
SZKOŁA GŁÓWNA HANDLOWA W WARSZAWIE
Kolegium Analiz Ekonomicznych

Podyplomowe Studia Data Science w biznesie

# Poprawa metryk jakości modelu predykcyjnego poprzez włączenie danych pozyskanych z nieustrukturyzowanego tekstu za pomocą metodologii NLP

Imię i nazwisko autora:
Kamil Dąbrowski
Nr albumu: 128994

Praca końcowa napisana pod kierunkiem
mgr. inż. Marcina Zadrogi

Warszawa 2024

# Spis treści

# Wprowadzenie

W nowoczesnej rzeczywistości korporacyjnej oczywistym, nie wymagającym uzasadnienia, sposobem postępowania jest podejmowanie decyzji na podstawie gromadzonych i przetwarzanych przez organizację danych. Korzyści z tak prowadzonego procesu decyzyjnego zostały jednoznacznie wykazane i nie są już w dużych podmiotach gospodarczych poddawane w wątpliwość. Dowiedziona została także pozytywna zależność między stopniem oparcia działania na danych a wydajnością, rentownością aktywów czy też wartością rynkową przedsiębiorstwa (Provost i Fawcett, 2023, s. 29). Firmy wykazują większą niż w przeszłości uwagę i dbałość o obszar danych, co widać przykładowo w pojawieniu się odnoszącego się do tej kwestii nazewnictwa stanowisk (Data Engineer, Data Scientist, Chief Data Officer), elementów struktury korporacyjnej oraz pojęć w ramach różnych metodologii zarządzania (np. Data Tribes lub Guilds w ramach Agile). Powszechnie, nie tylko w branżach związanych nowymi technologiami, powtarzane jest hasło, iż dane są jednym z najważniejszych i najcenniejszych zasobów organizacji. Ku analityce danych, mającej na celu ich wykorzystanie celem osiągnięcia lub utrzymania przewagi strategicznej, skłaniają się firmy z tradycyjnych branż. Jednocześnie funkcjonowanie wielu założonych w XXI w. organizacji, jak Facebook czy Twitter (obecnie X), jest w całości oparte na zdobywanych lub wytwarzanych zasobach danych (Provost i Fawcett, 2023, s. 35).

Wraz ze wzrostem zainteresowania danymi organizacje zaadaptowały różne rozwiązania z zakresu nauki o danych, zarówno w relacjach z klientami, jak i w wewnętrznych procesach. Stosowanie nadzorowanych i nienadzorowanych metod uczenia maszynowego wykracza już poza pierwotne skojarzenie, czyli zdobycz nauk ścisłych udomowioną przez świat prywatnego biznesu. Znajdują one coraz powszechniejsze wykorzystanie także w administracji publicznej. Tym samym mają coraz większy wpływ na codzienne życie obywateli. Skutkiem jest większa obecność tej tematyki w sferze publicznej, często w negatywnym kontekście. Przykładowo, afera związana z pełnym błędów wdrożeniem i niewłaściwym nadzorem nad algorytmami predykcyjnymi w systemie opieki społecznej stała się jedna z przyczyn upadku rządu premiera Niderlandów Marka Ruttego (Litorowicz, 2024).

Upowszechnienie narzędzi Data Science, często w szerokim kontekście określanym jako Sztuczna Inteligencji (AI), wpłynęło na zainteresowanie tym obszarem także ustawodawców. Spektakularnym przykładem jest EU AI ACT, czyli prawo kompleksowo regulujące tę materię na poziomie Unii Europejskiej, którego wdrożenie zostało już przegłosowane przez Parlament Europejski. Według deklaracji dedykowane prawo ma zarówno umożliwić organizacjom korzystanie z dobrodziejstw technologii określanych jako AI, w zgodności z długofalowymi celami polityki Unii, jak również chronić europejskich obywateli przez możliwymi zagrożeniami (European Parliament, 2023). Warto zwrócić uwagę, iż w regulacji tej przyjęto podejście do poszczególnych zagadnień i ich zastosowań zależne od ryzyka. Wymienione enumeratywnie strefy zostały zdefiniowane jako "high-risk" i będą podlegać szczególnym restrykcjom, w tym także konieczności weryfikacji przez człowieka wyników modeli oraz odpowiedniego stopnia ich interpretowalności. Dotyczyć to będzie także sektora bankowego (credit scoring, jako narzędzie mające wpływ na strefę dostępu do usług finansowych), ubezpieczeniowego (analogicznie do credit scoringu mogą być traktowane narzędzia oceny ryzyka ubezpieczeniowego). Również w Polsce mamy już pierwsze sygnały zainteresowania władz tematyką AI i przesłanki, że obszar ten może doczekać się w przyszłości dedykowanych regulacji prawnych. Przykładowo, w niedawnej wypowiedzi Ministra Pracy i Polityki Społecznej potwierdziła zainteresowanie swojego Resortu algorytmami wykorzystywanymi przez pracodawców do celów rekrutacji czy ewaluacji wyników pracy osób zatrudnionych (PAP Biznes, 2024). Regulacje narzucające ścisłą kontrolę ludzką nad wynikami algorytmów mogą potencjalnie kierować przedsiębiorstwa w stronę narzędzi Data Science charakteryzujących się większą interpretowalnością i mniejszą skłonnością do dopasowywania się do danych, jak modele nieoparte na sieciach neuronowych. Potencjalnym skutkiem może także być zmiana sposobu myślenia o jakości stosowanych rozwiązań w firmach działających w obszarach zdefiniowanych przez UE jako "high-risk". Dbałość o ich odpowiedni poziom będzie już nie tylko wewnętrzną sprawą organizacji, ale także jej obowiązkiem w świetle roli, którą pełnią w społeczeństwie np. jako dostawcy usług finansowych.

Wszystko powyższe skłania do refleksji, że jakość stosowanych modeli predykcyjnych, może, niezależnie czy dotyczy sektora prywatnego czy administracji publicznej, w znaczący sposób oddziaływać nie tylko na samą organizację, ale i jej interesariuszy. Efekt skali może skutkować przełożeniem nawet niewielkiej zmiany w jednej lub więcej metryk jakościowych modelu na znaczące zmiany w procesach firmy, efektywności wykorzystania zasobów, relacjach z klientami. W konsekwencji, w kontekście finansowym, odpowiednie przygotowanie, wdrożenie i utrzymanie modeli, przekłada się na poziomy przychodów i kosztów. Co więcej, niewłaściwe starania w tym zakresie będą mogły narazić przedsiębiorstwo także na konsekwencje ze strony instytucji nadzoru.

Jednocześnie ostatnie lata pozwoliły nam być świadkami ogromnego postępu w zakresie rozwiązań sztucznej inteligencji opartych na przetwarzaniu języka naturalnego (w dalszej części pracy najczęściej używany będzie powszechnie przyjęty skrót NLP – od Natural Language Processing). Dotyczy to bardzo

zaawansowanych narzędzi wykorzystujących uczenie głębokie i modelowanie generatywne. Dużo uwagi zyskał w szczególności ChatGPT, czyli rozbudowany chatbot i asystent wirtualny. Obecnie byłby on w stanie napisać np. streszczenie niniejszej pracy nieodróżnialne od wykonanego przez człowieka. Dobrym przykładem jest także tłumacz, będący częścią Galaxy AI - rozwiązań umieszczanych w telefonach marki Samsung. Aplikacja ta pozwala na transkrypcję w czasie rzeczywistym tekstu wprowadzanego zarówno za pomocą klawiatury, jak i głosu, bezpośrednio na syntetyczną mowę. W ostatnich miesiącach mieliśmy także wysyp informacji dotyczących książek wygenerowanych przy pomocy sztucznej inteligencji i związane z tym protesty autorów (Tapper, 2023).

Przedmiotem zainteresowania pracy nie będą jednak rozbudowane sieci neuronowe. Poszczególne narzędzia z zakresu NLP można bowiem, wybierając je według potrzeb, wykorzystać do celów uzupełnienia modeli predykcyjnych opartych na regresji liniowej czy drzewach decyzyjnych o nowe informacje na temat modelowanych zjawisk. Zgodnie z zasadą Brzytwy Ockhama nie zawsze bowiem rozwiązanie bardziej zaawansowane będzie tym pożądanym. Aplikacje NLP powiązane z uczeniem głębokim mogą być obarczone wadami, takimi jak choćby: niska interpretowalność, nadmierna adaptacja dziedzinowa, trudność w uczeniu na nielicznych przypadkach czy też znaczne koszty wdrożenia (Vajjala i in, 2023, s. 51-52). Od stosowanych w organizacjach modeli predykcyjnych niekoniecznie oczekujemy też zwrócenia najlepszego możliwego wyniku. Często ich wyniki nie są centralną częścią danego procesu w organizacji. Wskazania modelu mogą być częścią składową pracy danej komórki, podlegać dalszej analizie lub przetworzeniu. Przykładowo w branży ubezpieczeniowej modele antyfraudowe są wykorzystywane do wstępnego filtrowania zgłoszonych roszczeń i są jednym ze źródeł, obok np. oflagowania przez pracowników merytorycznych, kierowania spraw do dalszej analizy. Wraz z codzienną pracą z modelem, w której nacisk jest kładziony na interpretowalność wyniku, pracownicy mogą więc oszczędzać czas i rozwijać swoje umiejętności analityczne. Wykorzystanie mniej zaawansowanych rozwiązań, np. heurystyk, umożliwia także analitykom danych eksperymentowanie i szukanie nieodkrytych jeszcze związków między posiadanymi danych lub ich nowych zastosowań.

# Rozdział I: Język i tekst w kontekście pozyskiwania informacji na temat modelowanych zjawisk

Na potrzeby poniższych rozważań słowa "język" i "tekst" będą stosowane niejako zamiennie. Oczywiście nie są to jednak pojęcia tożsame. Słownikowo, "język" jest to ustrukturyzowana forma komunikacji, wykorzystująca zależności i kombinacje elementów składowych (Vajjala i in, 2023, s. 333). Dziedziną nauki, której polem zainteresowania są języki, jest lingwistyka. Definicji słowa "tekst" jest wiele, najczęściej jednak podkreślają one, że z "tekstem" mamy do czynienia w przypadku graficznie utrwalonego ciągu znaków składających się na pewną całość znaczeniową (Wikipedia, 2024). W języku informatyki funkcjonuje pojęcie "tekstowego typu danych" (z ang. "string"), które określa tekst w formie zmiennej lub stałej przechowywanej w pamięci komputerowej. (Wikipedia, 2024). Pewne atrybuty mające wpływ na pole zainteresowania niniejszej pracy będą miały zastosowanie bardziej do języka jako sposobu komunikacji. Niektóre z kolei będą dotyczyć tekstu jako formy zapisu języka. Jeśli dane wprowadzane są przy pomocy głosu są z reguły zamieniane, właśnie za pomocą narzędzi NLP, na tekst. Niezależnie od zastosowanego uproszczenia, przed omówieniem NLP jako dziedziny informatyki warto wskazać, jakie atrybuty języka i tekstu będą miały wpływ na ich przetwarzanie przez program komputerowy. Należy przy tym pamiętać, że pojęcie NLP jest szerokie i obejmuje dwustronną komunikację na linii człowiek/komputer oraz różne jej formy. Omówione zostaną także podstawowe pojęcia z zakresu nauk o języku, które będą mają znaczenie dla praktycznego zastosowania NLP.

## 1.1 Tekst jako potencjalne źródło istotnych i dostępnych informacji

Literatura jest zgodna, że jednym ze sposobów poprawy modeli predykcyjnych jest rozszerzenie ich o nowe dane, mogące nieść istotne informacje dotyczące modelowanego zjawiska. Przyjmijmy bowiem, że odpowiedź na zadane pytanie można przynajmniej w pewnym stopniu przewidzieć za pomocą algorytmu regresji lub klasyfikacji. W praktyce musi to oznaczać, że istnieją dane istotne dla konkretnego zagadnienia, których pozyskanie oraz odpowiednie przetworzenie może zwiększyć jakość modelu. Właśnie za tą możliwością powinna podążyć organizacja chcąca maksymalizować efekt wykorzystania algorytmu. W tym celu konieczne może być wyjście poza wcześniejsze ramy, w tym skierowanie uwagi w stronę zmiennych nieustrukturyzowanych. Takim mianem określane są wszelkie dane, które w naturalny sposób nie są układane w sposób stabelaryzowany, czyli posiadający rzędy obserwacji i kolumny opisujących je wartości cech. Oczywiście zarówno tekst, jak i obraz czy dźwięk są możliwe do przedstawienia w taki sposób. W odróżnieniu jednak od danych ustrukturyzowanych pojedyncza wartość na przecięciu rzędu i kolumny nie niesie żadnej informacji (Foster, 2021, s. 41). Dla przykładu fakt, że dwudziestą piątą literą ciągu znaków

jest “p” nie przekazuje żadnej wiedzy na temat tematyki tekstu czy jego nastawienia emocjonalnego. Oznacza to konieczność szukania użytecznego sposobu reprezentacji danych nieustrukturyzowanych na potrzeby ich eksploracji pod kątem rozwiązania konkretnego problemu.

W takie spojrzenie wpisują się właśnie dane tekstowe. Przede wszystkim są one bowiem powszechne i jednocześnie często niewykorzystane. Są też bezpośrednio powiązane z odczuciami, postawami i potencjalnymi decyzjami klientów, do których nie zawsze mogą prowadzić inne gromadzone zmienne. Charakter dużej części tradycyjnych zmiennych wydaje się bowiem zbyt ograniczony, żeby oddać pełnię ludzkiego sposobu rozumowania, także w sprawach niezwykle istotnych dla biznesu jak np. decyzje zakupowe.

Tradycyjnie tekst, nawet umieszczony w systemach informatycznych, był wykorzystywany do komunikacji między ludźmi. Jednocześnie w dobie popularności serwisów internetowych określanych jako Web 2.0 powszechne stało się gromadzenie dużej ilości danych, powstających w związku z interakcyjnym charakterem sieci, na styku relacji między klientem i firmą lub między użytkownikami jednego serwisu (Provost i Fawcett, 2023, s. 32). Jako oczywistość przyjmuje się obecnie, nawet jeśli charakter danego serwisu nie dotyczy bezpośrednio interakcji międzyludzkich, możliwość pozostawienia przez użytkownika swobodnej opinii, recenzji produktu czy rekomendacji dla innych. Jest to gotowy rezerwuar informacji do pozyskania przez organizację chcącą faktycznie “wsłuchać” się w głosy klientów w poszukiwaniu potencjalnie istotnych dla swoich celów informacji.

Zarówno w potocznych rozmowach, jak i na gruncie nauk o zarządzaniu często przypisuje się organizacjom czysto ludzkie cechy. Oznacza to także ich zdolność do popełniania typowych błędów w rozumowaniu, których nie pozwala uniknąć nawet rozbudowany system decyzyjny. Jednym z dobrze zdefiniowanych pułapek myślenia jest wyciąganie pochopnych wniosków na podstawie ograniczonych danych, zwane także WYSIATI - “what you see is all there is” (Kahneman, 2012, s. 117). Próba pozyskania nowych informacji z tekstu, a także innych nieustrukturyzowanych źródeł, w celu poprawy jakości stosowanych algorytmów jest dobrym sposobem na przezwyciężenie potencjalnych niebezpieczeństw związanych z nadmiernym poleganiem na niedopasowanym modelu. Z tego ujęcia także wpisuje się w kontekst rozważań niniejszej pracy.

Zasadnym jest więc podejmowanie prób wykorzystania na potrzeby algorytmów gromadzonych już danych tekstowych, które były dotychczas procesowane bez pomocy narzędzi AI. Autor pozwoli sobie podać przykład z własnej kariery zawodowej w branży ubezpieczeniowej. W czasie jego kilkunastoletniej pracy w likwidacji szkód komunikacyjnych, w tym głównie przeciwdziałaniu przestępczości ubezpieczeniowej, tekstowy opis zawarty w zgłoszeniu szkody zmienił swój charakter i wykorzystanie. Początkowo był najczęściej odręcznym zapisem na druku zgłoszenia szkody znajdującym się fizycznie w teczce szkodowej. Następnie spopularyzowano telefoniczne zgłoszenia i systemy informatyczne do likwidacji szkód. W efekcie

opis pojawił się jako dedykowane pole w interfejsie, z początku najczęściej uzupełniane przez konsultantów przyjmujących zgłoszenia. Modele wykorzystywane do predykcji prawdopodobieństwa wyłudzenia były wówczas oparte na scoringu punktowym i nie wykorzystywały tekstu. Obecnie szkody są najczęściej zgłaszane bezpośrednio przez klientów lub pełnomocników za pomocą formularza umieszczonego na stronie internetowej. W ten sposób zaawansowany technologicznie ubezpieczyciel dysponuje, tuż po zgłoszeniu roszczeń, opisem sformułowanym przez klienta w formie gotowej zmiennej tekstowej. W efekcie treść opisu, którą doświadczony likwidator szkód określiłby jako "mogącą wskazywać na wysokie prawdopodobieństwo próby wyłudzenia odszkodowania" można poddać próbie wykorzystania np. do celów predykcji. W firmach ubezpieczeniowych powstają już modele, nie tylko z zakresu przeciwdziałania wyłudzeniom, których uzupełnieniem lub wręcz podstawą są informacje, które tradycyjnie podlegały jedynie manualnej weryfikacji specjalistów. Przykład ten pokazuje swoistą ewolucję w sposobie wykorzystania jakiej mogą podlegać dane tekstowe.

## 1.2 Tekst jako wyzwanie dla Data Scientists

Jak wspomniano we wcześniejszej części pracy, tekst jest najczęściej traktowany jako dane nieustrukturyzowane. Nie ma on bowiem formy dającej się w intuicyjny sposób umieścić w tabeli rekordów z przewidywalnymi polami, przechowującymi jednoznaczne zmienne. Jednocześnie jednak tekst ma bez wątpienia charakter uporządkowany. Ład ten, który umożliwia nam zrozumienie treści, jest jednak przeznaczony do komunikacji międzyludzkiej. Stąd trudność jego przełożenia na język maszynowy (Provost i Fawcett, 2023, s. 244).

Ludzie nie potrzebują komputerów, żeby napotykać na szereg problemów komunikacyjnych. Spontanicznie napisany tekst będzie często zawierał błędy różnego typu, utrudniające, czasem wręcz uniemożliwiające, jego zrozumienie. Nawet poprawnie napisana wypowiedź może, ze względu na wiele czynników, być trudna do interpretacji. Znacznie bardziej niż w przypadku innych rodzajów danych istotny jest kontekst, który może całkowicie odwrócić znaczenie wypowiedzi (Provost i Fawcett, 2023, s. 245). W tradycyjnym ujęciu trudna do wyobrażenia wydaje się sytuacja, w której identyczna wartość zmiennej w danym polu tabeli rekordów może oznaczać de facto zupełnie różne wartości w zależności od innych zmiennych znajdujących się nie tylko w jej sąsiedztwie, ale także poza zbiorem danych. Do powyższego należy dodać, że każdy język posiada swoją własną strukturę lingwistyczną i zasady obejmujące słowotwórstwo, fleksję czy składnię. Fakt ten powiększa wątpliwości co do użyteczności tekstu w kontekście modelowania. Co więcej, a ramach jednego języka narodowego funkcjonują różne odmiany np. dialekty, slangi czy żargony. Szczegółowy opis tych pojęć wykracza dalece poza ramy niniejszej pracy. Należy sobie

jednak uświadomić, że przetwarzany tekst może pochodzić od osób posługujących się bardzo oddalonymi od siebie językami np. młodzież i osoby starsze.

Jakie cechy języka sprawiają, że tekst w kontekście uczenia maszynowego należy patrzeć jak na wyzwanie? Jedną z nich jest **niejednoznaczność/wieloznaczność**. Nietrudno znaleźć przykłady słów o różnych możliwych znaczeniach, jak choćby "ognisko" czy "zamek". Kontekst sprawia, że z reguły umiemy odróżnić znaczenie słów w szerzej perspektywie, całego zdania lub tekstu. Do tego w języku występują idiomy, czyli konstrukcje językowe, w których znaczenie zestawienia słów jest inne, niż to literalne, wynikające z poszczególnych wyrazów. Przełożenie tego na zero-jedynkowy język maszynowy nastręcza poważny problem, co można sprawdzić choćby wpisując frazy zawierające wyrazy wieloznaczne lub idiomy do gotowego systemu NLP, takiego jak Google Translate (Vajjala i in, 2023, s. 38). W dniu pisania tych słów, chcąc przetłumaczyć "owijać w bawełnę" Autor otrzymał angielski ekwiwalent "beat around the bush". Fakt ten uznamy zapewne za świadectwo dużego zaawansowania popularnego tłumacza. "Nie dziel skóry na niedźwiedziu" wciąż zwróci jednak literalne "Don't split the bear's skin", zamiast znaczeniowego odpowiednika "Don't count your chickens before they're hatched". Ciekawostką jest, że angielskie słowo "ambiguous" także jest wieloznaczne (Stanford Encyclopedia of Philosophy, 2011). W języku polskim mamy przynajmniej intuicyjne rozróżnienie między czymś niejednoznacznym, czyli trudnym do jednoznacznej oceny, a wieloznacznym - mającym wiele możliwych znaczeń.

Ważną cechą każdego języka jest jej powiązanie z **"powszechną wiedzą"**. Trudno wyobrazić sobie komunikat, który w jakimś stopniu nie odnosi się do znajomości pewnych zewnętrznych, niewspomnianych w samym komunikacie, informacji. Pojęcie to zostało wzięte w cudzysłów, ponieważ "powszechny" jest tutaj literalnym, choć nie do końca oddającym faktyczny sens, tłumaczeniem słowa "common". Każdy dokument czy konwersacja może bowiem odnosić się de facto do innego zakresu wiedzy, która może być "powszechna" tylko dla jej czytelników i uczestników.

Poza regułami w każdym języku jest również przestrzeń na eksperymenty. **Kreatywność** jest kolejnym aspektem języka, który trudno jest "zrozumieć" komputerom. Dobrym przykładem jest poezja (Vajjala i in, 2023, s. 38). W wierszach, ale nie tylko, często mamy do czynienia z poszukiwaniem nowych metafor czy słowotwórstwem świadomie przekraczającym standardowe ramy.

Następnie **różnorodność**, trudności z którą mają bardzo złożony charakter. Jak wskazano już powyżej na przykładzie pary "powszechny"/"common" często nawet przyjęte, słownikowe tłumaczenia słów nie do końca oddają dokładnie ich znaczenia. Do tego, co również zostało wspomniane powyżej, języki różnią się między sobą w zakresie struktury i zasad. Rozmaitość języków idzie znacznie dalej. Przykładowo język angielski jest często określany jako precyzyjny i konkretny, m.in. z uwagi na liczebność czasów, które pozwalają dużo informacji umieścić w samej zastosowanej konstrukcji zdania. Polski język zdecydowanie nie posiada tej cechy, wymaga często zastosowania dużej ilości słów i jest bardziej kontekstowy. Tego typu

porównania można by mnożyć. Cześć języków np. tajski nie używa spacji między słowami oraz wielkich liter jako sygnalizacji początku zdania. Wszystko to czyni bardzo trudnym przygotowanie uniwersalnych rozwiązań z zakresu NLP. System sprawdzający się w jednym języku może być bezużyteczny w drugim (Vajjala i in, 2023, s. 38). Różnorodność ma także inny aspekt w kontekście tematu pracy. Narzędzie przetwarzające tekst jest silnie związane z językiem, który ma być jego przedmiotem. Większość rozwiązań z zakresu sztucznej inteligencji może być bezproblemowo transferowana między krajami. Często wystarczające jest wykonanie lokalizacji wersji oryginalnej, najczęściej angielskiej, na język urzędowy danego kraju. W wielu branżach korzysta się powszechnie z języka angielskiego. W przypadku rozwiązań przetwarzających tekst niezbędne jest jednak znacznie więcej niż wykonanie tłumaczenia systemu. Z tego powodu część komercyjnych rozwiązań NLP nie jest dostępna dla języka polskiego, którym w skali globalnej posługuje się niewielki odsetek populacji. Dodatkowo jest on postrzegany jako bardzo trudny w opanowaniu, przede wszystkim z uwagi na rozbudowaną fleksję. Producenci nie widzą zapewne biznesowego uzasadnienia dla ponoszenia kosztów przygotowania programów dla potrzeb użytkowników mniej popularnych języków. W późniejszej części omówiony zostanie model wykorzystujący spaCy - bibliotekę NLP dla języka programowania Python. Jej wykorzystanie dla języka polskiego jest oparte na gotowych potokach bazujących na pracy wykonanej w ramach inicjatyw naukowych - Narodowego Korpusu Języka Polskiego oraz Polish Dependency Bank 2.0.

Co więcej, języki **ewoluują**. Z czasem słownictwo i struktura języka ulegają zmianie. Można to sobie łatwo uświadomić czytając teksty sprzed kilkudziesięciu lat w zestawieniu z wynikami popularnego plebiscytu na "Młodzieżowej Słowo Roku" organizowanego przez Wydawnictwo Naukowe PWN. Wyniki za rok 2023, w którym wyróżnione zostały "rel", "sigma", "oporowo" oraz "oddaje" dobrze oddają przykładowe mechanizmy tworzenia nowych słów i zmian ich znaczenia (Słownik Języka Polskiego PWN, 2023). Może wydawać się, ż tekst nie wyróżnia się na tym tle względem innych zmiennych. Literatura poruszająca tematykę praktycznego wykorzystania modeli predykcyjnych jest zgodna, że z czasem model traci zdolność do opisywania rzeczywistości. Stąd konieczność monitorowanie jego wyników, ponownego trenowania lub zmiany modelu (Treveil i in., 2020, Rozdział 7). Metody z zakresu NLP mają jednak najczęściej własne wytyczne dotyczące monitorowania wyników. Dla najbardziej zaawansowanych zastosowań stosowane jest np. automatyczne, codzienne trenowanie na nowych danych (Vajjala i in, 2023, s. 369). Wynika to z faktu, iż język, którym posługują się klienci w kontaktach z przedsiębiorstwem może zmienić się praktycznie z dnia na dzień np. na skutek konkretnego wydarzenia ze świata gospodarki, kultury lub polityki.

Z przetwarzaniem tekstu często wiązać się będzie też konieczność zaangażowania znacznych **zasobów systemowych**. Kwestię tę dobrze zobrazuje przykładowy model omówiony w dalszej części pracy. W jednej z faz pracy modelu bazującego na NLP powstanie bowiem tabelaryczny zbiór danych mający kilkanaście tysięcy kolumn, odpowiadających wyekstraktowanym z tekstu słowom bazowym. Duże

zapotrzebowanie na zasoby systemowe przekłada się oczywiście na czas działania rozwiązań bazujących na przetwarzaniu języka. W konsekwencji, zwłaszcza dla rozwiązań przetwarzających dane w czasie rzeczywistym, niezbędne jest zapewnienie adekwatnej mocy obliczeniowej.

Praktyczne zastosowanie NLP często będzie wymagało także pewnej wiedzy z zakresu językoznawstwa. Przydatna, lub wręcz niezbędna, może być znajomość zagadnień z gramatyki, fleksji, czy też słowotwórstwa, które są poruszane w toku edukacji szkolnej. W codziennym stosowaniu języka informacje takie jak części zdania czy deklinacje ulegają jednak zapomnieniu i zwykle ustępują intuicji i praktyce językowej. Poniżej zaprezentowane zostaną pokrótce najważniejsze pojęcia. Im bardziej zaawansowane narzędzia NLP tym większy będzie poziom wymaganej wiedzy na temat języka. Zagadnieniom z zakresu lingwistyki niezbędnym do celów modeli uczenia głębokiego poświęcone są odrębne, niezwykle szczegółowe, opracowania.

## 1.3 Części składowe języka

W odniesieniu do tematu pracy najbardziej użyteczne jest rozumienie języka jako złożonego z czterech podstawowych składowych. Są to: fonemy, morfemy i leksemy, składnia oraz kontekst. W praktycznych aplikacjach NLP elementy te są wykorzystywane w różnym stopniu (Vajjala i in, 2023, s. 33). Ogólne omówienie tych pojęć pozwoli na wprowadzenie w tematykę następnego rozdziału, który będzie dotyczył bezpośrednio przetwarzania języka za pomocą narzędzi NLP.

**Fonemy** są podstawowymi dźwiękowymi częściami składowymi języka. Poprzez ciągi fonemów formowane są pojedyncze słowa. Zapis dotyczący fonemów jest umieszczany między ukośnikami i można się z nim spotkać w sytuacji, kiedy obszarem zainteresowania poza samym tekstem, jest także wymowa (Wikipedia, 2024). Aplikacje NLP obejmujące dźwiękowy aspekt komunikacji, np. rozpoznanie mowy, identyfikację mówiącego czy też transkrypcję między tekstem a mową, będą wykorzystywały fonemy. W przypadku języka angielskiego najczęściej klasyfikuje się 44 fonemy, powiązane z pojedynczymi literami lub ich parami. W języku polskim, w zależności od klasyfikacji, identyfikowanych jest od 31 do 42 fonemów (Wikipedia. 2024).

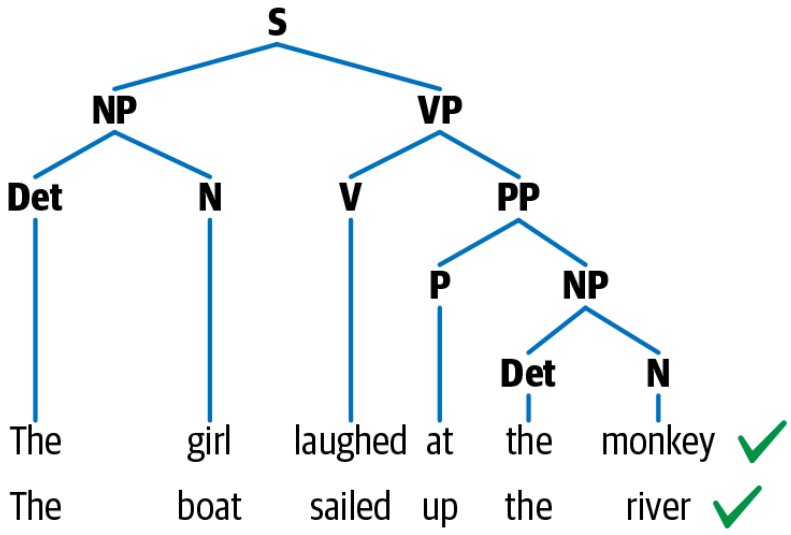
Przejście na wyższy poziom wprowadza pojęcia **morfemów** oraz **leksemów,** z których budowane są słowa. Morfemem określany jest najmniejszy niepodzielny ciąg fonemów, który posiada znaczenie. Niekoniecznie jest to całe słowo. Przyrostki lub wrostki także są morfemami, ponieważ niosą ze sobą informację modyfikującą znaczenie słowa lub słów, do których przylegają. Leksem jest pojęciem pozwalającym na grupowanie morfemów według znaczenia, fleksji (Vajjala i in, 2023, s. 34-35). Morfologia fleksyjna to proces, który pozwala na modyfikację słowa bazowego za pomocą przyrostków oraz

sąsiadujących słów, określający jego funkcję gramatyczną bez zmiany części mowy. W omówionym w dalszej części pracy modelu szczególnie pojęcia te będzie miało istotne znaczenie. Narzędzia NLP takie jak: tokenizacja, osadzanie słów, tagowanie części mowy, lematyzacja, bazują w znacznej mierze na identyfikowaniu i operacjach na leksemach i morfemach. Poniżej zaprezentowano przykład identyfikacji informacji na temat słów stworzonych na bazie morfemu "read".

| CONTEXT | SURFACE | LEMMA | POS | MORPHOLOGICAL FEATURES |
|---|---|---|---|---|
| I was reading the paper | reading | read | VERB | VerbForm=Ger |
| I don't watch the news, I read the paper | read | read | VERB | VerbForm=Fin , Mood=Ind , Tense=Pres |
| I read the paper yesterday | read | read | VERB | VerbForm=Fin , Mood=Ind , Tense=Past |

*Rys. 1. Morfologia fleksyjna zastosowana na słowie "read" w kilku zdaniach (źródło: spaCy, 2024)*

**Składnią** nazywamy zestaw reguł, obejmujących zarówno porządek, jak i relacje między elementami, umożliwiających łączenie słów w poprawne gramatycznie zdania. Dziedziną nauki dotyczącą składni jest syntaktyka. Znając reguły składniowe, które mogą być odmienne dla różnych języków, możemy dokonać procesu odwrotnego tj. podziału zdania na części. W języku NLP taki proces nazywa się parsowaniem (Vajjala i in, 2023, s. 36). Poniżej zaprezentowano przykładową analizę, w formie drzewa dwóch zdań o identycznej strukturze. Poszczególne etapy dzielą zdanie na coraz mniejsze części, obrazujące relacje między poszczególnymi frazami i słowami oraz ich rolę w całym zdaniu. Do metodologii NLP weszły w ten sposób szczegółowe pojęcia z domeny syntaktyki. Na poszczególnych poziomach drzewa widnieją więc zarówno poszczególne części mowy, jak i frazy. Przy pierwszym podziale zdania najczęściej spotkamy się z dwiema frazami: rzeczownikową (Noun phrase – NP) oraz czasownikową (Verb phrase - VP). Odpowiada to szkolnej wiedzy na temat podmiotu, przedmiotu i orzeczenia. Dalsza analiza przykładowego zdania prowadzi do wyróżnienia wyrażenia przyimkowego, konkretnie przedimkowego (Prepositional phrase – PP), które dalej ulega podziałowi na przedimek (Preposition – P) i frazę rzeczownikową zawierającą podmiot zdania.




*Rys. 2. Struktura składniowa dwóch podobnych zdań (źródło: Vajjala i in, 2023, s. 36)*

Najczęściej, aby przekazać treść, zdanie jest niewystarczające. W tym celu konieczne jest zestawienie dosłownego znaczenia zdania z **kontekstem**, obejmującym długoterminowe odniesienia, wiedzę powszechną (jak wskazano we wcześniejszej treści rozdziału może to oznaczać wiedzę wspólną tylko dla rozmówców czy wąskiej grupy odbiorców komunikatu) i zdrowy rozsądek (Vajjala i in, 2023, s. 36). Aby nauczyć się kontekstu narzędzia NLP wykorzystujące uczenie głębokie przetwarzają, nierzadko w czasie rzeczywistym, ogromne zasoby danych z różnych źródeł.

# Rozdział II: Przetwarzanie języka naturalnego - omówienie i najważniejsze narzędzia

## 2.1 NLP i jego zastosowania

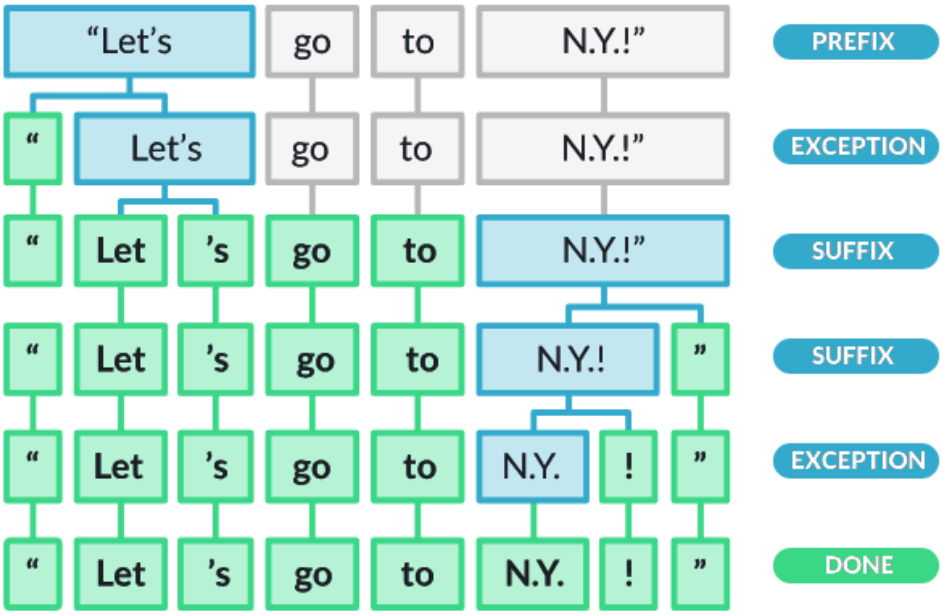
NLP to bardzo obszerne pojęcie. W najszerszym ujęciu mianem tym określa się wszystkie metodologie mające na celu formalizację interakcji między ludźmi i komputerami, w których wykorzystywany jest język (Ahmad, 2021, s.232). Jak wskazano już w pierwszym rozdziale NLP jest dziedziną interdyscyplinarną, w której krzyżują się zagadnienia z lingwistyki, informatyki, statystyki czy programowania. Co należy podkreślić forma komunikacji nie jest ograniczona do tekstowej. Obecnie wiele rozwiązań NLP odczytuje i generuje również mowę.

Komputerowe przetwarzanie języka ma swoje źródło w latach 40stych i 50tych. Za symboliczny początek uważany może być test Turinga, w którym jedno z zadań dotyczyło interpretacji i generowania wypowiedzi. W ten sposób jeszcze nie nazwane NLP weszło w skład tematów powiązanych z szeroko rozumianą sztuczną inteligencją. Początkowa, trwająca do początku lat 90tych, faza rozwoju dziedziny bywa nazywana symboliczną. Rozwiązania z tego okresu, w tym pierwsze chatboty, bazowały na zestawie ręcznie definiowanych reguł (Wikipedia. 2024). Z dzisiejszej perspektywy można je uznać za prymitywne, wyrażały jednak żywe zainteresowanie naukowców obszarem komunikacji między ludźmi i maszynami. W tym okresie zagadnienie to zaczęło także zyskiwać swoje miejsce w kulturze. Motyw samoświadomego komputera sterującego statkiem kosmicznym i komunikującego się z jego załogą pojawia się choćby w filmach: “2001: Odyseja kosmiczna” (1968) czy też “Obcy – ósmy pasażer Nostromo” (1979). W literaturze temat ten, jako jeden z aspektów sztucznej inteligencji, pojawił się nawet wcześniej.

Pomysł wykorzystania zdobyczy matematyki na gruncie komunikacji również dotyczy okresu bezpośrednio po drugiej wojnie światowej. Wykorzystanie procesu Markova do generowania tekstu datuje się na artykuł z 1948 r. “A Mathematical Theory of Communication” autorstwa C.E. Shanonna. Mapowanie prawdopodobieństwa występowania po sobie poszczególnych słów i fraz pozwala niewielkim nakładem kodu języka Python wytworzyć nowy tekst na wzór zadanego (Downey, 2015, s. 130). Tzw. łańcuchy Markova wykorzystywane są także obecnie w pojedynczych etapach bardzo zaawansowanych narzędzi NLP.

Wzrost mocy obliczeniowej, w połączeniu z rozwojem lingwistyki generatywnej, poskutkował znacznym rozwojem dziedziny, która od lat 90tych do dzisiaj notuje wręcz logarytmiczny wzrost. Rozwiązania, które kilka lat temu wydawały się najbardziej zaawansowane bledną w zestawieniu z przytoczonym wcześniej ChatGPT. Aktualnie NLP znajduje się więc w tzw. “neuronowej” fazie rozwoju. Do rozwiązywania zadanych problemów powszechnie korzysta się bowiem z sieci neuronowych (Wikipedia, 2024).

Z biegiem czasu NLP, jako autonomiczny obszar nauki o danych, wykształciło swój własny słownik terminów, które pochodzą z różnych dziedzin. **Korpus** (po łacinie "corpus" - ciało) to zbiór wszystkich dokumentów, z których pobrano i przetworzono dane wejściowe do rozwiązania danego problemu (Ahmad, 2021, s. 233). Tak zdefiniowany korpus jest źródłem, na którym model uczy się kontekstu. Dotyczy to zarówno skomplikowanego procesu uczenia głębokiego, jak i prostszych modeli. **Dokumentem** nazywamy pojedynczy fragment tekstu, wyodrębnioną obserwację w ramach korpusu. Czasem pod tym pojęcie źródła rozumieją tekst już poddany potokowi czynności w ramach NLP. W zależności od badanego problemu może to być zarówno stanowiący zamkniętą całość i liczący kilkaset stron dokument, jak i pojedynczy post na blogu, czy krótka wypowiedź klienta (Provost i Fawcett, 2023, s. 245). W omówionym później przykładzie korpus stanowić będą łącznie wszystkie zmienne tekstowe przyjęte do przetwarzaniami metodami NLP. Tekst powiązany z jedną obserwacją, w naszym przypadku pobrany z jednego ogłoszenia o pracę, można określić mianem dokumentu. W procesie analizy dokumentu jest on najpierw dzielony na **segmenty**, które możemy utożsamiać ze zdaniami. W większości języków oznacza to proste wydzielenie części między kropkami, znakami zapytania i wykrzyknikami. Potrzebne są jednak wyjątki, które powinny pewne zastosowanie znaków przestankowych wykluczyć np. po zastosowaniu ich w skrótach. Przytoczony powyżej język tajski wymaga zapewne bardziej skomplikowanego zestawu reguł. Następny etap podziału tworzy **tokeny,** czyli wektory zmiennych dotyczących pojedynczych fragmentów tekstu. Nie jest to pojęcie tożsame ze słowem, aczkolwiek w dużej ilości przypadków tak właśnie będzie. Algorytm dokonujący tokenizacji opiera się na specyficznych dla danego języka regułach, które mają pozwalać na oddzielenie faktycznych obiektów, które powinniśmy traktować jako odrębne słowa. W praktycznych zastosowaniach konieczne może być zdefiniowanie własnych reguł, które pozwolą nam wyodrębnić tokeny w założony sposób (Vajjala i in, 2023, s. 73). Poniżej przedstawiony jest przykład tokenizacji zdania w języku angielskim. Na rysunku widać sposób działania algorytmu. Najpierw dzieli on tekst na części oddzielone spacjami. Następnie działa od lewej do prawej i aplikuje reguły, które pozwalają mu na prawidłowe wydzielenie tokenów. Jak widać po rezultacie słowo "Let's" to w istocie dwa tokeny. Z kolei "N.Y." - skrót od New York, nie został we wcześniejszym kroku rozdzielony na segmenty oraz jest traktowany łącznie. Token reprezentujący określony obiekt nazywamy **encją**. W zależności od zastosowania pojęcie to może obejmować różne rzeczy. Encjami mogą być nazwy państw lub organizacji, nazwiska, nazwy miejsc, daty, produkty, identyfikatory ustaw itp. (Vajjala i in, 2023, s. 178).




Rys. 3. Tokenizacja segmentu w praktyce, działanie algorytmu z biblioteki spaCy (źródło: spaCy, 2024)

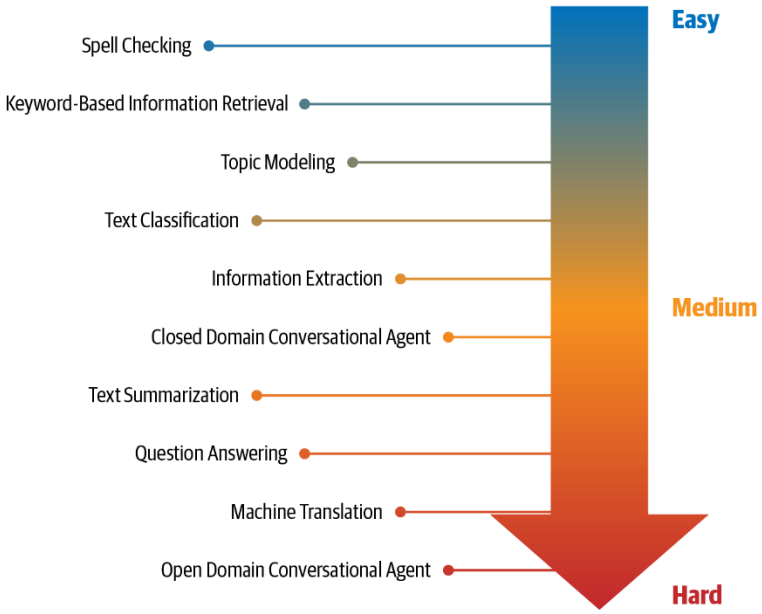
Jedną z wielu informacji, które mogą być przechowywane w wektorze cech tokenu są **lematy** i **stemy.** Te drugie bywają w polskich źródłach zamiennie nazywane tematami. W ogólnym ujęciu są to formy bazowe słów np. rzeczowniki liczby pojedynczej w mianowniku lub bezokoliczniki czasowników. Różnica polega na metodzie ich pozyskania. W przypadku lematów źródłem jest predefiniowany słownik. Stemy powstają z wykorzystania zestawu reguł, które np. identyfikują przedrostki, przyrostki i wrostki i usuwają je. W efekcie stem nie zawsze będzie formą poprawną lingwistycznie, co nie musi mieć znaczenia w kontekście budowanego zastosowania NLP. Przykładowo słowo "tynkowałem" po stemmingu może zwrócić jako stem "tynkać".

Jak już wspomniano wcześniej obecnie NLP znajduje wiele zastosowań praktycznych. Zwykły użytkownik aplikacji komputerowych może nie zdawać sobie sprawy, że korzysta z narzędzia, za którym stoją dokonania NLP. Warto więc podać kilka przykładów, nie wchodząc w szczegóły. **Sprawdzanie pisowni** dostępne choćby w programie Word jest jednym z takich rozwiązań. Mechanizm jego działania jest dość prosty - jeśli wpisane w edytor słowo nie występuje w słowniku zostaje podkreślone. Program proponuje również jego zastąpienie prawidłowym słowem o największym podobieństwie, definiowanym według zestawu reguł zastępowania liter, do tego które zostało zidentyfikowane jako błąd. W nowszych edytorach podobne reguły są definiowane co do gramatyki czy też interpunkcji. Przykładem **pobierania informacji na podstawie słów kluczowych** są algorytmy zwracające wyniki zapytań wpisywanych w wyszukiwarki internetowej. Do zadań z gatunku **klasyfikacji tekstu**, czyli przypisywania treści do kategorii, należą choćby filtry antyspamowe czy też analizy sentymentu, mające zidentyfikować dokumenty o konkretnym nastawieniu np. negatywne komentarze pod postem zamieszczonym w mediach społecznościowych. **Ekstrakcja informacji** jest podstawą narzędzi wydobywających z tekstu istotne informacje według wcześniejszych założeń. Szczególną formą tych rozwiązań jest **analiza sentymentu**, czyli identyfikacja

tekstów o odzwierciedlających emocje np. Bardzo złych recenzji produktu lub pozytywnych artykułów. Przy wykorzystaniu algorytmów z tego zakresu popularny produkt do obsługi poczty elektronicznej firmy Google proponuje nam dodanie do kalendarza nowego zdarzenia, ilekroć w treści otwartej wiadomości pojawi się data. Telefon może zaproponować dodanie do kontaktów nowego numeru telefonu, znajdującego się w wiadomości przesłanej poprzez komunikator. **Modelowanie tematyczne** to określenie rozwiązań mających zidentyfikować teksty dotyczące interesującego nas tematu. Są powszechnie wykorzystywane przez korporacje do automatycznego odszukiwania w wielu źródłach bieżących wiadomości na temat bliższego i dalszego otoczenia biznesowego. Z algorytmów tego typu korzystają także choćby pisarze chcący przyuczyć się w temacie, który będzie pełnił istotną rolę w ich nowej książce. Poniżej zaprezentowano najpopularniejsze zastosowania, według ich poziomu trudności. Cztery najtrudniejsze są zdecydowanie domeną uczenia głębokiego. **Streszczanie tekstu** czy **tłumaczenie maszynowe** nie wymagają wyjaśnienia. **Agentem konwersacyjnym** nazywane są zastosowania mogące przyjmować polecenia i zwracać wyniki z wykorzystaniem tekstu lub mowy. Przykłady to Siri koncenu Apple, Asystent Google, czy wspomniany ChatGPT. Wynikiem pracy agenta niekoniecznie musi być tekst, agenci będący aplikacjami w telefonie mogą np. wykonać połączenie lub stworzyć zapis w kalendarzu.

Na poniższym rysunku nie zaprezentowano oczywiście wszystkich rozwiązań, w tym tych z pogranicza NLP i innych dziedzin. W tej grupie na wspomnienie zasługują przede wszystkim często poprzedzające przetwarzanie języka narzędzia **OCR** (Optical Character Recognition), które pozwalają na ekstrakcję do zmiennej tekstowej tekstu z grafiki lub innych formatów plików zawierających edytowalny tekst np. pdf.

Warto dodać także, że metodologia NLP opracowała także swoje własne rozwiązywania pewnych problemów typowych dla uczenia maszynowego. Mianem **rozszerzania danych (*data augmentation*)** określane są sposoby na wytworzenie większej ilości danych z relatywnie niewielkiego zbioru. Jest to działanie analogiczne do oversamplingu. Technik rozszerzania jest wiele. Najprostszymi z nich są np. zastępowanie synonimów czy tłumaczenie zwrotne. Pierwsze podejście oznacza losowy wybór określonej ilości słów i zastąpienie ich w wygenerowanej obserwacji synonimami. Drugie rozwiązanie oznacza przetłumaczenie tekstu na inny język za pomocą gotowego narzędzia i następnie wykonanie procesu odwrotnego. Dla potrzeb generowania danych dla modeli uczenia głębokiego wykorzystywane są bardziej zaawansowane techniki jak np. generowanie szumu, czyli tworzenie nowych obserwacji poprzez symulowanie błędów w pisowni. Skonfrontowanie modelu z błędną pisownią pozwala to na trenowanie bardziej niezawodnych rozwiązań przetwarzających np. wpisy na platformach społecznościowych, które z definicji są obarczone problemami z pisownią (Vajjala i in, 2023, s. 63).

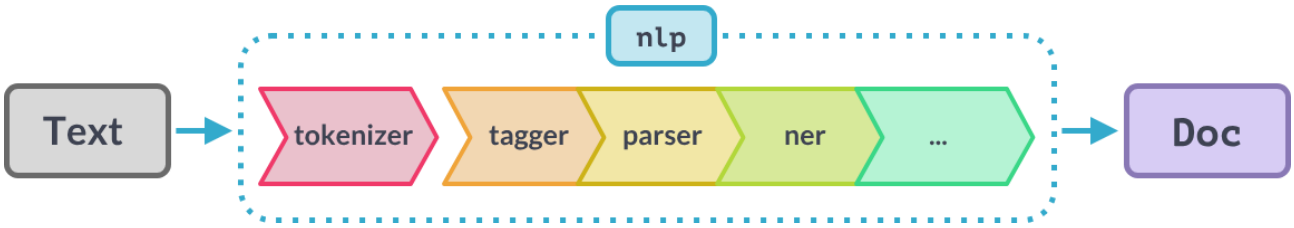



Rys. 4. Popularne zastosowania NLP według relatywnej trudności (źródło: Vajjala i in, 2023, s. 33)

## 2.2 Elementy potoku przetwarzania języka

Podobnie jak w przypadku innych zastosowań szeroko pojętej nauki o danych przetwarzanie języka odbywa się w potokach. Zdefiniowanie i rozplanowanie kolejnych kroków potoku jest stałym elementem budowy każdej praktycznej aplikacji NLP (Vajjala i in, 2023, s. 59). Potok dotyczący czynności stricte z zakresu NLP może być częścią składową całej orkiestracji projektu, niekoniecznie w całości związanego z przetwarzaniem języka. Tak będzie w przykładzie omówionym w dalszych rozdziałach. Poszczególne elementy potoku podlegają dostosowaniu do potrzeb danego rozwiązania. Poniżej zaprezentowano przykładowy potok. Jego wynikiem jest Doc, czyli obiekt, który w nomenklaturze biblioteki spaCy skupia w sobie wyniki wszystkich przeprowadzonych wcześniej czynności, nadających poszczególnym tokenom wartości kolejnych cech. Przykładowe elementy potoku zostały już częściowo omówione w poprzednim podrozdziale, przy okazji definiowana kluczowych pojęć z zakresu NLP. W poniższych rozważaniach celowo pominięte zostaną, jako wykraczające poza zakres pracy, niektóre z elementów nieodnoszących się ekskluzywnie do NLP. Dotyczy to np. etapu pozyskiwania danych, modelowania, wdrożenia czy ewaluacji. Można spotkać się z różnymi klasyfikacjami, oraz oczywiście kolejnością, poszczególnych czynności w ramach potoku. Dla potrzeb niniejszej pracy wprowadzono także sztuczne rozróżnienie na etapy potoku oraz narzędzia z zakresu NLP. Podział ten ma na celu podanie wiedzy w sposób ustrukturyzowany. W rzeczywistości pojęcia te często przenikają się. Poszczególne elementy potoku są w istocie także

rozwiązaniami NLP.




*Rys. 5. Przykładowy potok biblioteki spaCy (Źródło: spacy, 2024)*

### 2.2.1 Przetwarzanie wstępne i czyszczenie

Pojęciami tym są określane wszystkie czynności wykonywane wstępnie na pozyskanym tekście w celu przygotowania do dalszego procesowania. W tym etapie możliwe są różne czynności, zależne od potrzeb. Może on obejmować np. usunięcie podwójnych spacji lub innych niechcianych znaków lub zdefiniowanych wyrażeń, przetworzenie całości tekstu na małe litery, czy też poprawienie błędów typowych dla systemu, z którego dane zostały pozyskane. Przed poddaniem tekstu dalszemu przetwarzaniu wskazane może być formatowanie elementów, które będziemy chcieli w kontekście projektu zdefiniować jako encje. Może to obejmować np. ujednolicenie formatu dat, czy też zapisu tablic rejestracyjnych pojazdów. Swoje miejsce na tym etapie przetwarzania ma także **normalizacja**, czyli ujednolicenie lub zmiana kodowania znaków. Pominięcie tego rodzaju czynności może skutkować poważnymi błędami w dalszych krokach potoku (Vajjala i in, 2023, s. 67).

### 2.2.2 Segmentacja

Proces ten został omówiony przy okazji definicji segmentu. Bywa czasem nazywany także tokenizacją zdań. W poszczególnych gotowych rozwiązaniach zestaw reguł służących do wydzielania segmentów może być bardzo rozbudowany, dostosowany do potrzeb i właściwy dla przetwarzanego języka. Dodatkowo biblioteki oferują rozwiązania z zakresu, który w rozważaniach niniejszej pracy będzie dla celów porządkowych określony jako “narzędzia”, przydzielając do wydzielonych zdań zestawy cech. Przykładowo segmentacja z wykorzystaniem biblioteki NLTK w warstwie “span” zwraca listę z numeracją znaku rozpoczynającego i kończącego poszczególne słowa. Pozwala to np. na obliczenia średniej ilości słów w zdaniach czy długości słów (NLTK, 2023).

### 2.2.3 Tokenizacja

Proces, który wydziela z segmentu tokeny i niejako otwiera możliwość przypisywania do nich, w kolejnych etapach potoku, wartości poszczególnych cech. Ich katalog oczywiście nie jest zamknięty i może podlegać eksperymentom. Także same zasady podziału mogą podlegać modyfikacjom w zależności od informacji, które chcemy wyodrębnić. W wielu projektach NLP narzędzie do segmentacji i tokenizacji jest napisane od zera do konkretnego celu i właściwego przetwarzania interesujących danych (Vajjala i in, 2023, s. 73). W ramach tego etapu bywają stosowane także rozwiązania poprawiające błędy w pisowni. Algorytmy tego typu identyfikują tokeny niewystępujące w słowniku i zamieniają je na takie, które charakteryzują się największym podobieństwem według przyjętego punktu odcięcia.

### 2.2.4 Lematyzacja i steming

Pod względem słownikowym określenie lemat i zbudowany na jego bazie termin lematyzacja pokazuje fakt, iż NLP skupia w sobie osiągnięcia wielu dziedzin. Termin ten pochodzi pierwotnie z języka matematyki i oznacza pomniejsze twierdzenie, które jest stosowane dla wyodrębnienia etapu większego dowodu (Wikipedia, 2024). Twórcy narzędzi NLP najwyraźniej chcieli podkreślić podzielność zdań i fakt, że pojedyncze słowa niosą ze sobą informacje dotyczące szerszego ujęcia danego tematu.

Procesy te, wraz z ich otoczeniem, zostały zasadniczo opisane poprzez definicje lematu i stemu. Warto dodatkowo wskazać, że po etapie lematyzacji token, któremu nie zostanie przypisanie słowo bazowe może być odpowiednio potraktowany. Brak lematu w stosownej warstwie wektora cech pozwala na identyfikację prawdopodobnie błędnego, przynajmniej w rozumieniu wykorzystywanego narzędzia, słowa. W przykładowym modelu fakt ten zostanie wykorzystany do obliczenia procentowego udziały błędów w danym dokumencie.

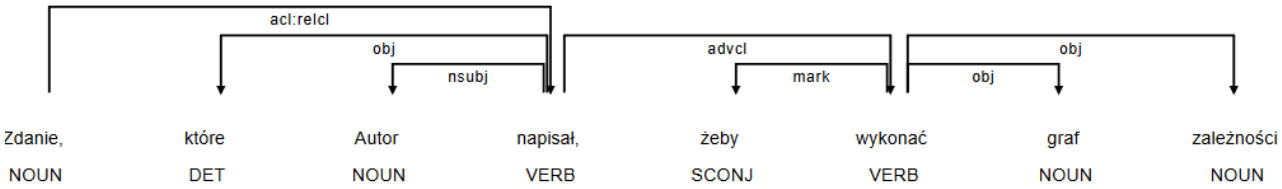
Pojęcia te bywają stosowane zamiennie. Twórcy biblioteki spaCy zrezygnowali z rozróżnienia na dwa procesy. Sprowadzania do słów bazowych jest nazywane jednorodnie lematyzacją, która może być oparta na słowniku lub regułach. Jest to zależne od języka danego tekstu. Języki o przejrzystych regułach i małej ilości wyjątków, jak angielski, preferują steming. Gotowe potoki biblioteki bywają dwuetapowe tj. gdy pierwszy etap, np. przygotowany słownik, zawiedzie podejmowana jest próba odnalezienia lematu w drugiej instancji. Niezależnie od przyjętej nomenklatury przyczyną stosowania lematyzacji jest najczęściej ograniczenie przestrzeni cech (Vajjala i in, 2023, s. 74). Z reguły nie chcemy bowiem aby nasze rozwiązania inaczej traktowało dokumenty, w których padną słowa np. “samochodem” i “samochodu”. Dla zastosowań NLP opartych na heurystykach czy modelach uczenia maszynowego wskazane będzie analizować oba użycia łącznie poprzez powiązanie w postaci lematu “samochód”.

### 2.2.5 Tagowanie części mowy (POS)

Identyfikowanie części mowy właściwej dla danego tokenu jest nazywane tagowaniem POS (part of speech), lub w skrócie tagowaniem. Choć proces ten bazuje na regułach specyficznych dla danego języka, działa inaczej niż omówione wcześniej elementy potoku. Algorytmy tagowania nie pracują na pojedynczym tokenie. Żeby prawidłowo określić część mowy, zwaną też tagiem syntaktycznym, niezbędna jest analiza całego segmentu, lub nawet większej porcji tekstu. Cześć słów może mieć bowiem identyczną formę w ramach różnych części mowy. Najważniejszą podpowiedzią co do właściwego oznaczenia są słowa sąsiadujące. Dla przykładu, w języku angielskim podmiot czynności (rzeczownik) poprzedza orzeczenie (czasownik). Prawidłowo pracujący algorytm tagowania jest kluczem do przetłumaczenia na język maszynowy struktury zdania i jego znaczenia. Czyni to tagownie niezbędnym elementem potoków w rozwiązaniach opartych na uczeniu głębokim. W powiązaniu z wynikami omówionych niżej: parsowania i identyfikacji encji, ma istotne znaczenie dla zaawansowanych rozwiązań NLP. Czynności te są czasem kolektywnie nazywane etykietowaniem sekwencji (*sequence labelling*). Algorytmy generujące tekst wykorzystują wyuczone informacje o następstwach i zależnościach części mowy do budowania wypowiedzi. (Jurafsky i Martin, 2023, Rozdział 8).

### 2.2.6 Parsowanie

W rozdziale I na jednym z rysunków zaprezentowano proste podejście do struktury składniowej zdania, które zostało podzielone na frazy: rzeczownikową i czasownikową. Tego typu podejście, zwane jest gramatyką bezkontekstową i nie jest jedynym możliwym. W naukach o języku rozróżniane są także inne gramatyki formalne, operujące własnymi nomenklaturami. Procesem, w ramach którego identyfikowana jest składnia zdania, w oparciu o przyjętą gramatykę formalną, jest parsowanie. Znając części mowy poszczególnych słów w segmencie możliwe jest połączenie ich w ewentualne frazy i określenie ich wzajemnych relacji. W tym kontekście zdanie jest siecią tych związków. W sposób graficznych wynik tego procesu przedstawiany jest za pomocą drzew parsowania. Szczegółowe omówienie tego elementu potoku wykracza poza zakres pracy. Reguły budowania zdań są specyficzne dla języka i przyjętego podejścia do kwestii składni. W nowoczesnych rozwiązaniach NLP obowiązującą teorią są tzw. gramatyki zależności, w których identyfikowane są związki podległości między poszczególnymi słów i frazami, z czasownikiem jako ich źródłem (Jurafsky i Martin, 2023, Rozdział 18). Nawet pobieżna analiza gotowych rozwiązań z tego zakresu pozwala na dostrzeżenie stopnia złożoności zjawiska. Przykładowo wspomniany wcześniej Polish Dependancy Bank rozróżnia 67 rodzajów relacji między 17 typami tagów części zdania.




*Rys. 6. Drzewo parsowania (źródło: opracowanie własne)*

### 2.2.7 Identyfikacja encji (NER), rozwikływanie odniesień i ekstrakcja relacji (RE)

Odnalezienie w tekście encji (*named entity recognition*) o interesującym nas charakterze może być częścią złożonego procesu, mającego na celu wydobycie z tekstu większej ilości informacji. Encje łączyć mogą relacje, które także są przedmiotem zainteresowania części zastosowań NLP. Algorytmy służące ekstrakcji relacji (RE), korzystając z wyników tagowania i parsowania, identyfikują te powiązania i wskazują na ich charakter. Jest to ważny etap m.in. systemów ekstrakcji informacji, odpowiadania na pytania czy też agentów konwersacyjnych. Dla przykładu organizacja chcąca badać na bieżąco publikacje z zakresu finansów będzie potrzebować aktualizowanej bazy wiedzy, łączącej ze sobą organizacje z nazwiskami ludzi (np. prezesem, członkami zarządu) i wydarzeniami (Vajjala i in, 2023, s. 188). Zarówno NER, jak i ER pokazują płynność rozwiązań w ramach NLP. Jako gotowe wytrenowane modele stają się elementem potoku gotowych bibliotek. Algorytmy identyfikujące encje są same w sobie rozwiązaniami NLP, dokonującymi klasyfikacji tokenów. Rozwiązania z zakresu ekstrakcji relacji wiążą ze sobą encje, określając także charakter ich relacji. Modelowanie obu tych zagadnień jest zwykle procesem dwuetapowym. W pierwszej kolejności dokonywana jest klasyfikacja binarna odpowiadająca na elementarne pytania: czy dany token jest encją według zadanych kryteriów, czy między tokenami występuje relacja. Następnie encje i ich wzajemne powiązania są klasyfikowane wieloklasowo. Dla przykładu w ramach kompletnego procesu Polska zostanie zidentyfikowana jako państwo (NER) a Donald Tusk jako jego Prezes Rady Ministrów (ER). Rozwiązania problemu klasyfikacji w tekście mogą być oparte zarówno na ręcznie opracowanych regułach wykorzystujących np. informacje dotyczące zapisu czy składni, jak i na modelach nadzorowanego uczenia maszynowego, w tym rekurencyjnych sieciach neuronowych (Vajjala i in, 2023, s. 189).

Dodatkowo elementem potoku może być rozwikływanie odniesień, czyli identyfikowanie słów, które odnoszą się do tokenów rozpoznanych już jako encja. Dotyczy to słów takich jak np. “on”, “jego” czy “jej”, które zostają powiązane odniesieniem z obiektem, którego dotyczą np. nazwą organizacji lub nazwiskiem osoby.

| TEXT | LEMMA | POS | TAG | DEP | SHAPE | ALPHA | STOP |
|---|---|---|---|---|---|---|---|
| Apple | apple | PROPN | NNP | nsubj | Xxxxx | True | False |
| is | be | AUX | VBZ | aux | xx | True | True |
| looking | look | VERB | VBG | ROOT | xxxx | True | False |
| at | at | ADP | IN | prep | xx | True | True |
| buying | buy | VERB | VBG | pcomp | xxxx | True | False |
| U.K. | u.k. | PROPN | NNP | compound | X.X. | False | False |
| startup | startup | NOUN | NN | dobj | xxxx | True | False |
| for | for | ADP | IN | prep | xxx | True | True |
| $ | $ | SYM | $ | quantmod | $ | False | False |
| 1 | 1 | NUM | CD | compound | d | False | False |
| billion | billion | NUM | CD | pobj | xxxx | True | False |

*Rys. 7. Wynik przykładowego potoku z wektorami zmiennych poszczególnych tokenów (Źródło: spaCy, 2024)*

W ramach potoków spotkamy się także z innymi terminami, których przyswojenie jest istotne dla zrozumienia języka, którym posługiwać się będzie literatura dotycząca tematu NLP. Krokiem przetwarzania tekstu może być choćby **identyfikacja kształtu**, która wskazuje strukturę znaków tokenu. Kształt, jak zwykła zmienna tekstowa, może podlegać dalszemu przetworzeniu w celu np. wykluczenia z docelowego tekstu cyfr, zmiany formatu dat lub definiowania w reguł w zastosowaniach NLP opartych na heurytykach. Dokonać tego można przy wykorzystaniu **wyrażeń regularnych** – zdefiniowanych wzorców kształtu i wykorzystanych znaków używanych do identyfikacji podłańcuchów tekstu (Vajjala i in, 2023, s. 41). Często wykorzystywanym pojęciem są także **słowa stopu (*stop words*)**, czyli powszechnie występujące w języku wyrazy, które z reguły nie wnoszą do tekstu istotnych informacji w kontekście badanego problemu. Dotyczy to np. spójników czy przyimków, które często są wyłączane z przetworzonego tekstu, ponieważ ich obecność w danych treningowych nie ma powiązania z kategorią, w ramach której została sklasyfikowana dana obserwacja (Vajjala i in, 2023, s. 73). Potoki NLP, którymi przetwarzane są teksty w różnych językach zawierają element identyfikujący język i dobierający na tej podstawie właściwe narzędzia dla dalszych etapów obródki. Spotkać można się z krokiem umożliwiającym korzystanie z różnych języków, ułatwiającym pracę z tekstem np. zawierającym specjalistyczną, łacińską terminologię. To samo dotyczy zestawu znaków np. różnych alfabetów. Przekształcenie jednego sposoby zapisu na drugi określane jest mianem **transliteracji**.

## 2.3 Reprezentacja tekstu

Aby poprzedzić praktyczny przykład wykorzystania NLP należy omówić jeszcze łącznik między przetworzeniem języka przez potok, a architekturą docelowego rozwiązania np. modelem predykcyjnym. Przetworzony tekst musi w tym celu być poddany **inżynierii cech (*feature engineering*)**, czyli zbiorowi metod, które pozwolą na jego przedstawienie w formie liczbowej, "zrozumiałej" dla wybranego rozwiązania z zakresu data science. Proces ten jest nazywany też reprezentacją tekstu (Vajjala i in, 2023, s. 80). Używany jest także ogólny termin **ekstrakcja cech (*feature extraction*)**, który w szerszym ujęciu odnosi się do znajdowania liczbowej reprezentacji danych wejściowych dla każdego projektu uczenia maszynowego. Dotyczy to zarówno tekstu, jak również obrazu czy dźwięku.

Już samo przetworzenie tekstu przez potok może być wystarczające dla przygotowania prostych zastosowań NLP opartych ma heurystyce. Zdefiniowanie elementarnych reguł nie będzie wymagać więcej niż gotowych wektorów cech poszczególnych tokenów lub ich niezaawansowanego przetworzenia np. za pomocą wyrażeń regularnych. Przykładowo na tej podstawie możemy identyfikować dokumenty zawierające określone słowo czy encję, adres e-mail lub kod produktu. Reguły mogą także stanowić źródło zmiennych dla modeli predykcyjnych.

Pewną szczególną formą reprezentacji tekstu może być **chmura wyrazowa**, czyli graficzne przedstawienie najczęstszych terminów w korpusie. W połączeniu z wartością zmiennej celu rozwiązanie to może pomóc zidentyfikować najważniejsze słowa wskazujące na przynależność do danej kategorii.

### 2.3.1 Kodowanie one-hot

Jest to bardzo prosta, stąd obecnie rzadko używana, metoda reprezentacji tekstu, w której każde słowo występujące w słowniku korpusu otrzymuje swój unikatowy identyfikator. Następnie poszczególnym słowom przypisywany jest wektor binarny, który wypełniany jest zera poza pozycją odpowiadającą identyfikatorowi danego słowa. Przykładowo, jeśli cały korpus stanowiłyby zdania "algorytm uczenia maszynowego" oraz "algorytm uczenia głębokiego", słowo "algorytm" otrzyma identyfikator 1, "uczenia" - 2, "maszynowego" - 3, "głębokiego" - 4. Po wektoryzacji postać pierwszego z segmentów wyglądałaby następująco: algorytm - [1,0,0,0], uczenia - [0,1,0,0], maszynowego - [0,0,1,0]. Drugie zdanie zostałoby przedstawione następująco: algorytm - [1,0,0,0], uczenia - [0,1,0,0], głębokiego - [0,0,0,1].

Podejście tego typu ma oczywiście liczne wady, w tym duży rozmiar wektorów one-hot dla dużych porcji tekstu, brak uwzględnienia podobieństwa między słowami czy problem **wyjścia poza słownik (*OOV – out of vocabulary*)**. Model wytrenowany na określonym słowniku nie znalazłby bowiem właściwej

reprezentacji dla nowych słów pojawiających się w zbiorze testowym. Cześć problemów tego typu kodowania znajduje rozwiązanie w drugim omówionym podejściu (Vajjala i in, 2023, s. 104).

### 2.3.2 Worek słów (*bag of words*)

Nastawienie to polega na przedstawieniu tekstu jako zestawu niepowiązanych ze sobą słów. W tym ujęciu każdy element słownika korpusu staje się słowem-kluczem, zmienną, której wartość dla każdej obserwacji należy przypisać. Jest to pojęcie niewrażliwe na kwestie porządku słów czy gramatykę. Zwykle także fleksję, ponieważ dzięki zastosowaniu lematów wybrane z tekstu może zostać ta postać słowa (Provost i Fawcett, 2023, s. 246). Metodę tę relatywnie łatwo zrozumieć i zaimplementować.

Podejście to może być rozwijane w oparciu o zmianę definicji cechy. Pozwala to na przynajmniej częściowe rozwiązanie problemu ignorowania kolejności wyrazów. Jest też oczywiste, że pewne zestawy wyrazów niosą ze sobą inną informację niż pojedyncze słowa, z których się składają. Informacja ta może być istotna w kontekście modelowanego zagadnienia. Jednym z takich rozwiązań jest więc zastosowanie **n-gramów**, czyli dzielenie tekstu na fragmenty o określonej ilości słów. W ten sposób wyróżnione zostaną frazy takie jak np. “wady i zalety” (3-gram lub inaczej trigram) czy “worek słów” (2-gram zwany bigramem). Metoda uwzględniająca n-gramy wciąż nie rozwiązuje problemy wyjścia poza słownik, pozwala jednak na uchwycenie przynajmniej części kontekstu (Vajjala i in, 2023, s. 104).

Stosuje się również podejścia rozszerzające cechę w oparciu o wykorzystanie zidentyfikowanych w tekście encji lub reguły np. składniowe. Przykładem mogą być **noun chunks** lub **verb chunks**. W narzędziach tego typu łączone są ze sobą rzeczownik lub czasownik i opisujące go wyrazy. Ilość słów jest więc płynna i odpowiada długość frazy rzeczownikowej lub czasownikowej w zdaniu. W ten sposób łącznie potraktowane zostaną takie zbitki wyrazowe jak “cały lewy bok”, “Dolina Krzemowa” czy “szybko i zdecydowanie odpowiedział”.

W ramach metody worka słów i jej rozszerzeń stosowane są różne metody przypisywania wartości, którym przyjrzymy się poniżej. Wybór sposobu kodowania może być kluczowy dla użyteczności powstałej reprezentacji słów dla rozwiązania analizowanego problemu. Na wynikach kodowania bazują także prostsze, nieuwzględniające składni, algorytmy wyliczające wartość podobieństwo dokumentów. Można przyjąć różne sposoby kodowania, przyjmuje się jednak, że żaden z nich nie jest wolny od podstawowych wad takiego podejścia: nieuwzględnienie porządku i relacji między wyrazami, generowanie wysokowymiarowych wektorów cech, co ogranicza możliwość uczenia modeli oraz problem OOV (Vajjala i in, 2023, s. 109).

**a) binarne**

Jest to bardzo intuicyjny sposób na umieszczenie słowa w przestrzeni wektorowej. Występowanie danego tokenu poskutkuje przypisaniem danej obserwacji 1, jego brak – 0. Jest stosowana, gdy większą wartość informacyjną niesie za sobą sam fakt wykorzystania słowa w segmencie, a nie częstość czy częstotliwość jego użycia.

**b) liczba wystąpień**

Reprezentacją tego typu nazywamy sytuację, w której wartość danej cechy rośnie wraz z kolejnym użyciem powiązanego z nią słowa. Wartość dla obserwacji odpowiada ilości użycia danego słowa w dokumencie. Przykładowo, jeśli przedmiotem zainteresowania będą lematy, zdanie "uczenie maszynowe nadzorowane różni się zastosowaniami od uczenia maszynowego nienadzorowanego" otrzyma wartość cechy 2 dla zmiennych "uczenie" i "maszynowe".

**c) częstotliwość (*true frequency* - TF)**

Jest to metoda określająca udział danego terminu (słowa lub n-gramu) w całości dokumentu. Liczba wystąpień dzielona jest więc przez długość dokumentu. Powstała w ten sposób wartość określa niejako ważność słowa lub frazy w kontekście całej obserwacji. Warto wskazać, iż jest to rodzaj kodowania wrażliwy na liczbę słów w tekście. Każde dopisanie do obserwacji nowego segmentu pomniejszy bowiem wartość TF dla niewystępujących w nim terminów. Wartość cechy określana jest wzorem:

$$\left(\frac{liczba\ wystąpień\ ter\ \min u\ \ w\ dokumencie}{\text{łączna}\ liczba\ terminów\ w\ dokumencie}\right)$$

**d) odwrotność częstotliwości występowania w dokumentach (*inverse document frequency* – IDF)**

Terminy z danego dokumentu można także opisać poprzez częstotliwość ich występowania w całym korpusie. W ten sposób wyższe wartości kodowania otrzymują słowa rzadsze w całym badanym kontekście. IDF obliczana jest za pomocą wzoru:

$$1 + \log\left(\frac{\text{łączna}\ liczba\ dokumentów}{liczba\ dokumentów\ zawierających\ termin}\right)$$

Podejście to rodzi własne problemy i zagadnienia do przemyślenia. Poszczególne słowa mogą bowiem przyjmować wartości zarówno bardzo duże, jak i bardzo małe. W ten sposób terminy mogą bardzo ciążyć na wyniku modelu, któremu przekażemy wynik kodowania jako dane wejściowe. Dla wielu zastosowań np. klasteryzacji, wydaje się to niewłaściwe. Można więc spotkać się podejściem narzucającym zarówno dolną, jak i górną lub górną granicę liczby dokumentów, w których słowo może wystąpić. Pozwala to na eliminację z modelu zarówno zbyt rzadkich, jak i zbyt powszechnych terminów (Provost i Fawcett,

2023, s. 248). IDF może być także narzędziem do eliminowania stop słów, ponieważ wartość najbardziej powszechnych terminów będzie bliska 1.

**e) TF-IDF**

Metoda ta stanowi połączenie dwóch poprzednich i ma na celu zwrócenie wartości wskazującej na wartość danego słowa w połączonym kontekście dokumentu i korpusu. Wartość jest wynikiem mnożenia TF x IDF. Wyznaczona w ten sposób waga słowa rośnie zarówno z jego obecnością w danym dokumencie, jak i odwróconą częstotliwością w korpusie. Metody tej używa się często do rozwiązań z zakresu klasyfikowania tekstu i wyszukiwania informacji, z reguły lepiej identyfikuje ona bowiem podobieństwa między słowami (Vajjala i in, 2023, s. 109).

Poniżej przedstawiano trywialny przykład, którego celem jest zobrazowanie wpływu wybranego sposobu kodowania na wartości zmiennych dla dwóch obserwacji składających się z kilkuwyrazowych zdań. Ze zbioru terminów wyeliminowano stop-słowa "z" i "w". Jak widać największą wartość kodowania TF-IDF przyjmie słowo "biznes" w drugim dokumencie. Ma ono tę samą wartość IDF co inne słowa, które pojawiają się w obu zdaniach (1+log(2/1)=~1,3). Jednak jego częstotliwość jest wyższa, ponieważ jest jednym z trzech (a nie czterech) słów w zdaniu.

| | studia | podyplomowe | data | science | biznes |
|---|---|---|---|---|---|
| | kodowanie binarne/liczba wystąpień | | | | |
| studia podyplomowe z data science | 1 | 1 | 1 | 1 | 0 |
| data science w biznesie | 0 | 0 | 1 | 1 | 1 |
| | kodowanie TF | | | | |
| studia podyplomowe z data science | 0,25 | 0,25 | 0,25 | 0,25 | 0,00 |
| data science w biznesie | 0,00 | 0,00 | 0,33 | 0,33 | 0,33 |
| | kodowanie IDF | | | | |
| studia podyplomowe z data science | 1,30 | 1,30 | 1,00 | 1,00 | 0,00 |
| data science w biznesie | 0,00 | 0,00 | 1,00 | 1,00 | 1,30 |
| | kodowanie TF-IDF | | | | |
| studia podyplomowe z data science | 0,33 | 0,33 | 0,25 | 0,25 | 0,00 |
| data science w biznesie | 0,00 | 0,00 | 0,33 | 0,33 | 0,43 |

*Tab. 1. Wpływ rodzaju kodowania na wartości wewnątrz wektora cech (źródło: opracowanie własne)*

### 2.3.3 Modele tematyczne

Jest to to metoda polegająca na wprowadzeniu dodatkowego poziomu umiejscowionego pomiędzy tokenami a dokumentem, zwanego warstwą tematyczną. Zadaniem algorytmu wprowadzającego tę warstwę jest dokonanie klasyfikacji, często za pomocą technik uczenia nienadzorowanego, słów do poszczególnych tematów. Danymi wejściowymi dla docelowego modelu stają się nie słowa, a informacja na temat ich przynależności do tematu (Provost i Fawcett, 2023, s. 257). Szczegółowe omówienie algorytmów modelowania tematycznego wykracza poza zakres pracy. Warto jednak wspomnieć, iż wykorzystując różne podejścia, w tym także omówione wcześniej miary, jak TF-IDF, algorytmy modelowania tematycznego identyfikują wyrazy kluczowe, które dzielą poszczególne dokumenty na grupy tematyczne. Jeśli na wejściu algorytm otrzyma teksty z zakresu NLP i rolnictwa powinien wskazać, iż “dane”, “lemat” czy “język” to terminy wskazujące przynależność tekstu do pierwszej klasy.

### 2.3.4 Osadzanie słów (*word embedding*)

Osadzanie to bardzo złożone zagadnienie, służące przygotowaniu wektorowej reprezentacji tekstu na potrzeby rozwiązań z zakresu uczenia głębokiego. Z tego powodu w niniejszej pracy zostanie potraktowany skrótowo, aby nakreślić jedynie bardzo ogólną intencję stojącą za tym pojęciem. Celem osadzenia jest możliwie najlepsze dobranie wartości wewnątrz wektora cech, aby odzwierciedlić znaczenia słowa w korpusie i umożliwić dalsze działania na nim.

Osadzanie należy do rozwiązań wykorzystujących koncepcję tzw. reprezentacji rozproszonych, które opracowano w odpowiedzi na wady omówionego powyżej podejścia worka słów. Kluczowym założeniem dla tej metody reprezentacji jest możliwość wywnioskowania znaczenia słowa z otaczającego go kontekstu, czyli tzw. podobieństwo dystrybutywne. Wykorzystując tę wiedzę algorytmy osadzania ma na celu zmniejszyć wymiarowość wektora cech powstałego jako wynik omówionych wcześniej metod, które z definicji mają długość odpowiadającą wielkości słownika dla całego korpusu (Vajjala i in, 2023, s. 109).

W tym celu algorytm wykorzystuje omówione wcześniej narzędzia i pojęcia, w szczególności reprezentacje rozproszone i modelowanie tematyczne. Jest bowiem bardzo możliwe, iż słowa, które występują w podobnym kontekście i zakwalifikowane do tych samych tematów mają zbliżone znaczenie (Vajjala i in, 2023, s. 111). Metod osadzania jest wiele, często napisanych pod kątem docelowego rozwiązania. Sieć neuronowa, która otrzyma powstałe w ten sposób wektory jako dane wejściowe ma za zadanie móc prawidłowo “zidentyfikować” znaczenie słów i relacje między słowami. Wektory powinny dać możliwość wykonywania prostych działań, które mają zwracać słowo będącego prawidłowym wynikiem konkretnego rozumowania. Książkowymi przykładami są proste równania jak: równanie król - mężczyzna +

kobieta = królowa. Zaawansowane algorytmy osadzania sprawiają, że współczesne rozwiązania z zakresu NLP potrafią wyszukiwać analogie czy generować spójny i poprawny tekst na zadany temat.

Osadzenie słów służy z reguły innym celom niż omówione wcześniej podejścia z grupy worka słów. Warto jednak odnotować, iż wektory będące wynikiem osadzenia, nawet zaawansowanych algorytmów, są z reguły niskowymiarowe i gęstsze niż takie, których długość determinuje wprost rozmiar słownika korpusu (Vajjala i in, 2023, s. 111). Pojedyncze słowa mogą być więc przedstawione jako zbiór wektorów, powiązanych za pomocą warunków lub w inny, które łącznie będą mieć znacznie mniejszy wymiar niż reprezentacja typowa dla podejścia worka słów.

Metody wytwarzające wektory wymagają z reguły bardzo dużego korpusu tekstu w celu wytworzenia kontekstu dla przetwarzanych słów. M.in. z tego powodu trenowanie osadzeń jest bardzo kosztownym procesem. W efekcie w powszechnym użyciu są zestawy wytrenowanych osadzeń, opartych na różnych zbiorach wejściowych np. artykułach prasowych czy też Wikipedii. Najpopularniejsze tego typu rozwiązania to np. Word2Vec (Google), GloVe (Stanford), fastText (facebook), ELMo (część TensorFlow, popularnej biblioteki języka) czy BERT (Patil, 2023). Wymienione zestawy wektorów oparte są na różnych korpusach i metodach obliczania. Ponieważ temat ten wybiega poza zainteresowania pracy, a jednocześnie jest niewątpliwie ciekawy i stanowi łącznik między omówionymi zagadnieniami NLP a zaawansowanymi rozwiązaniami, Autor zmuszony jest odesłać do dokumentacji wyżej wskazanych w celu uzyskania bardziej wyczerpujących informacji. Przykładowo metoda Word2Vec opiera się na wygenerowaniu dla poszczególnych słów wektorów z losowymi wartościami. Następnie wartości te są poprawiane poprzez przez dwuwarstwową sieć neuronową, analizującą rozkład są innych słów w korpusie (Vajjala i in, 2023, s. 114)

# Rozdział III: Przykład wprowadzenia zmiennych tekstowych do modelu

## 3.1 Definicja problemu biznesowego

Obecnie duża cześć procesów rekrutacyjnych w organizacjach odbywa się z wykorzystaniem popularnych serwisów oferujących możliwość zamieszczania ogłoszeń o pracę. Portale te stały się nieodłącznym elementem krajobrazu dla obu stron, zarówno firm poszukujących pracowników, jak i potencjalnych kandydatów. Polski potentat – Pracuj.pl, w opublikowanym raporcie za rok 2023 chwali się rekordami na wszystkich polach: w ilości opublikowanych przez pracodawców ogłoszeń (ponad 800 tysięcy), ilości użytkowników serwisu i wysłanych aplikacji (Pracuj.pl, 2024).

Także i z powodu popularności typu portale znalazły się na celowniku oszustów. Wyłudzenia związane z procesami rekrutacyjnymi są jednym z problemów doby internetu, obok spamu, phishingu, czy manipulacji treścią serwisów otwartych do edycji np. Wikipedii (Vidros i in., 2017). Oszustwa związane z rekrutacją on-line przybierają kilka postaci. Jedna z nich polega na publikowaniu fałszywych ogłoszeń, często podszywających się, także poprzez identyfikację wizualną, pod faktycznie istniejące firmy. Kandydaci odpowiadający na ogłoszenie są narażeni na możliwość wyłudzenia swoich danych osobowych, przekazywanych wraz z CV i innymi dokumentami, lub nawet kradzież pieniędzy. Oszuści podejmują bowiem z kandydatami korespondencję, podczas której przekazują, analogicznie do phishingu, linki służące pozyskaniu poufnych informacji lub zainfekowania komputera kandydata szkodliwym oprogramowaniem (BBC, 2024). Inną formą przestępstwa są włamania na serwery portali rekrutacyjnych w celu wydobycia danych użytkowników, zarówno kontaktowych, jak i dotyczących płatności za usługi portalu np. subskrypcje premium.

Z uwagi na skalę działania portale rekrutacyjne nie praktykują weryfikacji każdego ogłoszenia. Aplikują więc rozwiązania z zakresu sztucznej inteligencji celem wykrycia ofert wymagających weryfikacji. Są one integralną częścią systemu, który ma na celu obronę wiarygodności portalu w oczach jego użytkowników. Inne kanały, którymi mogą zostać zidentyfikowane próby wyłudzeń to np. kontakt ze strony użytkowników czy ogłoszeniodawców. Wykrycie fałszywych ogłoszeń nie jest zadaniem łatwym. Ma na to wpływ szereg czynników, w szczególności fakt, iż, celem oszusta jest stworzenie ogłoszenia możliwie nieodróżnialnego od prawdziwego. Odróżnia to tego typu próby wyłudzenia od phishingu, który z definicji jest nakierowany na zwrócenie uwagi użytkownika naiwnego. Wiadomości o tym charakterze przekładają więc często krzykliwość nad realizm. Co więcej, w przypadku ogłoszeń o prace relatywnie niewielki jest obszar danych kontekstowych, które mogłyby posłużyć modelom uczenia głębokiego. Oszuści starają się

także skrócić czas logowania do portalu. Ma to na celu uczynienie możliwie mało użytecznymi dla algorytmów uczenia maszynowego danych behawioralnych, obrazujących zachowanie użytkownika na stronie (Vidros i in., 2017). Co więcej w ogłoszeniach przewagę nad "klasycznymi" zmiennymi mają informacje tekstowe. Zadanie wykrywania oszustw w rekrutacji on-line jest więc jednym z zadań, w których zastosowanie znajdą omówione metody z zakresu NLP.

## 3.2 Zbiór danych

Dla celów pracy wykorzystane publiczne dane w zawarte w The Employment Scam Aegean Dataset (EMSCAD). Jest to zbiór przygotowany przez Uniwersytet Egejski w ramach badań nad oszustwami w rekrutacji on-line i możliwością automatyzacji procesu ich wykrywania. Zawiera on dane z autentycznych ogłoszeń rekrutacyjnych, zamieszczonych w latach 2012-2014 na portalu Workable. Spośród 17 880 ofert 866 zostało oznaczonych jako próby wyłudzeń przez dedykowany zespół pracowników portalu. Proces identyfikacji oszustw oparty był na weryfikacji informacji z kilku źródeł: podejrzanej aktywności na portalu, sygnałów od kandydatów i ogłoszeniodawców, fałszywych informacji kontaktowych lub o firmie, jak również okresowych analiz portfela (Vidros i in., 2017).

Od czasu publikacji zbiór ten cieszy się on uwagą entuzjastów data science, skupionych m.in. na portalu Kaggle, gdzie również jest dostępny. Zainteresowanie to obejmuje w szczególności pole eksperymentowania z metodami NLP. Zawiera on zmienne różnych typów, co czyni go szczególnie użytecznym w kontekście rozważań podjętych w pracy. Poza tekstem w zbiorze dostępne są zmienne o charakterze binarnym i kategorialnym. Możliwe jest więc zbudowanie modelu bazującego tylko na zmiennych "tradycyjnych" i uzupełnienie poprzez wykorzystanie reprezentacji tekstu uzyskanej za pomocą NLP. Szczegółowy opis poszczególnych zmiennych w zbiorze znajduje się poniżej:

| Nazwa | Opis | Rodzaj zmiennej | Braki |
|---|---|---|---|
| job_id | Identyfikator oferty | | - |
| title | Nazwa stanowiska | tekstowa | - |
| location | Lokalizacja miejsca pracy | tekstowa | 1,93% |
| department | Jednostka organizacyjna, której dotyczy oferta | tekstowa | 65% |
| salary_range | Wynagrodzenie | Wyrażenie regularne [zm. ciągła-zm. ciągła] | 84% |
| company_profile | Informacje o ogłoszeniodawcy | tekstowa | 19% |

| | | | |
|---|---|---|---|
| description | Opis stanowiska pracy | tekstowa | - |
| requirements | Wymagania | tekstowa | 15% |
| benefits | Benefity pracownicze | tekstowa | 40% |
| telecommuting | Czy oferuje możliwość pracy zdalnej? | binarna | - |
| has_company_logo | Czy ogłoszenie zawiera oznakowanie graficzne ogłoszeniodawcy? | binarna | - |
| has_questions | Czy do oferty dołączony jest kwestionariusz? | binarna | - |
| employment_type | Rodzaj zatrudnienia | kategorialna | 19% |
| required_experience | Wymagane doświadczenie | kategorialna | 39% |
| required_education | Wymagane wykształcenie | kategorialna | 45% |
| industry | Branża | tekstowa | 27% |
| function | Rodzaj stanowiska pracy | tekstowa | 36% |
| fraudulent | Oznaczenie jako próba oszustwa (zmienna celu) | binarna | - |

*Tab. 2. Opis zmiennych w danych wejściowych (źródło: opracowanie własne na podstawie (Kaggle, 2020)*

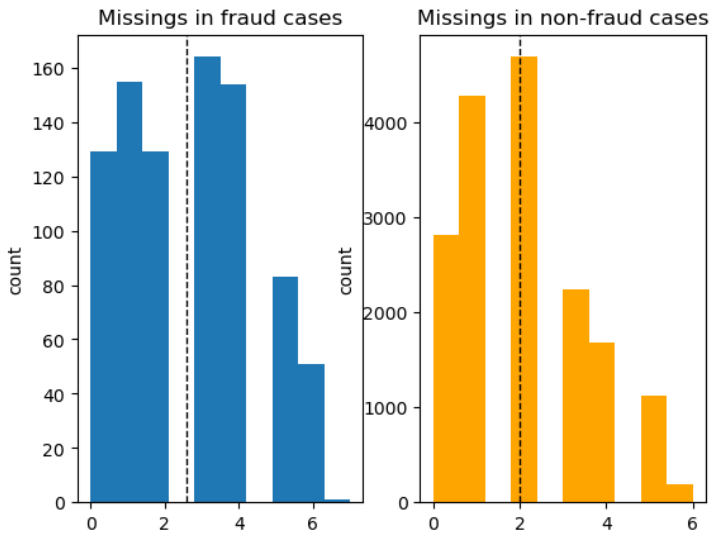
## 3.3 Model bazowy

Jak wspomniano powyżej w pierwszym ujęciu podjęta zostanie próba zbudowania modelu wykorzystującego zmienne, które nie wymagają procesowania metodami NLP. Oczywistymi zmiennymi do wykorzystania w takim ujęciu są zmienne binarne i kategorialne. Dodatkowo, po przetworzeniu z wykorzystaniem możliwości wyrażeń regularnych, w podstawowym modelu wykorzystać można informacje dotyczące lokalizacji oraz widełek wynagrodzenia. Ta pierwsza cecha zostanie przetworzona do postaci pozwalającej na jej traktowanie jako kategorialnej. Druga – zmiennej ciągłej.

### 3.3.1 Przetwarzanie danych

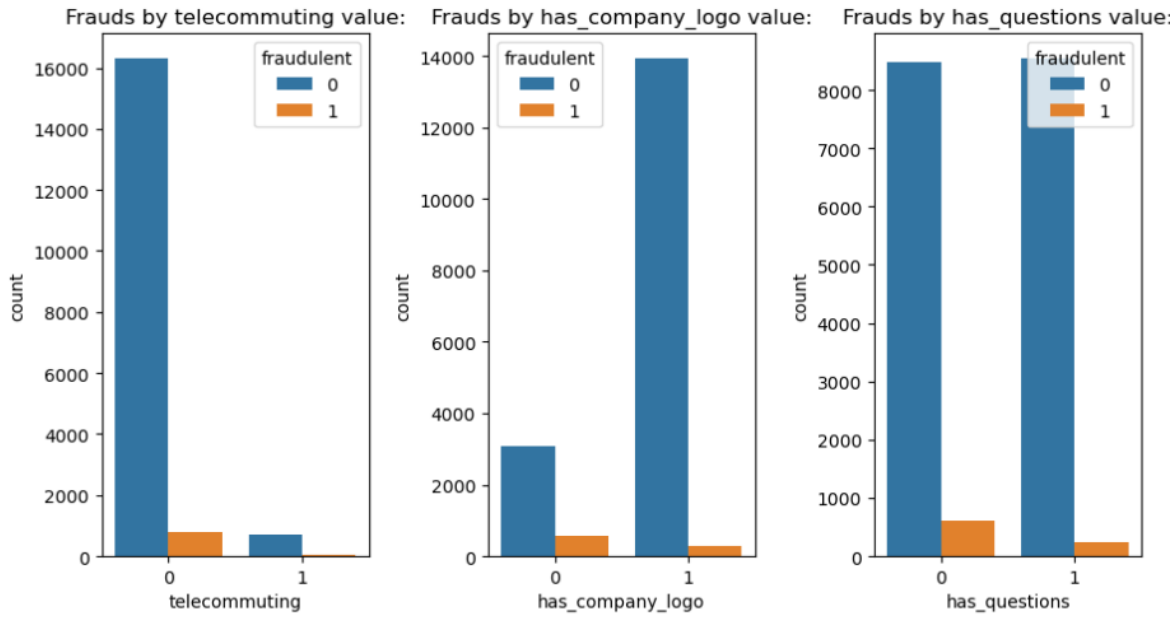
Wstępna inspekcja danych wydaje się uzasadniać przypuszczenie, że oferty stanowiące próbę wyłudzenia często zawierają braki w polach tekstowych. Fakt ten można łatwo przełożyć na nową zmienną skokową, określającą w ilu polach tekstowych brakowało wartości dla danej obserwacji. Dotyczy to siedmiu interesujących nas zmiennych tekstowych zawierających braki. Powstała zmienna ma więc minimalną

wielkość 0 i maksymalną 7. Aby ocenić przydatność zmiennej do modelu obliczono średnie dla dwóch kategorii zmiennej celu oraz zobrazowano graficznie, za pomocą histogramów, dystrybucję poszczególnych wartości. Ponieważ średnie różnią się (ok. 0,61) i rozkłady mają inny charakter pozostawiono zmienną w modelu.



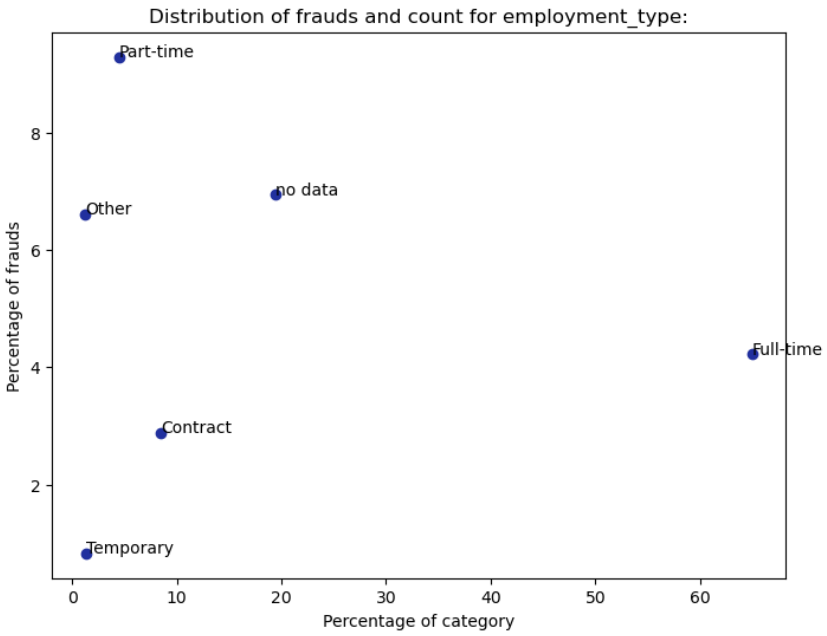

*Rys. 8. Histogramy, wraz ze średnią, liczby brakujących wartości zmiennych tekstowych (źródło: opracowanie własne)*

W następnym kroku obszarem zainteresowania będą zmienne binarne i kategorialne. Dla tych pierwszych graficzne zobrazowanie pokazuje zależności między ich wartościami a zmienną celu. Ogłoszenia bez identyfikacji graficznej i kwestionariusza zawierają większy udział oszustw. Wydaje się, że może to potwierdzać pewne podobieństwo, przynajmniej części prób wyłudzeń, do mechanizmu phishingu. Mniej profesjonalne ogłoszenia ma za cel przyciągnięcie potencjalnych ofiary, które z większym prawdopodobieństwem klikną w link. Z tego powodu część materiałów o charakterze phishingu jest dość oczywista np. zawiera błędy ortograficzne, wątpliwe jakościowo materiały graficzne lub wątpliwe tłumaczenia z innego języka. Co więcej, jak wskazano wcześniej, oszuści często minimalizują czas spędzony w serwisie, co także może być przyczyną braków na tych polach. Wątpliwość dotyczyć może oznaczenia pracy zdalnej, w przypadku której, z uwagi na skalowanie i relatywnie mały udział oszustw (problem *low-default portfolio*) zależność nie jest oczywista na wykresie. W celu oceny przydatności zmiennej posłużono się średnią. Ponieważ jest ona de facto prawie dwukrotnie wyższa (7,4% ogłoszeń o pracy zdalnej wśród oszustw, 4,1% wśród ofert oznaczonych jako prawidłowe) zmienną pozostawiono w modelu.




*Rys. 9. Liczba oszustw według poszczególnych kategorii zmiennych binarnych (źródło: opracowanie własne)*

Z uwagi na wskazane powyżej problemy dla zmiennych kategorialnych posłużono się wyliczeniami. Wykresy okazały się przeważnie niepraktyczne. W formie tabel ujęto więc poszczególne kategorie wraz z ich liczebnością i udziałem oszustw. Wcześniej jednak rozbito zmienną zawierającą informację o lokalizacji oferty na trzy kolumny oznaczające kraj, stan oraz miasto, w celu potraktowania jej jako zmiennej kategorialnej. Tego typu proste przetworzenie wykorzystuje pojęcie wyrażeń regularnych, stanowi więc dość prostą metodę z domeny NLP. W wyniku eksperymentacji ustalono, że tak szczegółowe informacje nie są zapewne użyteczne w kontekście modelowania. Gdyby tak ujętą cechę lokalizacji potraktować kodowaniem ilość zmiennych w modelu rozrosłaby się do poziomu uniemożliwiającego odpowiednią reprezentację poszczególnych kategorii. Z tego powodu za nowy punkt wyjścia przyjęto kraj. Wyjątek zrobiono dla ofert dotyczących Stanów Zjednoczonych. Udział ogłoszeń z tego kraju w całości wynosi 60%. Dystrybucja oszustw bardzo różni się między stanami. Nowa zmienna określająca kraj dla US oznacza więc de facto zbitkę kraju i stanu.




*Rys. 10. Wykres liczebności (oś x) i udziału oszustw (oś y) dla poszczególnych form zatrudnienia (źródło: opracowanie własne)*

Analiza wspomnianych tabel pozwoliła na dokonanie grupowania dla zmiennej oznaczającej lokalizację na sześć kategorii według udziału oszustw w danej grupie krajów lub stanów. Metodą prób i błędów przyjęto wartości 0% (ok. 16,58% ogółu obserwacji), 0-5% (ok. 44,31%), 5%-10% (ok. 29,81%),10-18% (ok. 6,92%), 18-35% (ok. 2,15%), >35% (0,21%). Ostatnia grupa jest mało liczna. Zdecydowano się jednak na jej wyróżnienie, ponieważ znajdują się w niej kraje o ponad pięćdziesięcioprocentowym udziale oszust np. Malezja czy Bahrajn. W przypadku zmiennej dotyczącej rodzaju zatrudnienia (rys. 10) połączono zbliżone do siebie kategorie "no data", powstałą przez imputację brakujących wartości, oraz "Inne". Zmienną łączącą się z wymaganym doświadczeniem pozostawiono bez ingerencji, ponieważ poszczególne kategorie różnią się od siebie, zarówno co do liczebności, jak i udziału oszustw. Ciekawostką może być fakt, że największy udział mają: stanowiska kierownicze (zapewne z uwagi na zasobność portfela potencjalnych kandydatów) oraz oferty bez wymaganego doświadczenia (jest to najpewniej próba wykorzystania braku doświadczenia osób potencjalnie zainteresowanych tymi ogłoszeniami). Także dla wykształcenia zdecydowano się na grupowanie według ryzyka. W tym przypadku podziału dokonano na podstawie analizy tabeli. Przyjęto progi <3%, 3%-8%, 8-12% oraz >12%. W ostatniej grupie znajduje się jedna możliwa wartość - "kursy w trakcie liceum", w której jako oszustwa oznaczono prawie ¾ wszystkich obserwacji. Świadczy to zapewne o stosowanym przez oszustów szablonie, mającym także na celu zainteresowanie osób niedoświadczonych. Przeprocesowane w ten sposób zmienne kategorialne poddano kodowaniu typu one-hot.

Zmienna oznaczająca widełki wynagrodzenia również została potraktowana jako wyrażenie regularne. Za podstawę docelowych wartości przyjęto dolną granicę widełek. Ponieważ na portalu mamy do czynienia z pensją w ujęciu rocznym, typowym dla krajów anglosaskich, tak uzyskaną zamieniono na jednostkę tysięcy oraz poddano logarytmowaniu o podstawie 2. Zabieg ten pozwoli na zestawienie z

wartościami innych zmiennych i jednocześnie sprawi, żeby duże wartości, typowe dla tego typu zmiennej, nie zaburzały modelu.

Zmienna tekstowa, która zostanie wykorzystana w kolejnych krokach, powstała poprzez połączenia wszystkich zmiennych o tym charakterze. Jednocześnie nazwę zmiennej celu zmieniono na "#fraudulent". Autentyczne słowo mogłoby bowiem poskutkować dublowaniem zmiennych na etapie przetwarzania tekstu w ramach potoku NLP.

Tak przetworzony zbiór danych zawiera 30 zmiennych: 3 binarne, 2 skokowe (liczba pustych wartości w polach tekstowych oraz wynagrodzenie), zmienną tekstową oraz 24 zmienne odpowiadające poszczególnym kategoriom. Zbiór jest następnie dzielony na dwie mniej więcej równe części, co ma na celu rozdzielenie obserwacji, które posłużą do uczenia modelu NLP.

### 3.3.2 Budowa i ocena modelu

Poddany wcześniej przygotowaniu zbiór danych został w przykładzie podzielony na zbiór treningowy i testowy w proporcji 3:1. Jednocześnie w kodzie została umieszczona możliwość sterownia parametrem dotyczącym oversamplingu. Ponieważ cały zestaw danych charakteryzuje się relatywnie niskim udziałem oszust, wskazane jest jego zastosowanie, aby wyczulić model na wartości cech poszukiwanych spraw. W przykładzie przyjęto wartość 10%, użyto prostego algorytmu, który kopiuje obserwacje bez modyfikacji wartości zmiennych. W efekcie rozkłady wartości zmiennej celu są następujące:

Initial distribution: [(0, 8522), (1, 418)]
Distribution in training set before oversampling: [(0, 6396), (1, 309)]
Distribution in training set after oversampling to 10.0%: [(0, 6396), (1, 640)]
Distribution in test set: [(0, 2126), (1, 109)]

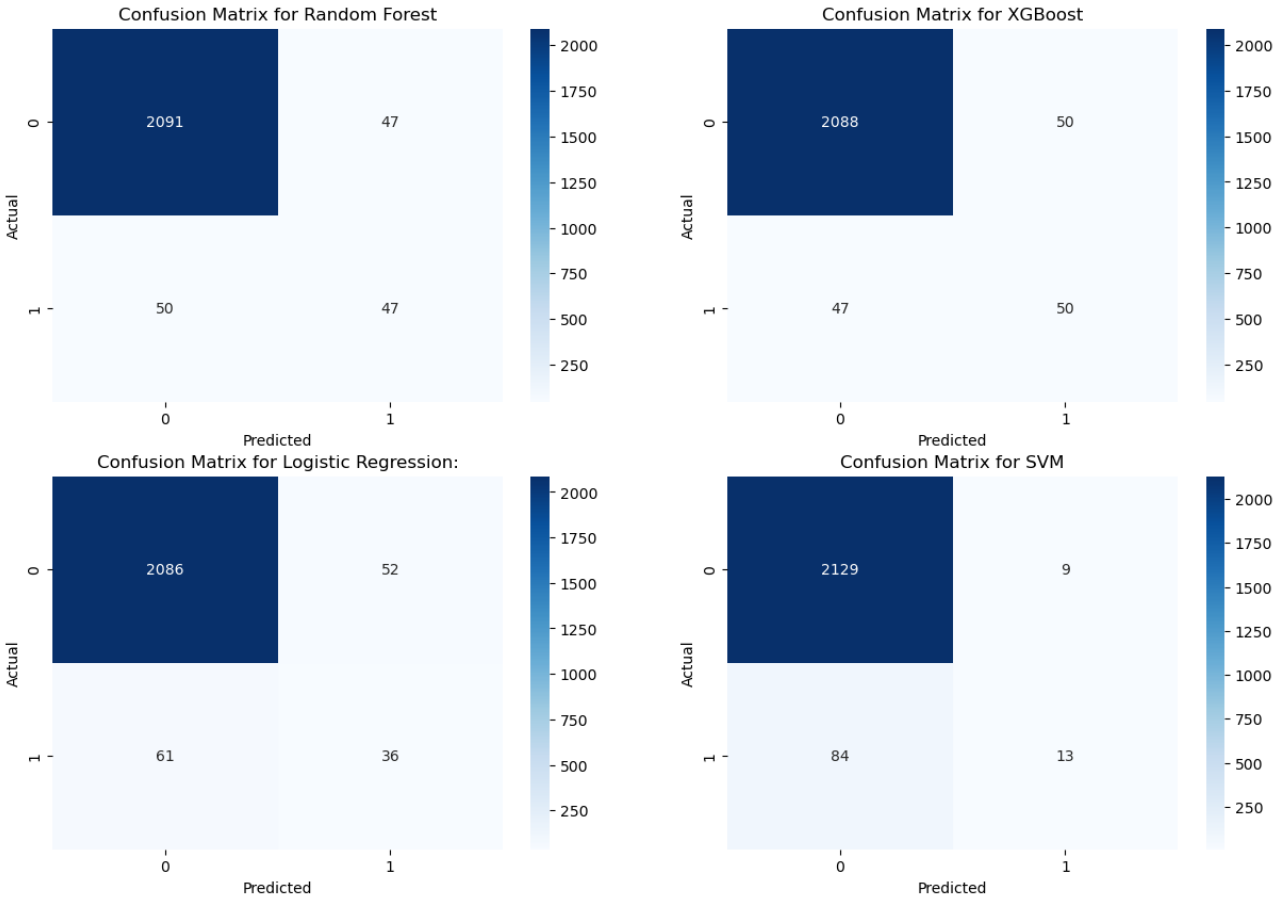
Ogólne wrażenie po analizie rozkładów wartości zmiennych w obserwacjach oznaczonych jako oszustwa prowadzi do wniosku, iż najprawdopodobniej najlepsze dla postawionego zadania będą rozwiązania oparte na metodach klasyfikacji z kategorii drzew decyzyjnych. Ze względu na swoją specyfikę drzewa mają bowiem zdolność do wykrywania nieliniowych związków między zmiennymi (Bruce i in., 2021). Aby możliwie najszerzej odnieść się do tezy zawartej w tytule pracy w przykładzie wytrenowane zostaną także, poza zagregowanymi modelami drzew opartymi na lasach losowych i XGBoost, modele liniowe – regresja i SVM.

Kod modelu bazowego przewiduje optymalizację hiperparametrów modeli, domyślnie jako cel tego zabiegu przyjęto maksymalizację czułości wyników. Jest to związane z potencjalnym wykorzystaniem

modelu jako jednego ze źródeł sygnałów trafiających do dedykowanego zespołu w organizacji. Wydaje się, że w takim kontekście czułość powinna być faworyzowana względem precyzji modelu. Takie podejście pozwoli na typowaniem liczniejszych, różnorodnych spraw, na których specjaliści mogliby dokonywać bardziej szczegółowych analiz, w celu lepszego poznania mechanizmów, którymi posługują się oszuści. Optymalizacja pod kątem precyzji skutkowałaby typowaniem mniejszej liczby, zapewne dość oczywistych spraw, co zmniejszyłoby użyteczność modelu w całym procesie. W uwagi na niski udział poszukiwanej wartości nie jest wskazane optymalizowanie pod kątem skuteczności. Poniżej zaprezentowano zestawienie zawierające metryki jakości oraz macierze pomyłek modeli:

| Models | Accuracy_bm | Precision_bm | Recall_bm | F1_bm | AUC_ROC_bm |
|---|---|---|---|---|---|
| Random Forest | 0.956600 | 0.500000 | 0.484536 | 0.492147 | 0.731276 |
| XGBoost | 0.956600 | 0.500000 | 0.515464 | 0.507614 | 0.746039 |
| Logistic Regression | 0.949441 | 0.409091 | 0.371134 | 0.389189 | 0.673406 |
| SVM | 0.958389 | 0.590909 | 0.134021 | 0.218487 | 0.564906 |

*Tab. 3. Metryki jakości modeli bazowych (źródło: opracowanie własne)*

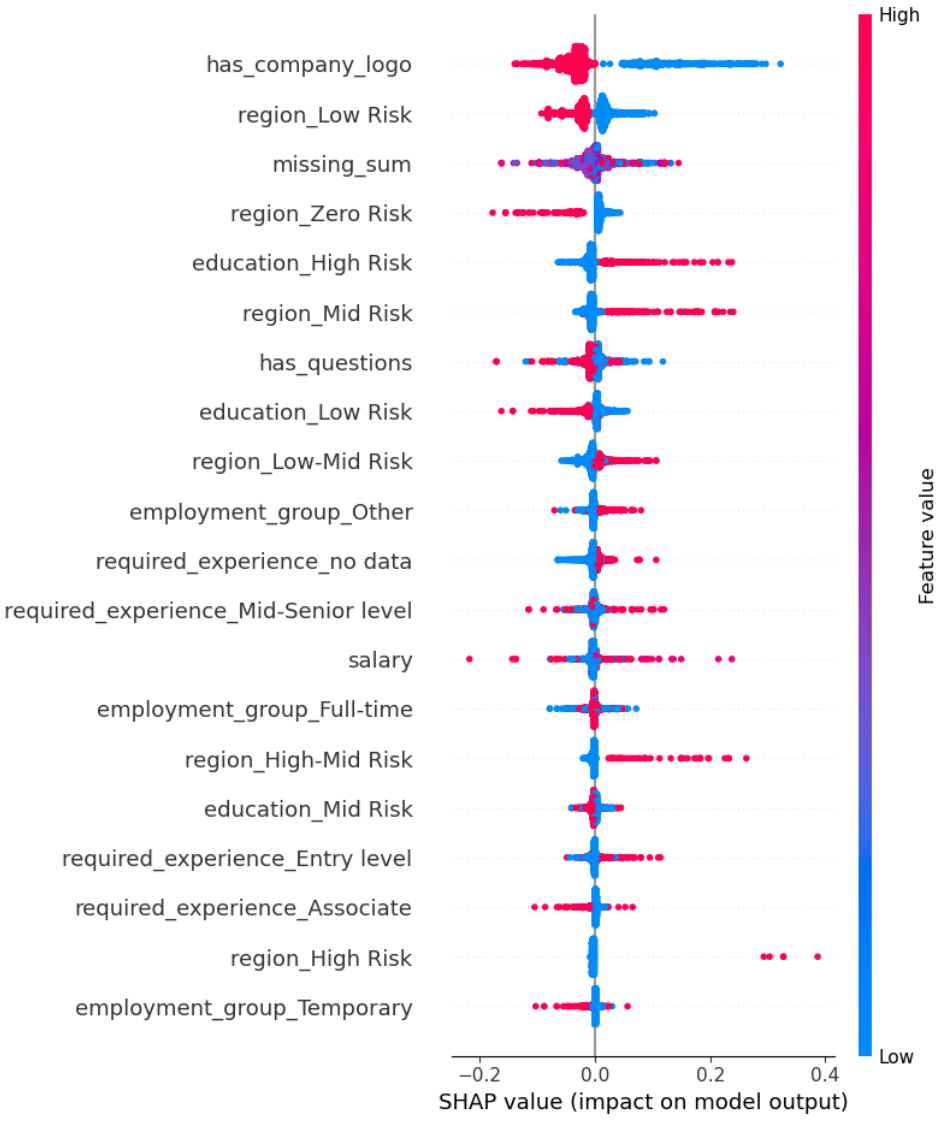



*Rys. 11. Matryce pomyłek modeli bazowych (źródło: opracowanie własne)*

Już w tej formie modele drzew decyzyjnych, w szczególności las losowy, wydają się znajdować praktyczne zastosowanie. Należy przy tym mieć na uwadze, że modele mające na celu przeciwdziałaniu wyłudzeniom charakteryzują się niezbyt imponującymi wskaźnikami służącymi ocenie ich jakości. Jest to związane z naturą problemu, który jest niedeterministyczny. Oszuści wciąż szukają nowych sposobów na przechytrzenie wymierzonych przeciwko sobie działań. Różnią się też między sobą poziomem profesjonalizmu i sposobami działania. W przypadku omawianego problemu nietrudno wyobrazić sobie bardzo podobne pod względem wartości cech ogłoszenia, z których tylko część będzie próbami wyłudzenia. Autor powoli sobie w tym miejscu na stwierdzenie, oparte na doświadczenia na polu przeciwdziałaniu przestępczości ubezpieczeniowej, iż model o czułości około 50% i obejmujący ponad 70% pola pod krzywą ROC byłby oceniony jako bardzo użyteczny. Modele liniowe mają wyraźnie niższe wartości metryk.

Celem sprawdzenia, które zmienne w kontekście całego modelu są najważniejsze, dla trzech modeli wygenerowano wartości SHAP (S*Hapley Additive exPlanations*). Jest to oparty na teorii gier sposób wyjaśnienia wyników algorytmów uczenia maszynowego (SHAP, 2018). Wygenerowany za pomocą tej metody wykres podsumowujący pozwala na ocenę stopnia ważności zmiennych i wpływ ich wartości na rezultat typowania. Cechami najbardziej determinującymi wynik dla modeli opartych na drzewach decyzyjnych są, zgodnie z przewidywaniami: obecność identyfikacji graficznej firmy, grupy ryzyka

lokalizacji i wymaganego poziomu edukacji, liczba braków w polach tekstowych czy obecność kwestionariusza. Warto podkreślić, że mniejsza wartość SHAP nie oznacza, że wartość danej cechy nie wykazuje korelacji ze zmienną celu. Może wynikać to z liczebności danej grupy, co widać na wykresie choćby w przypadku regionu o zdefiniowanym wysokim poziomie ryzyka. Jak widać wartość "1" dla tej zmiennej wyraźnie wpływa na wynik modelu. Jak wskazano w podrozdziale dotyczącym przetwarzania zmiennych jest to jedynie niewielki odsetek wszystkich obserwacji, stąd jej obecność na dole wykresu.

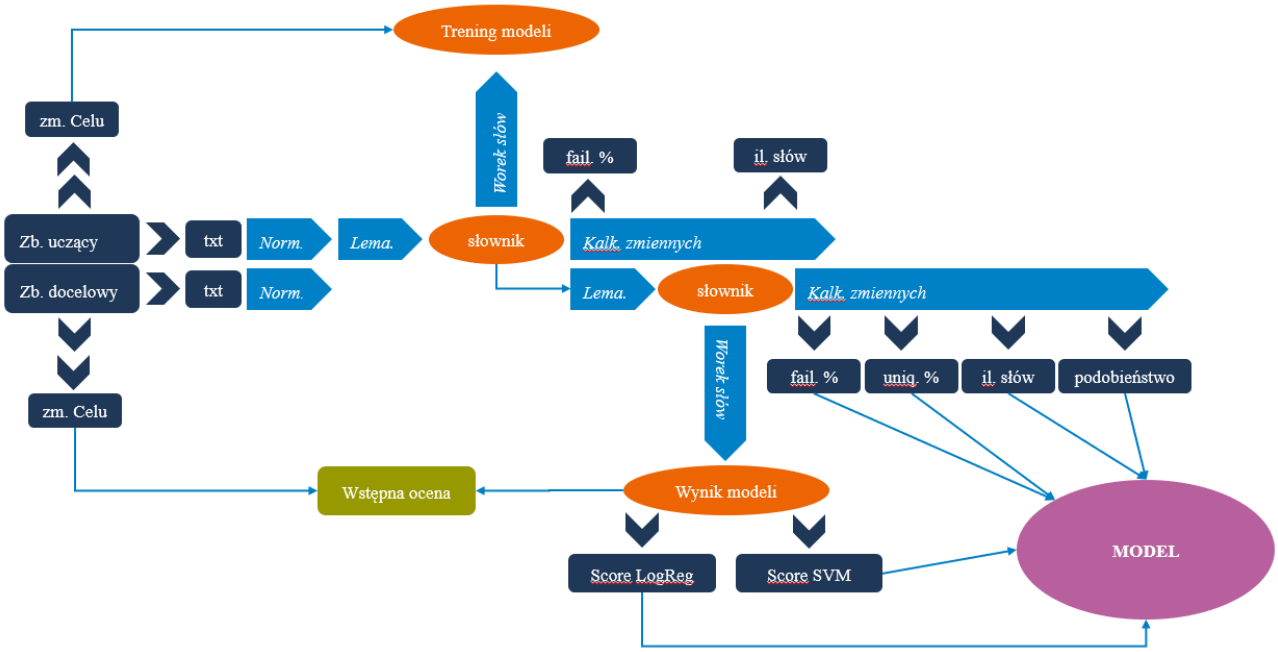



*Rys. 12. Wartości SHAP dla zmiennych w bazowym modelu lasu losowego (źródło: opracowanie własne)*

## 3.4 Potok NLP

W celu przetworzenia powstałem zmiennej tekstowej przez potok zbiór został podzielony na dwie części. Zachodzi tu pewna analogia ze zbiorami: treningowym i testowym. W poniższych rozważaniach będą one nazwane zbiorem uczącym i docelowym. Pierwszy z nich służy bowiem do wytrenowania modeli

bazujących na zmiennych pozyskanych w ramach potoku i ustalenia wzorca dla nowych zmiennych dodanych do zbioru docelowego. W tym celu oba zestawy obserwacji są jeden po drugim przetwarzane przez kolejne elementy potoku NLP. Schemat działania przedstawiono na poniższym schemacie, który od lewej ukazuje przejście od dwóch zbiorów wejściowych do przekazania do finalnego modelu sześciu nowych zmiennych.




*Rys. 13. Schemat działania potoku NLP (źródło: opracowanie własne)*

### 3.4.1 Normalizacja

W ramach tego kroku tekst jest konwertowany na małe znaki oraz usuwane są z niego znaki poza tymi określonymi w dedykowanej zmiennej. Zawiera ona wszystkie litery alfabetu oraz myślnik. Analiza wartości zmiennych tekstowych doprowadziła do wniosku, iż pojawiają się w nich znaki takie jak "#" czy "/", które mogłyby zaburzać proces lematyzacji. Co więcej w omawianym problemie nie zastosowano segmentacji, więc układ zdań nie będzie mieć znaczenia dla wyniku potoku. Następnie kolejne znaki dodawane są do pośredniej zmiennej celem oddzielenia słów pojedynczymi spacjami. Na tym etapie identyfikowane są również znaki oznaczające nowy wiersz i konwertowane na spacje. Skutkiem normalizacji są więc poszczególne słowa oddzielone pojedynczymi odstępami.

### 3.4.2 Lematyzacja

Tekst przetworzony w poprzednim kroku jest poddawany tokenizacji. Dla każdego tokenu badana jest obecność lematu według zasad przyjętych przez gotową bibliotekę dla języka angielskiego. Jeśli słowo posiada lemat, jest on dodawany do wynikowego ciągu znaków. Jeśli nie, token może zostać poddany drugiej instancji lematyzacji. Na tym etapie kod programu umożliwia bowiem zdefiniowanie własnego słownika, za pomocą którego można poprawić częste błędy. Ponieważ w ramach rozwiązania badanego problemu chcemy zidentyfikować także błędnie zapisane wyrazy, etap ten zostaje pominięty. Słowa bez lematu zostają dodane do wynikowego ciągu znaków jako słowo "#failed". Jako kolejną opcję w kodzie dodano możliwość ignorowania stop-słów. Przy takim wyborze lematy zidentyfikowane jako stop-słowa nie zostaną dodane do ciągu wynikowego. W omawianym przykładzie skorzystano z tej możliwości.

Etap lematyzacji na zbiorze docelowym zawiera jedną istotną różnicę. Aby nie zwracać dalej słów, które nie były obecne w zbiorze uczącym, także ich lematy nie stają się częścią ciągu wynikowego. Zamiast tego dodawane jest słowo "#unique", co pozwoli na obliczenie w dalszym kroku współczynnika, w jakim dana obserwacja posługuje się niewykorzystanymi wcześniej wyrazami.

W rezultacie każda obserwacja zostaje przedstawiona w formie lematów poszczególnych słów z wyjściowego łańcucha. Słowa nie posiadające lematu zostaną sprowadzone do postaci "#failed", nowe lematy w zbiorze docelowym jako "#unique". Na tej podstawie dla każdej obserwacji generowany jest słownik, który łączy dany lemat z jego liczebnością.

### 3.4.3 Kalkulacja zmiennych na podstawie słownika

Już na tym etapie, na podstawie tymczasowego słownika będącego wynikiem lematyzacji, dla poszczególnych obserwacji w zbiorze docelowym kalkulowane są trzy zmienne NLP: liczba słów, odsetek błędów oraz odsetek unikatowych słów. Intuicja stojąca za wykorzystaniem w modelowaniu pierwszej zmiennej jest związana z pułapką w rozumowaniu określaną jako błąd koniunkcji. Polega on na tym, że osoba może ocenić dwa zdarzenia łączne jako bardziej prawdopodobne i wiarygodne niż jednego z nich (Kahneman, 2012, s. 214). Na gruncie rozważań niniejszej pracy model powinien więc przynajmniej zbadać istnienie zależność między taką zmienną a prawdopodobieństwem próby wyłudzenia. Oszuści mogą starać się rozbudowywać opisy w tworzonych ogłoszeniach, żeby je uwiarygadniać. Z drugiej strony, poprzez analogię do phishingu, który stawia sobie za cel zmuszenie ofiary do szybkiego, nieprzemyślanego działania, także skąpe opisy mogą potencjalnie oznaczać próbę wyłudzenia. Podobna logika stoi za próbą wykorzystania w modelu odsetka błędnie zapisanych słów. Na tym etapie wydaje się, że zmienna określająca odsetek unikatowych słów nie będzie wyjątkowo przydatna w modelu. Można ją jednak wykorzystać w inny

sposób, np. do oznaczania do manualnej weryfikacji ogłoszeń posługujących się określonym odsetkiem "nowego" słownictwa. Pracownicy mogliby tego typu sprawy analizować jako obserwacje odstające np. w poszukiwaniu nowych wzorów wyłudzeń. Jest to pewna próba obejścia typowego dla worka słów problemu wyjścia poza słownik.

Ostatnią kalkulowaną na tym etapie zmienną jest binarne oznaczenie obserwacji, w których zmienna tekstowa jest bardzo zbliżona do zidentyfikowanego w zbiorze uczącym wyłudzenia. Analiza ogłoszeń stanowiących próby oszustwa prowadzi do wniosku, iż często zawierają one szablony, które są powielane w kolejnych wrzucanych na portal ogłoszeniach, z niewielkimi zmianami np. nową nazwą stanowiska. Celem identyfikacji tego typu spraw wykorzystano metodę badania podobieństwa dwóch tekstów z bilbioteki spaCy. Jest ona oparta na wytrenowanych wcześniej wektorach i zwraca wartość pomiędzy 0 a 1. Wektory są też wrażliwe pod kątem semantyki np. "I'd like to eat a burger" zwróci większą wartość podobieństwa z "I'd like to eat a sandwich" niż to samo zdanie zawierające niezwiązany tematycznie rzeczownik np. "car". Kod programu umożliwia wybranie punktu odcięcia, który będzie skutkował wartością "1" dla zmiennej. W przykładzie przyjęto 99%. Taki próg gwarantuje, że oznaczone zostaną obserwacje faktycznie bardzo zbliżone do któregoś ze zidentyfikowanych wcześniej oszustw, wręcz różniące się tylko pojedynczymi słowami. Co do zasady teksty dotyczące podobnego tematu będą bowiem generować dość wysokie podobieństwo.

### 3.4.4 Worek słów

Jest to etap najbardziej wykorzystujący zasoby pamięci. Na tym etapie w potoku NLP powstają dwa zbiory danych zawierające jako kolumny wszystkie lematy występujące w zbiorze uczącym. Kod programu pozwala na wybór metody przypisywania wartości słowom. Dla omawianego problemu najlepsze rezultaty zwrócił scoring binarny oraz TF-IDF. Ta druga metoda została finalnie wykorzystana. Stworzona w ten sposób reprezentacja tekstu służy następnie jako dane wejściowe dla modeli uczenia maszynowego.

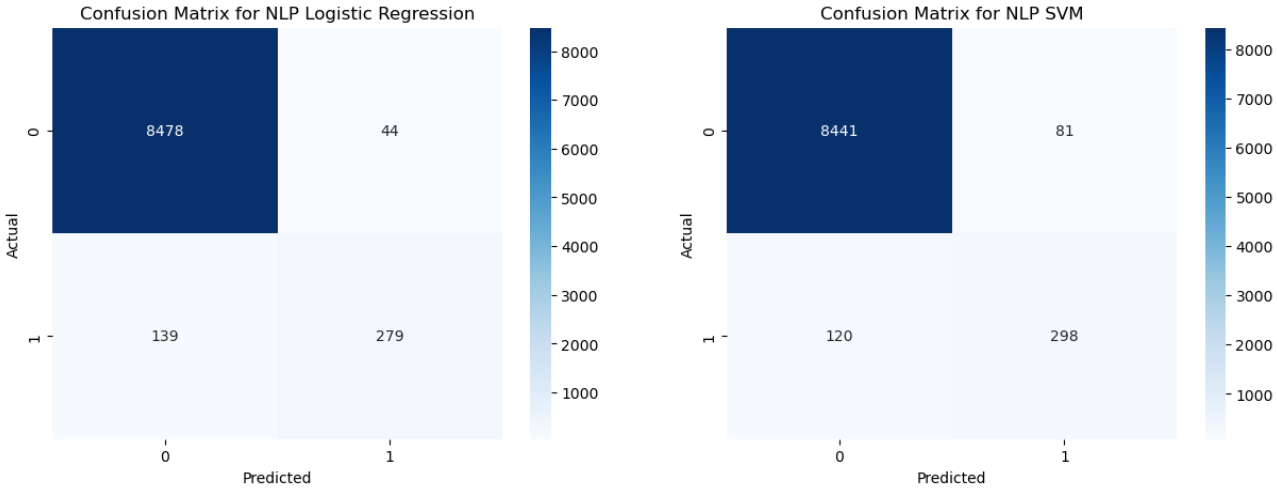
Z uwagi na ilość zmiennych zdecydowano się na modele liniowe: regresję logistyczną oraz SVM. W fazie testowania rozwiązań modele oparte na drzewach decyzyjnych zwracały znacznie gorsze wyniki. Przed trenowaniem modeli dostępna jest możliwość przetwarzania wektorów cech na kilka sposobów. Kod programu umożliwia usunięciu ze zbioru uczącego zarówno obserwacji o liczbie słów mniejszej niż wskazana, jak i słów o niewielkiej liczebności. Ma to celu zapobieżenie ewentualnemu zbytniemu ciążeniu na wynikach modelu. Opcjonalnie można ograniczyć słowa tylko do tych występujących w rzędach o poszukiwanej wartości zmiennej celu. W omawianym przykładzie opcje te nie zostały jednak wykorzystane.

Jako dalsze kroki zaprogramowano oversampling oraz selekcję zmiennych metodą k najlepszych. Kod jest elastyczny pod względem zarówno proporcji zwiększenia kategorii, jak i doboru ilości zmiennych i wyboru kryterium selekcji do modelu. W przykładzie posłużono się wartością 5000 lematów, która na etapie eksperymentów zwracała dobre wyniki, oraz metryką chi kwadrat. Ilość zmiennych w całym zbiorze bag-of-words wynosi ponad 15 tysięcy.

Przetworzone w ten sposób dane są podawane na wejście dwóch wskazanych wcześniej modeli, które mogą zostać poddane optymalizacji hiperparametrów. Również w tym miejscu można zdefiniować metrykę jakości modelu, która ma być optymalizowana. Analogicznie i konsekwentnie do wcześniejszych rozważań na temat modelu bazowego zdecydowano się na czułość, która jest celem optymalizacji w finalnym modelu.

### 3.4.5 Modele oparte na reprezentacji worka słów

Jako zmienne do zbioru docelowego dodawane są wyniki prawdopodobieństwa zwrócone przez oba modele. W celu oceny ich ewentualnej przydatności kalkulowane są także metryki ich jakości. Zbiór docelowy jest w tym przypadku traktowany jak testowy. Zastrzec jednak należy, że wartości te dotyczą de facto tylko dwóch z dodawanych do kompleksowego modelu zmiennych. Ich ocena ma więc na celu tylko odpowiedź na pytanie o czy tekst przyniósł nam informację, która może być użyteczna, jako wartość opisująca swoiste "ryzyko wyłudzenia wynikające z modelowania tekstu", w szerszym kontekście. Analiza matryc pomyłek oraz metryk jakości prowadzi do optymistycznych wniosków i wskazuje, że dodanie wszystkich zmiennych do finalnego modelu powinno znacząco wpłynąć na jego ewaluację.




*Rys. 14. Matryce pomyłek modeli worka słów (źródło: opracowanie własne)*

W wyniku działania potoku do modelu finalnego zostanie dodanych sześć nowych zmiennych:

| Nazwa | Opis | Rodzaj zmiennej | Braki |
|---|---|---|---|
| is_similar | Czy zmienna tekstowa jest podobna do zidentyfikowanego oszustwa? | binarna | - |
| failed_perc | Odsetek błędnie zapisanych słów | ciągła (0-1) | - |
| unique_perc | Odsetek słów spoza słownika | ciągła (0-1) | - |
| word_count | Liczba słów | ciągła | - |
| score_logreg | Wartość prawdopodobieństwa z modelu regresji logistycznej | ciągła (0-1) | - |
| score_svm | Wartość prawdopodobieństwa z modelu SVM | ciągła (0-1) | - |

*Tab. 4. Opis zmiennych wynikowych potoku NLP (źródło: opracowanie własne)*

## 3.5 Model rozszerzony

Następnym krokiem jest wprowadzenie zmiennych wygenerowanych przez potok NLP do modelu. Na początku, aby umożliwić porównanie z utworzonym wcześniej modelem bazowym, do danych uzupełnionych o zmienne z tekstu aplikowany jest podział na zbiory testowy i treningowy według tych samym proporcji oraz oversampling o identycznym parametrze.

### 3.5.1 Przetwarzanie danych

Jako pierwsze głębszej analizie poddane zostaną zmienne ciągłe. Użyteczność wyników modeli worka słów została potwierdzona wcześniej. Problem może sprawiać jednak pytanie: czy wykorzystanie obu zmiennych jednocześnie nie powoduje przeciążenia wyniku? Obie dotyczą bowiem tej samej informacji, czyli ryzyka oszustwa, które niosą ze sobą wykorzystane w tekście słowa. Temat ten może rodzić więc pewne kontrowersje natury metodologicznej. Ponieważ w modelu bazowym najlepiej wypadły algorytmy oparte na drzewach decyzyjnych, zdecydowano się na pozostawienie obu zmiennych. Możliwe są bowiem nieliniowe zależności skutkujące różnym stopniem wykorzystania tych zmiennych przez poszczególne podziały stanowiące odrębne gałęzie drzewa.

Zgoła inna konkluzja wiąże się ze zmiennym oznaczającymi odsetek słów spoza słownika i błędnych. Ocena oparta na stworzonych wykresach gęstości rozkładu dla kategorii zmiennej celu prowadzi do wniosku, że przydatność tych zmiennych w modelu jest wątpliwa. Obie kategorie skupiają się wokół tych samych wartości. Zmienne zostały więc wyłączone ze zbioru, intuicja stojąca za ich utworzeniem nie potwierdziła

się dla modelowanego zjawiska. Nie jest jednak wykluczone, że zespół będący odbiorcą wyników z modelu mógłby je wykorzystywać do identyfikowania i analiza obserwacji odstających.

W przypadku ilości wykorzystanych słów wykres gęstości wykazuje różnice w rozkładzie. Wartości dla ofert autentycznych skupiają się wokół większych wartości a pojedyncze obserwacje osiągają bardzo dużą liczbę słów, praktycznie niewystępującą w stwierdzonych oszustwach. Zmienna zostanie jednak poddana standaryzacji, aby wysokie wartości nie zaburzały wyniku predykcji.

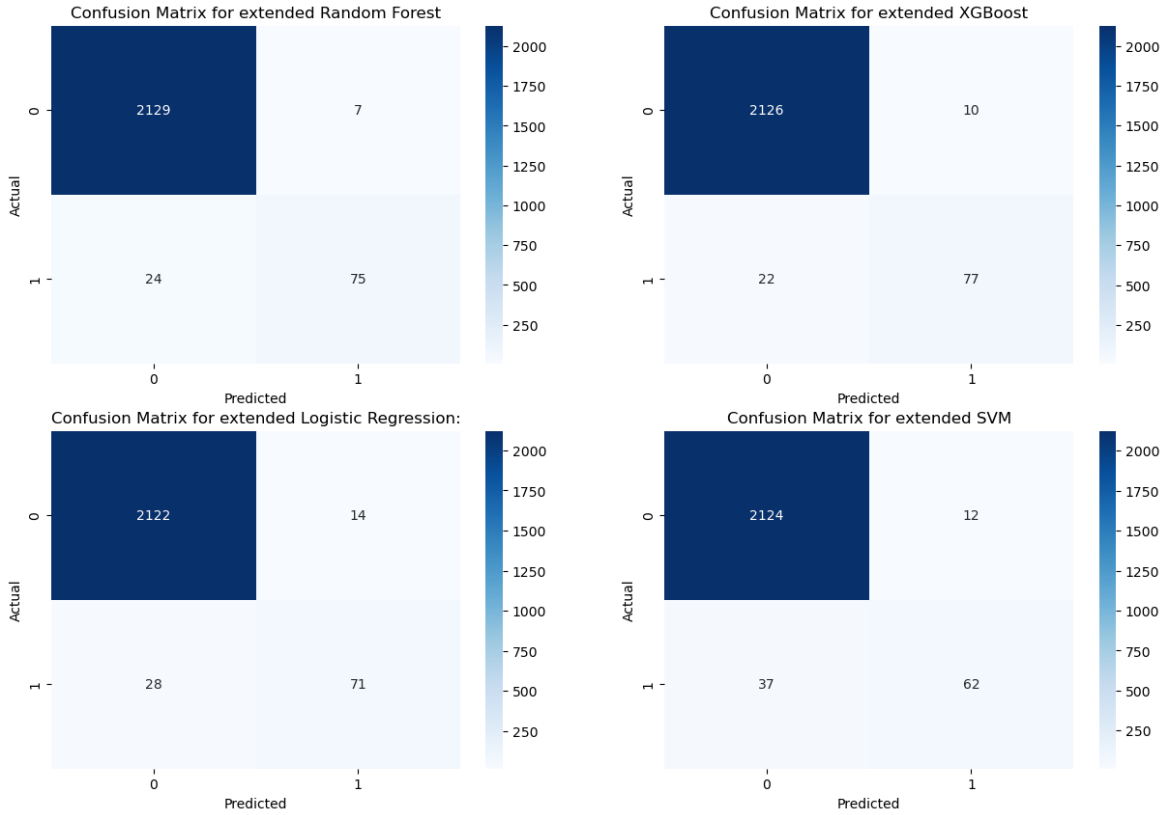
Osobną kwestią jest zmienna określająca podobieństwo do spraw oznaczonych w zbiorze uczącym. Dla jej oceny porównano średnie wartości zmiennej celu w obu kategoriach. Już takie spojrzenie pozwala na potwierdzenie jej mocy predykcyjnej. W przykładzie różnica tych dwóch wartości wynosi około 17 pkt. proc.

### 3.5.2 Budowa i ocena modelu

Aby najlepiej odnieść się do zasadniczego pytania stojącego za tematem pracy poszczególne modele są domyślnie trenowane na tych samych parametrach co w modelu bazowym. Umożliwia to porównanie wyników. Ponieważ dochodzą nowe zmienne nie jest wykluczone, że optymalizacja umożliwiłaby podbicie wartości metryk. Kod programu daje taką możliwość.

| Models | Accuracy_em | Precision_em | Recall_em | F1_em | AUC_ROC_em |
|---|---|---|---|---|---|
| Random Forest | 0.986130 | 0.914634 | 0.757576 | 0.828729 | 0.877149 |
| XGBoost | 0.985682 | 0.885057 | 0.777778 | 0.827957 | 0.886548 |
| Logistic Regression | 0.981208 | 0.835294 | 0.717172 | 0.771739 | 0.855309 |
| SVM | 0.978076 | 0.837838 | 0.626263 | 0.716763 | 0.810322 |

*Tab. 5. Metryki jakości modeli rozszerzonych (źródło: opracowanie własne)*




*Rys. 15. Macierz pomyłek modeli rozszerzonych (źródło: opracowanie własne)*

Ocena jakości predykcyjnej modelu rozszerzonego o zmienne z tekstu, a także jej zestawienie z metrykami modelu bazowego, jednoznacznie potwierdza główną tezę pracy. Wartości poszczególnych metryk znacznie wzrosły dla wszystkich modeli. W kontekście potencjalnego wykorzystania modelu warto odnotować istotne wzrosty czułości. Dzięki temu poszczególne algorytmy zwracają więcej trafionych spraw niż w modelu bazowym. Przeliczenie różnic na procenty pogłębia jedynie ogólne pozytywne wrażenie, wzrosty są bowiem kilkudziesięcioprocentowe. Wciąż najlepsze wartości metryk osiągają modele oparte na drzewach decyzyjnych. Modele liniowe, w szczególności SVM, wcześniej najsłabszy, zanotowały jednak największe wzrosty i w takiej formie również mogłyby znaleźć praktyczne wykorzystanie.

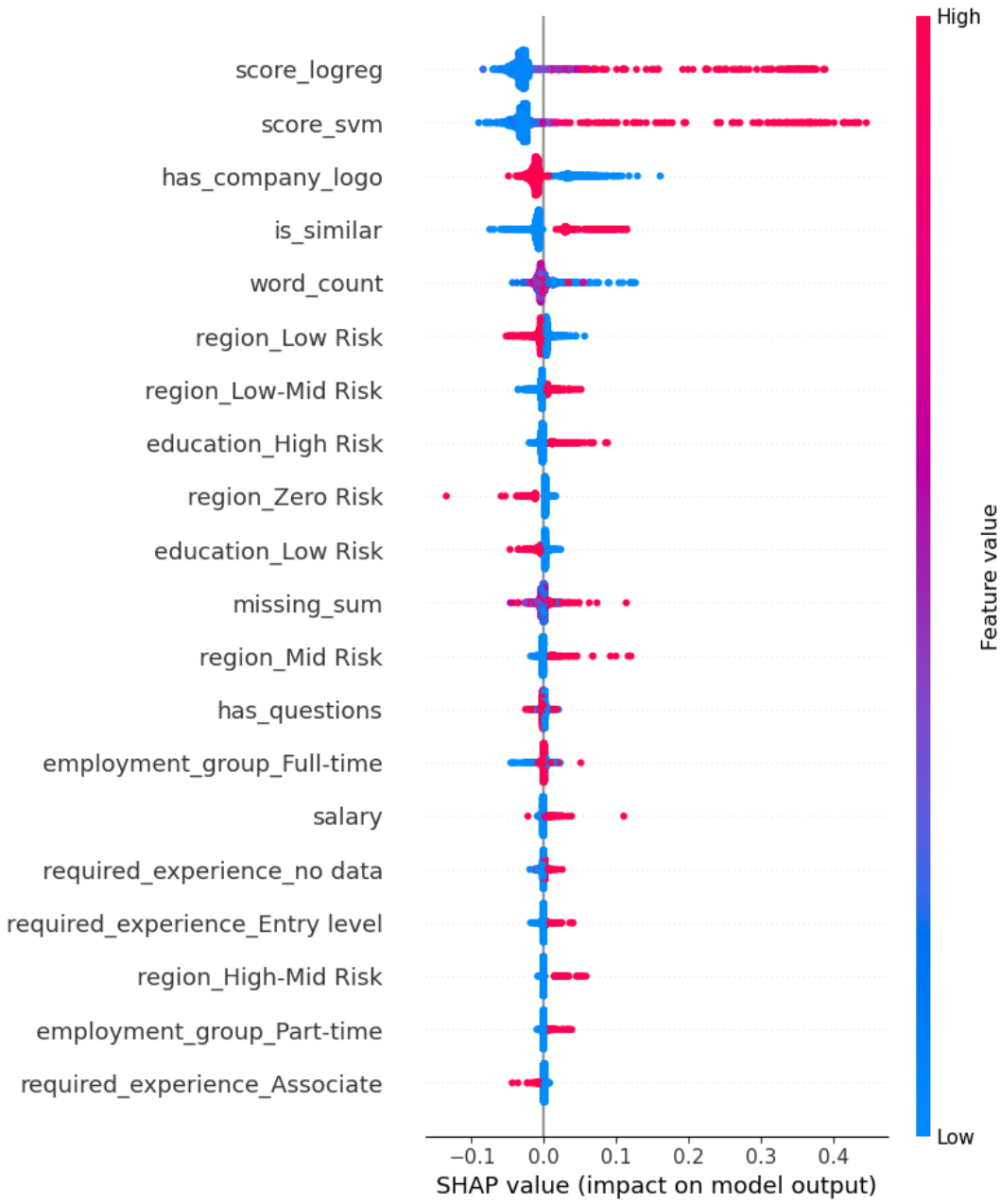
| Models | change_Accuracy | change_Precision | change_Recall | change_F1 | change_AUC_ROC |
|---|---|---|---|---|---|
| Random Forest | 3.086997 | 82.926829 | 56.350741 | 68.390737 | 19.947701 |
| XGBoost | 3.040225 | 77.011494 | 50.888889 | 63.107527 | 18.834045 |
| Logistic Regression | 3.345900 | 104.183007 | 93.237935 | 98.294082 | 27.012317 |
| SVM | 2.054155 | 41.787942 | 367.288267 | 228.056914 | 43.443863 |

*Tab. 6. Zmiana parametrów modeli po wprowadzeniu nowych zmiennych w ujęciu procentowym (źródło: opracowanie własne)*

| Models | delta_Accuracy | delta_Precision | delta_Recall | delta_F1 | delta_AUC_ROC |
|---|---|---|---|---|---|
| Random Forest | 0.029530 | 0.414634 | 0.273040 | 0.336583 | 0.145873 |
| XGBoost | 0.029083 | 0.385057 | 0.262314 | 0.320343 | 0.140509 |
| Logistic Regression | 0.031767 | 0.426203 | 0.346038 | 0.382550 | 0.181903 |
| SVM | 0.019687 | 0.246929 | 0.492242 | 0.498276 | 0.245417 |

*Tab. 7. Zmiana parametrów modeli po wprowadzeniu nowych zmiennych w ujęciu bezwzględnym (źródło: opracowanie własne)*

Warto odnotować także obecność dodanych zmiennych wśród cech z najwyższymi wartościami SHAP. Kolejność jest różna w zależności od modelu, co tylko uzasadnia, że warto nie ograniczać się w próbach definiowana nowych sposobów spojrzenia na dane. Kreatywność w znajdowaniu nowych cech i różnorodność stosowanych metod ich pozyskiwania może przekładać się bezpośrednio na finalną ocenę wartości predykcyjnej. W zależności od wybranego rozwiązania różna może być bowiem użyteczność poszczególnych zmiennych. W rozszerzonym modelu lasu losowego w czołowej piątce są aż cztery zmienne pozyskane z tekstu. W szczególności cieszy tu obecność zmiennej binarnej oznaczającej występowanie prawdopodobieństwa z inną obserwacją-oszustwem. W modelu XGBoost czy regresji logistycznej nie okazała się jednak tak ważna.




*Rys. 16. Wartości SHAP dla zmiennych w rozszerzonym modelu lasu losowego*

Reasumując, metody z zakresu NLP, nawet te prostsze, niewymagające wiedzy z zakresu uczenia głębokiego, mogą być narzędziem do pozyskiwania cennych informacji o analizowanych zjawiskach. Warto spojrzeć na tekst pod kątem możliwości wyodrębnienia z niego nowych danych. Wydaje się też uzasadnione stwierdzenie, że warto pod tym kątem eksperymentować. Stopień użyteczności zmiennych może być zapewne różny w zależności od zagadnienia. Jednak nawet nieznaczny wzrost jakości predykcyjnej modelu może w dłuższym okresie przełożyć na efektywność pracy jego odbiorców się i, w konsekwencji, na wyniki finansowe organizacji. W zaprezentowanym przykładzie bez wątpliwości dowiedziono zasadność rozszerzenia wykorzystywanych w organizacji metod z zakresu Data Science o NLP, pomimo pewnych trudności, które mogą się z tym wiązać. Żeby zrozumieć szerszy kontekst tego zagadnienia, warto także

zapoznać się z jego teoretyczną podstawą, czyli informacjami z zakresu nauki o języku. Mogą one być inspirujące w procesie eksploracji danych tekstowych. Ogólny wniosek z pracy bez wątpienia można rozszerzyć także na obszar maszynowego uczenia nienadzorowanego i inne potencjalne wykorzystania tekstu.

# Bibliografia


Ahmad, Imran. (2021). *40 algorytmów, które powinien znać każdy programista*. Gliwice: Helion

Bish, Alex, Hannett, Josie, Irving, Fiona. (2024). Fake Jobs: Scammers impersonate firms to target victims. W: BBC. Pobrano z: https://www.bbc.com/news/uk-england-surrey-68110626 (dostęp: 02.06.2024)

Bruce, Peter, Bruce, Andrew, Gedeck, Peter. (2021). *Statystyka praktyczna w data science*. Gliwice: Helion

Downey, Allen. (2015). *Think Python. How to Think Like a Computer Scientist*. Needham, Massachusetts: Green Tea Press

European Parliament/EU AI Act: first regulation on artificial intelligence. (2023). Pobrano z: https://www.europarl.europa.eu/topics/en/article/20230601STO93804/eu-ai-act-first-regulation-on-artificial-intelligence (dostęp 22.05.2024)

Foster, David. (2021). *Deep learning i modelowanie generatywne*. Gliwice: Helion

Jurafsky, Daniel, Martin, James H. (2023). *Speech and Language Processing. 3rd Edition Draft*. Pobrano z: https://oreil.ly/Ta16f (dostęp: 30.05.2025)

Kaggle/Real / Fake Job Posting Prediction. 2020. Pobrano z: https://www.kaggle.com/datasets/shivamb/real-or-fake-fake-jobposting-prediction (dostęp 15.05.2024)

Kahneman, Daniel. (2012). *Pułapki myślenia, o myśleniu szybkim i wolnym*. Poznań: Media Rodzina

Litorowicz, Michał. (2024). *Algorytmem w samotne matki. Na aferze zasiłkowej w Holandii wyłożyli się nie tylko politycy*. W: Krytyka Polityczna. Pobrano z: https://krytykapolityczna.pl/swiat/ue/holandia-skandal-zasilkowy-bledy-algorytmu/ (dostęp 22.05.2024)

NLTK/Documentation. (2023). Pobrano z: https://www.nltk.org/api/nltk.tokenize.html (dostęp 28.05.2024)

PAP Biznes/W perspektywie najbliższych kilku lat tydzień pracy należy skrócić do czterech dni - Dziemianowicz-Bąk. (2024). Pobrano z: https://biznes.pap.pl/en/news/info/3594860,w-perspektywie-najblizszych-kilku-lat-tydzien-pracy-nalezy-skrocic-do-czterech-dni--dziemianowicz-bak (dostęp 23.05.2024)

Patil, Aakanksha. (2023). *Top 5 Pre-trained Word Embeddings*. W: Medium. Pobrano z: https://patil-aakanksha.medium.com/top-5-pre-trained-word-embeddings-20de114bc26 (dostęp 08.06.2024)

Pracuj.pl/Raport Rynek Pracy Specjalistów 2023. Rekrutacja w obliczu gospodarczych wyzwań. (2024). Pobrano z: https://media.pracuj.pl/284949-raport-rynek-pracy-specjalistow-2023-rekrutacja-w-obliczu-gospodarczych-wyzwan (dostęp 02.06.2024)

Provost, Foster, Fawcett, Tom. (2023). *Analiza danych w biznesie*. Gliwice: Helion

*REL – Młodzieżowym Słowem Roku 2023!*. (2023). W: Słownik Języka Polskiego PWN. Pobrano z: https://sjp.pwn.pl/mlodziezowe-slowo-roku/haslo/REL-Mlodziezowym-Slowem-Roku-2023;9286107.html (dostęp 25.05.2024)

SHAP/*Welcome to SHAP Documentation*. (2018). Pobrano z: https://shap.readthedocs.io/en/latest/ (dostęp: 04.06.2024)

spaCy/Linguistic Features. (2024). Pobrano z: https://spacy.io/usage/linguistic-features (dostęp 20.05.2024)

Stanford Encyclopedia of Philosophy/Ambiguity. (2011). Pobrano z: https://plato.stanford.edu/entries/ambiguity/ (dostęp 25.05.2024)

Tapper, James. (2023). *Authors shocked to find AI ripoffs of their books being sold on Amazon*. W: Guardian. Pobrano z: https://www.theguardian.com/technology/2023/sep/30/authors-shocked-to-find-ai-ripoffs-of-their-books-being-sold-on-amazon (dostęp 27.05.2024)

Treveil, Mark, Omont, Nicolas, Stenac, Clémen, Lefevre, Kenji, Phan, Du, Zentici, Joachim, Lavoillotte, Adrien, Miyazaki, Makoto, Heidmann, Lynn. (2020). *Introducing MLOps*. O'Reilly Media, Inc.

Vajjala, Sowmya, Majumder, Bodhisattwa, Gupta, Anuj, Surana, Harshit. (2023), *Przetwarzanie języka naturalnego w praktyce*. Gliwice: Helion

Vidros, Sokratis, Kolias, Constantinos, Kambourakis, Georgios, Akoglu, Leman. (2017). *Automatic Detection of Online Recruitment Frauds: Characteristics, Methods, and a Public Dataset*. W: Future Internet. Pobrano z: https://www.mdpi.com/1999-5903/9/1/6 (dostęp 22.05.2024)

Wikipedia/Lemma (mathematics). (2024). Pobrano z: https://en.wikipedia.org/wiki/Lemma_(mathematics) (dostęp 28.05.2024)

Wikipedia/Natural Language Processing. (2024). Pobrano z: https://en.wikipedia.org/wiki/Natural_language_processing (dostęp 26.05.2024)

Wikipedia/Phoneme. (2024). Pobrano z: https://en.wikipedia.org/wiki/Phoneme (dostęp 25.05.2024)

Wikipedia/String (computer science). (2024). Pobrano z: https://en.wikipedia.org/wiki/String_(computer_science) (dostęp 25.05.2024)

Wikipedia/Text (literary theory). (2024). Pobrano z: https://en.wikipedia.org/wiki/Text_(literary_theory) (dostęp 25.05.2024)

# Spis rysunków i tabel

**Spis rysunków:**



**Spis tabel:**