywnej oraz czułość dla klasy negatywnej, obliczone podczas analizy wydźwięku na podstawie kolokacji (COL_{lin}, COL_{dep}), znajdują się w przedziale 72% - 79%. To, że jest to akurat taka para współczynników, można łatwo uzasadnić na podstawie definicji przedstawionej w rozdziale 4 (*false positive* dla klasy pozytywnej jest tym samym co *false negative* dla klasy negatywnej, stąd istnieje pewnego rodzaju korelacja). Takie wartości oznaczają, że pewne elementy w zbiorze, ze wzorcową (rzeczywistą) klasą negatywną, otrzymały od klasyfikatora klasę pozytywną. Innymi słowy: klasyfikator zbyt ”hojnie” nadawał obserwacjom klasę pozytywną.

5.6.2. Informatywność cech

Poniższe tabele przedstawiają cechy, które według klasyfikatora, niosły największą ilość informacji dotyczącej wydźwięku. Informatywność danej cechy f, która przyjmuje wartość v można zdefiniować jako:

$$\textit{informatywność} = \frac{P(f = v \mid c_1)}{P(f = v \mid c_2)} \tag{5.1}$$

gdzie c_1 oraz c_2 to zmienne przyjmujące wartości ze zbioru zdefiniowanych klas: (pozytywna, negatywna). Inaczej mówiąc: szukamy takich cech i ich wartości, których występowanie dla jednej z klas jest znacznie bardziej prawdopodobne od drugiej. Przykładowo, zapis:

cecha = contains(piękny), wartość = true, klasa=pozytywna, informatywność=50:1

oznacza, że jeśli tekst zawiera słowo *piękny*, to prawdopodobieństwo przynależności opinii do klasy pozytywnej jest 50 - krotnie większe niż do klasy negatywnej.

Wylistowano cechy dla metod obecnych w tabeli 5.1, czyli dla wariantów, które uzyskały najlepsze wyniki. Dla każdej z nich ukazano cechy (domyślną wartością cechy jest *true* - dane słowo występuje w tekście), klasy na jakie wskazują (c_1 z licznika w równaniu 5.1) oraz wyliczoną informatywność.

Cecha	***Klasa***	***Informatywność***
contains(beznadzieja)	negatywna	40.1 : 1
contains(bełkot)	negatywna	38.8 : 1
contains(strata)	negatywna	38.0 : 1
contains(magiczna)	pozytywna	36.5 : 1

Tabela 5.2. Najbardziej informatywne cechy dla BOW_{raw}

Cecha	***Klasa***	***Informatywność***
contains(żenujący)	negatywna	45.1 : 1
contains(niebanalny)	pozytywna	31.7 : 1
contains(przygryzać)	negatywna	30.1 : 1
contains(płaski)	negatywna	26.1 : 1

Tabela 5.3. Najbardziej informatywne cechy dla BOW_{lem}

Cecha	***Klasa***	***Informatywność***
contains(zalecać wszyscy)	pozytywna	66.2 : 1
contains(zły książka)	negatywna	63.8 : 1
contains(szkoda czas)	negatywna	63.8 : 1
contains(zdecydowanie zalecać)	pozytywna	33.9 : 1

Tabela 5.4. Najbardziej informatywne cechy dla COL_{lin}

Cecha	***Klasa***	***Informatywność***
contains(polecać nie neg)	negatywna	52 : 1
contains(słaby bardzo adjunct)	negatywna	48.7 : 1
contains(polecać gorąco adjunct)	pozytywna	40.7 : 1
contains(książka wspaniały adjunct)	pozytywna	31.2 : 1

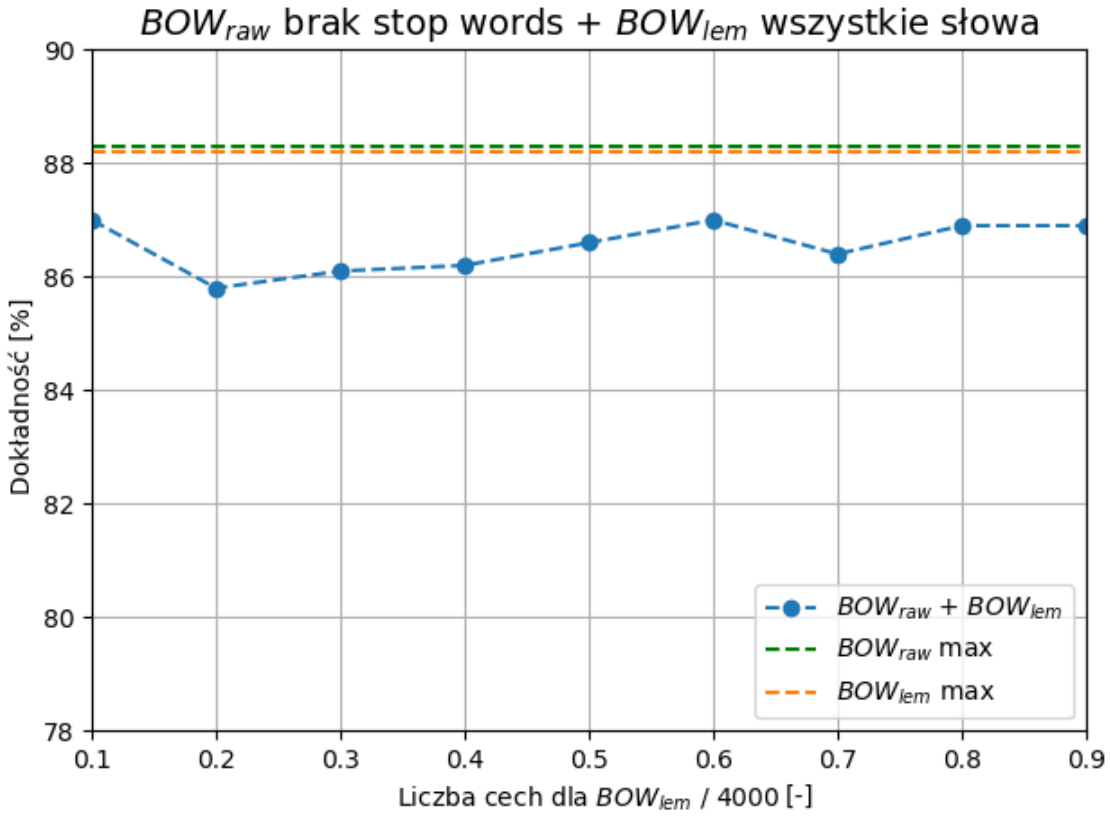
Tabela 5.5. Najbardziej informatywne cechy dla COL_{dep}

Dla BOW_{raw} oraz BOW_{lem} (tabela 5.2, tabela 5.3) dominują cechy, będące informacją o występowaniu przymiotników. Z dwóch dostępnych klas, ”łatwiejszą” do wytypowania okazuje się klasa negatywna. Wynika to zapewne z charakterystyki słownictwa wykorzystanego do sformułowania opinii o takim wydźwięku. Dla klasy negatywnej są to zazwyczaj wyrazy o dosadnym i jednoznacznym wydźwięku np. *beznadzieja* czy *żenujący*, które ciężko byłoby przypisać do opinii pozytywnej. Można przypuszczać, że zasób słownictwa używanego do wyrażenia zdania o wydźwięku negatywnym, jest mniejszy od zbioru słów opisującego tekst o nacechowaniu pozytywnym.

W przypadku kolokacji (tabela 5.4, tabela 5.5), cechy stają się bardziej zróżnicowane, a co za tym idzie, żadna z klas nie dominuje drugiej. Można jednak zauważyć, że jedno słowo (i jego pochodne) się wyróżnia - *polecać/zalecać*. Dla COL_{lin} *zalecać*, wraz z dodatkowym wyrazem w parze, najczęściej określają opinie pozytywne. Ciekawy przypadek występuje dla COL_{dep} - *polecać* występuje dwukrotnie, ale w dwóch różnych kontekstach. Pierwsza cecha *contains(polecać gorąco adjunct)* pomaga w identyfikacji tekstów o wydźwięku pozytywnym, natomiast druga cecha *contains(polecać nie neg)*, jest dobrym wskaźnikiem na opinie o charakterze negatywnym. Co ciekawe (i zapewne zgodne z intuicją), cechą, o największej wartości prawdopodobieństwa współwystępowania z klasą negatywną, jest obecność w tekście kolokacji, wyrażającej relację negacji między wyrazami.

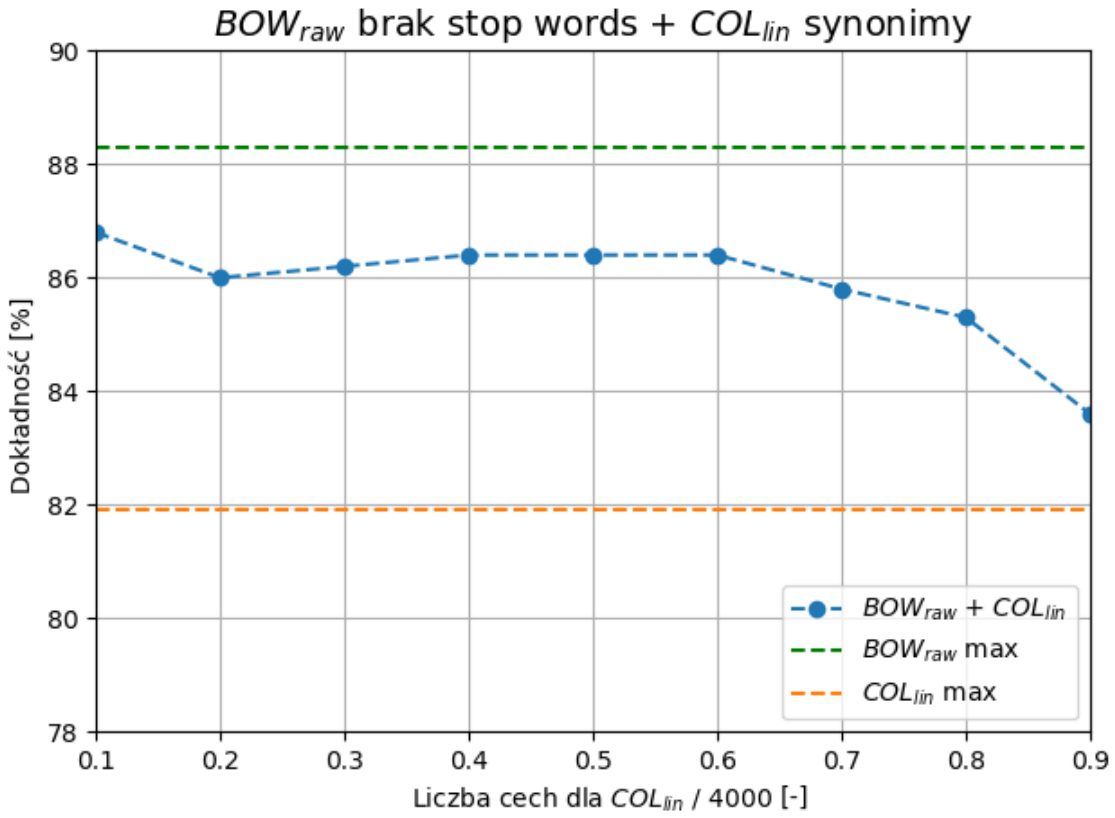
5.6.3. Analiza łączona

Wyniki dla łączonej analizy przedstawiono za pomocą wykresów, dla każdej z par osobno. Dla czterech metod, stosując strategię ”każdy z każdym”, przeprowadzono łącznie sześć niezależnych analiz. Łączny, nieprzekraczalny rozmiar wektora cech jest równy 4000. Stosunek liczby cech dla obu metod do rozmiaru wektora, był kolejno pobierany z przedziału od 0,1 do 0,9 z krokiem równym 0,1. Takie postępowanie ma na celu sprawdzenie, cechy której metody mają większy wpływ na jakość analizy łączonej. Poziomymi, przerywanymi liniami oznaczono dokładność dla każdej parowanej metody, pochodzących z tabeli 5.1. Ułatwia to weryfikację, czy połączenie dwóch różnych metod poprawia dokładność analizy.



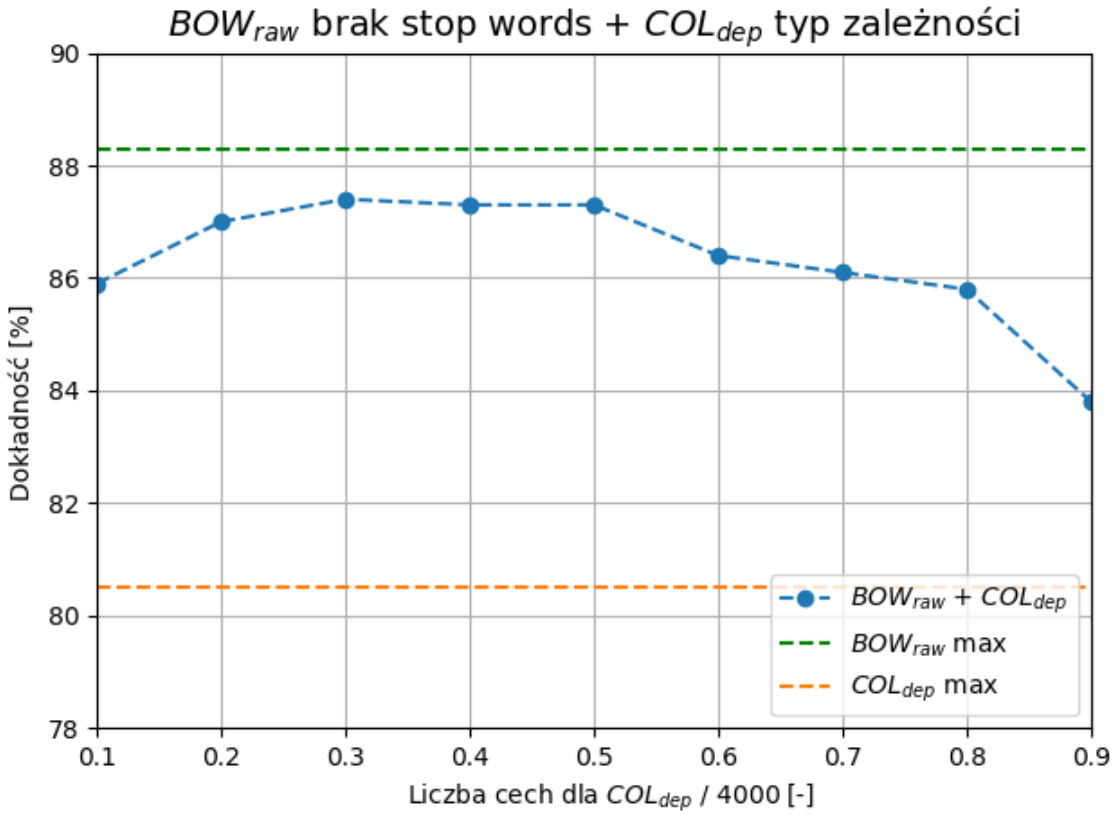
Rysunek 5.13. Wyniki dla metody łączonej - BOW_{raw} i BOW_{lem}

Wyniki (rysunek 5.13) dla BOW_{raw} i BOW_{lem} są bardzo zbliżone, przekraczają granicę 88%. Połączenie tych dwóch metod nie daje poprawy ”indywidualnych” osiągnięć. Dla każdego podziału liczby cech między metodami, wartość dokładności nie przekracza 87%.



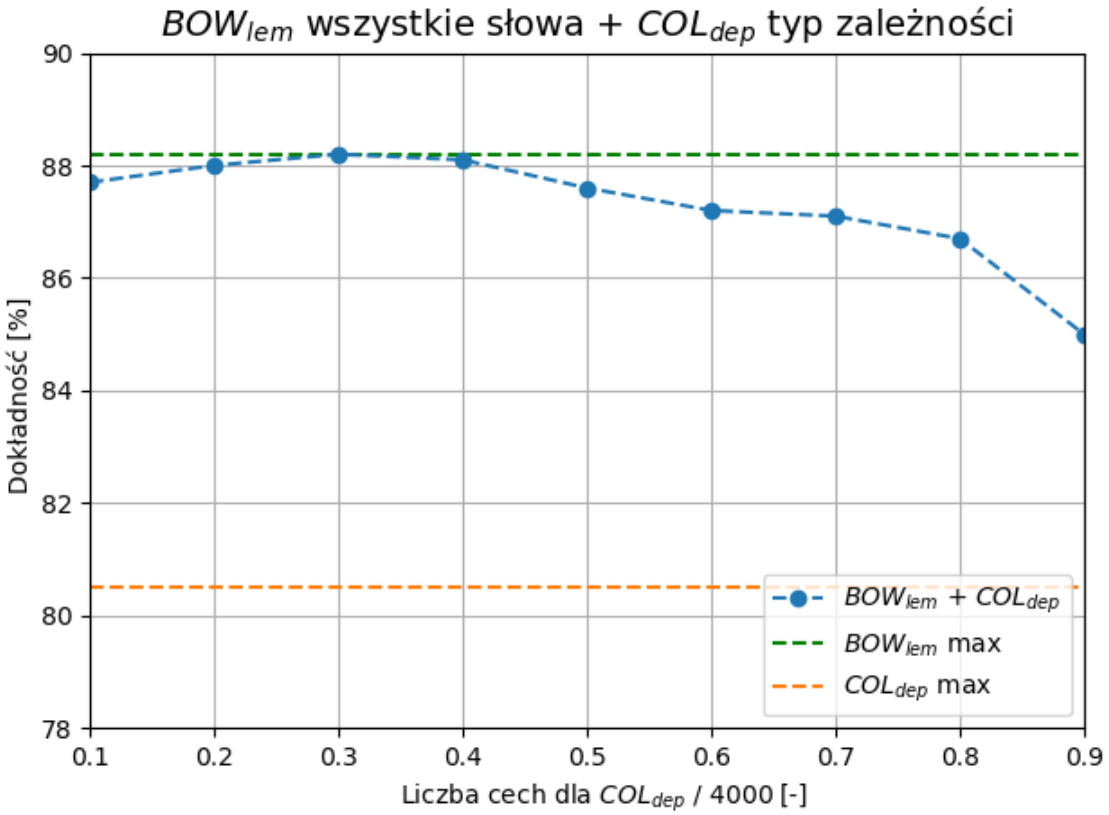
Rysunek 5.14. Wyniki dla metody łączonej - BOW_{raw} i COL_{lin}

Rozbieżność między wynikami (rysunek 5.14) dla BOW_{raw} i COL_{lin} jest zdecydowanie większa niż w poprzedniej parze. Na wykresie widoczny jest stopniowy spadek dokładności, wraz ze wzrostem udziału cech opartych na kolokacjach liniowych. Nie jest to zaskakujące, zważając na różnice w wynikach między metodami. Warto zauważyć, że rezultat łączny utrzymywany jest na przyzwoitym poziomie, nawet gdy COL_{lin} posiada przeważającą liczbę elementów w wektorze cech (dla 0.6 dokładność jest bliżej wyniku BOW_{raw} niż COL_{lin}).



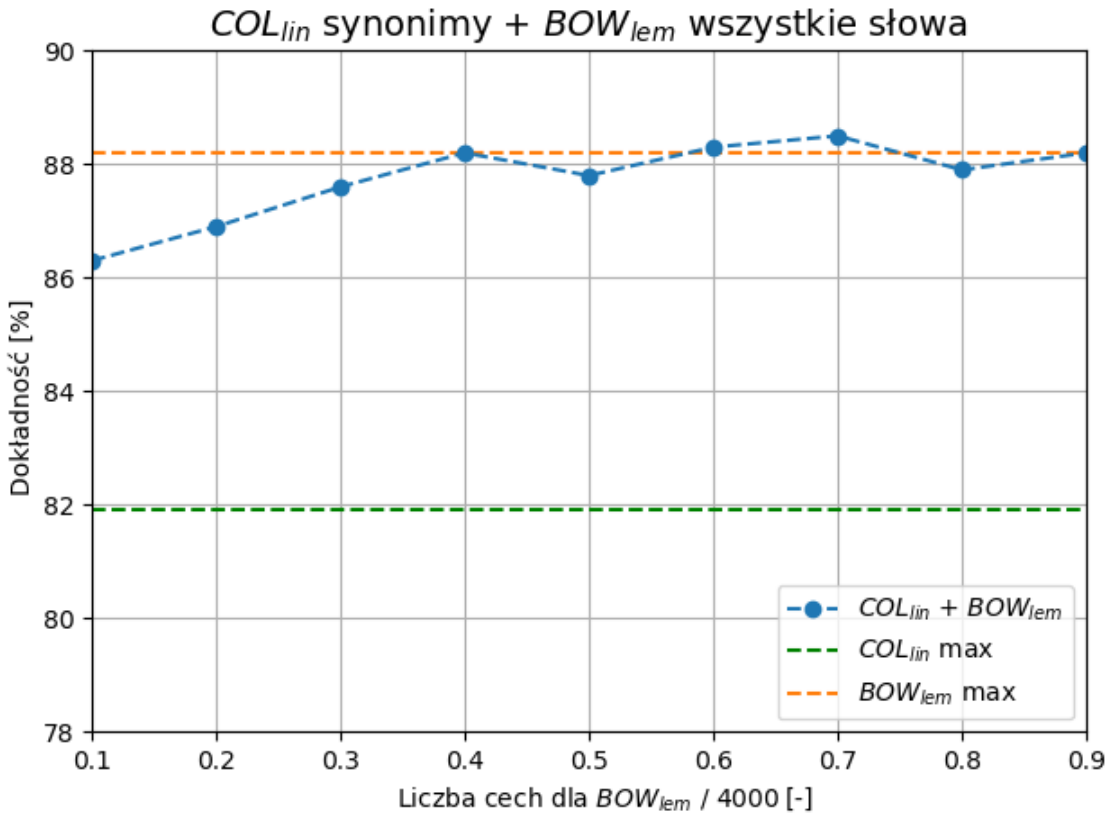
Rysunek 5.15. Wyniki dla metody łączonej - BOW_{raw} i COL_{dep}

Powyższy wykres (rysunek 5.15), ukazujący efekt połączenia metod BOW_{raw} i COL_{dep}, jest podobny do diagramu z poprzedniego przykładu. Jest widoczna tendencja spadkowa, która wzmacnia się w momencie uzyskania przez COL_{dep} zdecydowanej większości cech w wektorze. Istnieje jednak pewna różnica - zmniejszanie dokładności nie występuje od początkowych wartości podziału wektora. Najpierw zaobserwować można niewielki wzrost dokładności, która od połowy zaczyna powoli maleć.



Rysunek 5.16. Wyniki dla metody łączonej - BOW_{lem} i COL_{dep}

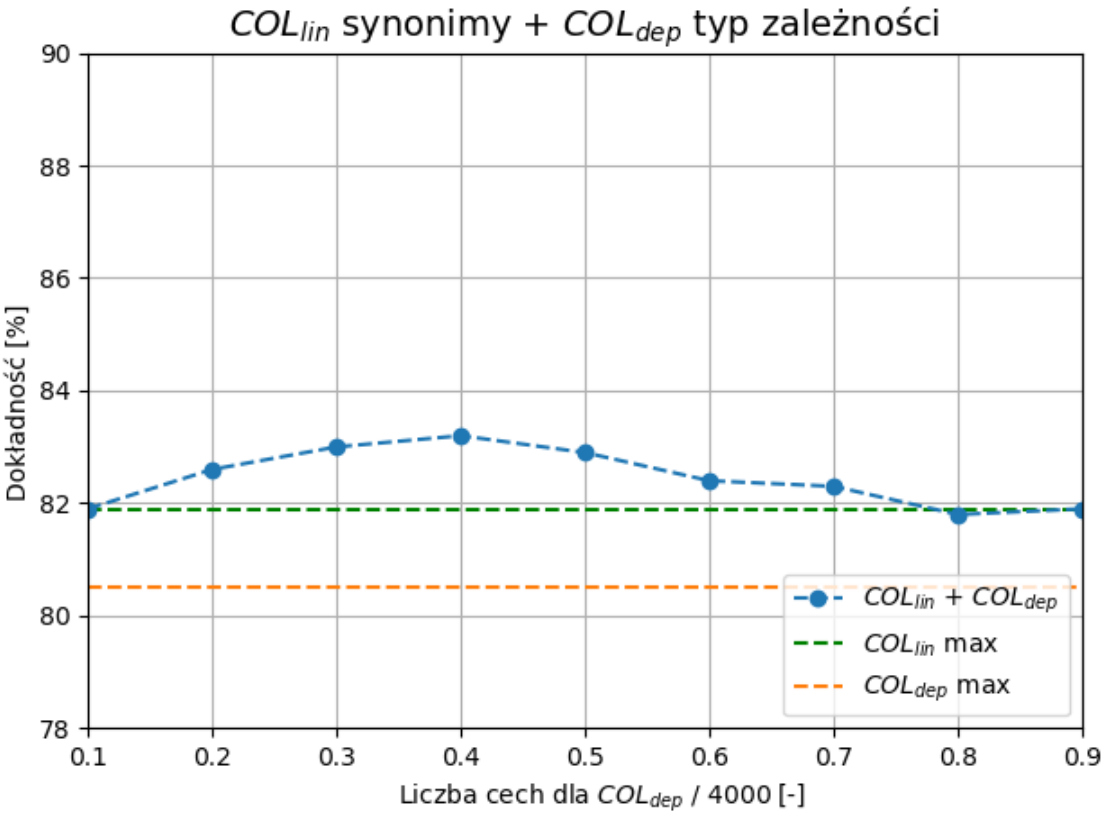
Wykres (rysunek 5.16) ukazujący relację między BOW_{lem} i COL_{dep} ma podobną charakterystykę do poprzedniego. Zasadniczą różnicą jest wyższa dokładność, niezależnie od stopnia podziału. Co ciekawe, dla 1200 (na osi 0,3) cech pochodzących od COL_{dep} wyrównano maksymalną wartość uzyskaną przez BOW_{lem} w izolacji.



Rysunek 5.17. Wyniki dla metody łączonej - BOW_{lem} z COL_{lin}

Połączenie BOW_{lem} z COL_{lin} jest do tej pory pierwszym, dla którego przekroczono wartość dokładności dla skuteczniejszej z metod dla analizowanej pary.

Zaczynając od początku wykresu (rysunek 5.17), wraz ze wzrostem liczby cech pochodzących od BOW_{lem}, dokładność powoli rośnie, wyrównując maksymalny wynik BOW_{lem}. Od momentu, gdy podział cech między metodami w wektorze jest równy, dokładność oscyluje wokół $BOW_{lem}\,max$, osiągając najlepszy, odnotowany podczas wszystkich badań, rezultat - 88,5%, dla 2800 cech od BOW_{lem} oraz 1200 od COL_{lin}.



Rysunek 5.18. Wyniki dla metody łączonej - COL_{dep} z COL_{lin}

Ostatnia para reprezentowana jest przez metody oparte na analizie wykorzystującej kolokacje - COL_{lin} oraz COL_{dep} (rysunek 5.18). Podobnie jak w przypadku pary BOW_{lem} i BOW_{raw}, oddzielne wyniki są do siebie zbliżone (jednak w porównaniu do pary reprezentantów *bag of words* wyraźnie słabsze).

Wyjątkowość tej analizy, polega na osiągniętych wynikach analizy łączonej. Poprawa dokładności z 81,9% do 83,2% może zostać uznana za znaczącą, porównując powyższe wyniki do poprzednich pięciu par, gdzie rezultaty uległy pogorszeniu, lub, w pojedynczych przypadkach, minimalnej poprawie.

5.7. Podsumowanie wyników

W rozdziale przedstawiono wyniki analizy wydźwięku opierającej się na naiwnym klasyfikatorze Bayesa, wykorzystującej dwa rodzaje cechy tekstu do jego reprezentacji - *bag of words* oraz kolokacje (liniowe, zależnościowe). Oba rodzaje podzielono na dwie metody, te zaś na specyficzne dla nich warianty. Dla każdej metody wyłoniono wariant, dla którego uzyskano najlepsze rezultaty. Wybrana czwórka została poddana analizie łączonej, której celem była weryfikacja, czy połączenie dwóch różnych typów cech tekstu poprawia jakość klasyfikacji.

Uzyskane wyniki mogą okazać się zaskakujące. Widoczna jest wyraźna przewaga metod *bag of words* nad analizą wykorzystującą kolokacje. Mogłoby się wydawać, że zastosowanie bardziej zaawansowanego sposobu ekstrakcji cech tekstu w postaci rozbioru zdania na drzewo zależnościowe, wpłynie pozytywnie na dokładność badań. Takie podejście niesie za sobą więcej informacji dotyczących składni i konstrukcji wyrażeń tekstowych w języku polskim. Okazuje się, że klasyfikatorowi, do skutecznej analizy wystarczą pojedyncze, nie posiadające kontekstu ani powiązań z innymi wyrazami słowa.

Co więcej, metody oparte na *bag of words* dają bardzo podobne wyniki. Zastosowanie lematyzacji, w celu ujednolicenia tekstu, nie podnosi jakości analizy wydźwięku. Zachowana forma fleksyjna wyrazów nie sprawia klasyfikatorowi problemów. Kolejną ważna kwestią jest filtrowanie tekstu poprzez usunięcie *stop words*. W większości przypadków taki zabieg poprawia dokładność, więc wykorzystując *bag of words* do analizy wydźwięku, warto wyczyścić tekst z wyrazów obecnych na liście *stop words*.

Jeśli chodzi o metody wykorzystujące kolokacje - osiągają zbliżone wyniki, aczkolwiek minimalnie lepsza okazuje się strategia ekstrakcji cech oparta na kolokacjach liniowych (bigramach). W przypadku obu metod warto zastosować zabieg zastępowania słów nieposiadających wydźwięku ich synonimami o nacechowaniu emocjonalnym.

Analiza łączona pokazuje, że mieszanie cech z różnych metod nie powoduje generalnej poprawy wyników. Tylko dla jednego przypadku udało się przekroczyć największą wartość uzyskaną dla analizy oddzielnej. Poprawę można także zaobserwować w przypadku połączenia cech obu metod dla kolokacji. Wniosek: jeśli do analizy wydźwięku wykorzystuje się tylko kolokacje, najskuteczniejszym podejściem jest wygenerowanie wektora cech, składającego się z elementów w postaci dwóch (liniowe, zależnościowe), a nie jednego typu kolokacji. Analiza skupiająca się na jednym typie jest mniej skuteczna.

6. Zakończenie

W dokumencie przedstawiono pracę dotyczącą analizy wydźwięku, wzbogaconą o informacje pochodzące z rozbioru zdań na drzewa zależnościowe. Przeanalizowano wpływ wykorzystania bardziej zaawansowanych sposobów ekstrakcji cech na dokładność analizy. Tak uzyskane wyniki porównano z klasycznymi metodami badania sentymentu, gdzie tekst reprezentowany jest za pomocą *bag of words*.

Z przeprowadzonych badań wynika, że eksperyment w postaci wykorzystania drzewa rozbioru do ekstrakcji cech, prowadzi do uzyskania gorszych rezultatów w porównaniu do standardowych metod *bag of words*.

Praca zawiera również opis wykorzystanych narzędzi do przetwarzania tekstu. Ich mnogość z jednej strony ułatwia analizę, a z drugiej powoduje problem dotyczący wyboru odpowiedniego tzn. stabilnego i niezawodnego narzędzia do danego zadania. Przed rozpoczęciem docelowych badań, warto jest dogłębnie przestudiować możliwości testowanego oprogramowania - ostateczna decyzja może mieć znaczący wpływ na wyniki analizy. W szczególności, należy być ostrożnym opierając się na dostępnych w sieci wersjach demonstracyjnych narzędzi. W przypadku zautomatyzowanej analizy na dużą skalę, dobrym pomysłem może okazać się uruchomienie, we własnym zakresie, aplikacji na dedykowanej do tego celu maszynie (oczywiście pod warunkiem, że autorzy oprogramowania wyrażają na to zgodę).

Praca nie jest pozbawiona pewnych niedociągnięć - istnieje kilka sposobów na jej usprawnienie i rozwoju w przyszłości. W związku z tym, że głównym celem pracy była analiza kolokacji, pierwszym krokiem mogłoby być wykorzystanie drugiego rodzaju drzewa rozbioru, którego opis pojawił się w dokumencie - drzewa składnikowego. Nieudaną próbę użycia narzędzia *Świgra 2* można ponowić, tym razem przygotowując dla aplikacji dedykowane środowisko uruchomieniowe, gdzie problemy wydajnościowe nie będą przeszkodą podczas analizy.

Kolejną możliwością usprawnienia analizy, jest manipulacja wartościami przyjętych parametrów. Przykładowo, dobrym pomysłem jest przeprowadzenie badań dla zwiększonego rozmiaru zbioru opinii (aktualnie 20000), czy zmodyfikowanego stosunku liczby elementów uczących do testujących (aktualnie - 85% : 15%). Warto zastanowić się również nad przesunięciem granicy, na podstawie której wyznaczane są klasy wzorcowe poszczególnych opinii ze zbioru treningowego. W pracy przyjęto, że jeśli ocena książki jest większa o 6, to wyraża opinię pozytywną, natomiast w przypadku wartości mniejszej od 4 - negatywną. Tego rodzaju modyfikacja sprawiłaby, że analiza skupiłaby się na opiniach wyrażających bardziej skrajne emocje. Można przypuszczać, że obecne w zbiorze elementy neutralne i powszechne zaburzają efektywność analizy.

Następnym pomysłem jest weryfikacja omawianych metod ekstrakcji cech tekstu, ale w oparciu o inny rodzaj klasyfikatora. W pracy analizę przeprowadzono wykorzystując naiwny klasyfikator Bayesa o rozkładzie Bernoulliego. Nic nie stoi na przeszkodzie, aby do badań użyć metodę bayesowską o innym rozkładzie.

Ostatecznie, praca spełnia założony na początku cel - weryfikuje wpływ, opartych na drzewie rozbioru, metod ekstrakcji cech na dokładność analizy wydźwięku. Podsumowaniem badań może być stwierdzenie, że najprostsza metoda okazuje się być lepszym wyborem niż ta, oparta na bardziej wyszukanych sposobach reprezentacji tekstu, w postaci par wyrazów wzbogaconych kontekstem gramatycznym.

Bibliografia

[1] A. Asmi, T. Ishaya. *Negation Identification and Calculation in Sentiment Analysis*. IMMM 2012 : The Second International Conference on Advances in Information Mining and Management, 2012.

[2] M. Dadvar, F. de Jong, C. Hauff. *Scope of negation detection in sentiment analysis*. Dutch-Belgian Information Retrieval Workshop, 2011.

[3] K. T. Durant, M. D. Smith. *Mining Sentiment Classification from Political Web Logs*. Proceedings of Workshop on Web Mining and Web Usage Analysis of the 12 th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2006.

[4] S. Eyheramendy, D. D. Lewis, D. Madigan. *On the Naive Bayes Model for Text Categorization*. 2003.

[5] A. Giachanou. *Like It or Not: A Survey of Twitter Sentiment Analysis Methods*. Journal ACM Computing Surveys, v.49, 2016.

[6] V. Hatzivassiloglou, K. R. McKeown. *Predicting the semantic orientation of adjectives*. Proceeding ACL '98/EACL '98 Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics and Eighth Conference of the European Chapter of the Association for Computational Linguistics, s. 174-181, 1997.

[7] D. Jurafsky, J. H. Martin. *Speech and Language Processing An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*. 2018.

[8] K. Krawiec, J. Stefanowski. *Sieci neuronowe i uczenie maszynowe*. Wydawnictwo Politechniki Poznańskiej, 2003.

[9] B.Liu. *Sentiment Analysis and Opinion Mining*. Morgan & Claypool Publishers, 2012.

[10] B.Liu. *Sentiment Analysis and Subjectivity*. Handbook of Natural Language Processing, 2010.

[11] P. Melville, W. Gryc, R.D. Lawrence. *Sentiment analysis of blogs by combining lexical knowledge with text classification*. Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining, s. 1275-1284, 2009.

[12] M. S. Neethu, R.Rajasree. *Sentiment analysis in twitter using machine learning techniques*. 2013 Fourth International Conference on Computing, Communications and Networking Technologies (ICCCNT), 2013.

[13] M. Ogrodniczuk, M. Lenart. *Web Service integration platform for Polish linguistic resources*. Proceedings of the Eighth International Conference on Language Resources and Evaluation, LREC 2012, s. 1164–1168, 2012.

[14] J. Opałka, W. Abramowicz, W. Sokołowska, T. Hossa. *Automatyczna analiza wydźwięku opinii o operatorach energetycznych jako element wsparcia podejmowanych decyzji*. Wydawnictwo Uniwersytetu Ekonomicznego w Katowicach, v. 243, s. 257-273, 2015.

[15] B. Pang, L. Lee. *Opinion mining and sentiment analysis*. Foundations and Trends in Information Retrieval, v. 2, s. 1-135, 2008.

[16] B. Pang, L. Lee, S .Vaithyanathan. *Thumbs up? Sentiment classification using machine learning techniques*. Proceeding EMNLP '02 Proceedings of the ACL-02 conference on Empirical methods in natural language processing, v. 10, s. 79-86, 2002.

[17] M. Taboada, J. Brooke, M. Tofiloski, K. Voll, M. Stede. *Lexicon-Based Methods forSentiment Analysis*. Computational Linguistics, s. 267-307, 2011.

[18] P. D. Turney. *Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews*. Proceeding ACL '02 Proceedings of the 40th Annual Meeting on Association for Computational Linguistics, s. 417-424, 2002.

[19] H. Thakkar, D. Patel. *Approaches for Sentiment Analysis on Twitter: A State-of-Art study*, 2015.

[20] M. Woliński. *Automatyczna analiza składnikowa języka polskiego*. Wydawnictwa Uniwersytetu Warszawskiego, 2019.

[21] M. Woliński. *Morfeusz reloaded*. Proceedings of the Ninth International Conference on Language Resources and Evaluation, s. 1106–1111, 2014.

[22] K. Wójcik, J. Tuchowski. *Wykorzystanie metody opartej na wzorcach w automatycznej analizie opinii konsumenckich*. Prace Naukowe Uniwersytetu Ekonomicznego we Wrocławiu, nr 385, Taksonomia 25: Klasyfikacja i analiza danych – teoria i zastosowania, s. 314–324, 2015.

[23] U. Yasavur, J. Travieso, C. Lisetti, N. Rishe. *Sentiment Analysis Using Dependency Trees and Named-Entities*. Proceedings of the Twenty-Seventh International Florida Artificial Intelligence Research Society Conference, 2014.

[24] H. Zou, X. Tang, B. Xie, B. Liu. *Sentiment Classification Using Machine Learning Techniques with Syntax Features*. International Conference on Computational Science and Computational Intelligence (CSCI), 2015.

[25] *Clarin-PL* [online][dostęp: 31.08.2019] http://clarin-pl.eu/pl/strona-glowna.

[26] *NLTK* [online][dostęp: 31.08.2019] https://www.nltk.org.

[27] *NLTK Book* [online][dostęp: 31.08.2019] https://www.nltk.org/book.

[28] *Scala Scraper* [online][dostęp: 31.08.2019] https://github.com/ruippeixotog/scala-scraper.

[29] *Słowosieć* [online][dostęp: 31.08.2019] http://plwordnet.pwr.wroc.pl/wordnet.

[30] *WCRFT* [online][dostęp: 31.08.2019] http://nlp.pwr.wroc.pl/redmine/projects/wcrft/wiki.

[31] *Stop words* [online][dostęp: 31.08.2019] https://pl.wikipedia.org/wiki/Wikipedia:Stopwords.

Spis rysunków

Spis tabel