Marek Troszyński
Collegium Civitas

Aleksander Wawer
Instytut Podstaw Informatyki PAN

https://doi.org/10.18778/1733-8069.13.2.04

Czy komputer rozpozna hejtera? Wykorzystanie uczenia maszynowego (ML) w jakościowej analizie danych

Abstrakt Celem artykułu jest przedstawienie procesu automatyzacji kodowania tekstów pochodzących z mediów społecznościowych. Wdrożenie tego procesu pozwala na ilościowe potraktowanie jakościowych metod analizy treści. W efekcie otrzymujemy możliwość przeprowadzenia analizy na korpusach liczących setki tysięcy tekstów, które są kodowane w oparciu o ich znaczenia. Jest to możliwe dzięki wykorzystaniu algorytmów uczenia maszynowego (ML).

Omawianą metodę kodowania prezentujemy na przykładzie projektu oznaczania „mowy nienawiści" w tekstach pochodzących z polskich forów internetowych. Kluczowym problemem jest precyzyjna konceptualizacja i operacjonalizacja tej kategorii. Pozwala to na przygotowanie dokładnej instrukcji kodowej oraz przeprowadzenie treningu zespołu kodującego. Efektem jest podwyższenie współczynnika zgodności kodujących. Oznaczone teksty zostaną wykorzystane jako dane treningowe dla metod automatycznej kategoryzacji opartych o algorytmy uczenia maszynowego. W dalszej części artykułu opisujemy zastosowane metody kodowania automatycznego. Tekst kończy podsumowanie wskazujące na czynniki, które są kluczowe dla procesu badawczego wykorzystującego uczenie maszynowe.

Słowa kluczowe jakościowa analiza treści, uczenie maszynowe, mowa nienawiści, zgodność kodujących

Marek Troszyński, doktor socjologii, kierownik Obserwatorium Cywilizacji Cyfrowej Collegium Civitas, adiunkt tamże. Zainteresowania naukowe: socjologia kultury, wykorzystanie metod automatycznej analizy języka naturalnego (NLP) w socjologicznych badaniach nad dyskursem.

Adres kontaktowy:

Collegium Civitas
Plac Defilad 1
00-901 Warszawa
e-mail: mtroszynski@civitas.edu.pl

Aleksander Wawer, doktor nauk technicznych w kierunku informatyka, absolwent socjologii i informatyki. Adiunkt w Zespole Inżynierii Lingwistycznej w Instytucie Podstaw Informatyki PAN. Zainteresowania naukowe obejmują wybrane problemy przetwarzania języka naturalnego, w szczególności analizę wydźwięku, ekstrakcję relacji oraz głębokie uczenie maszynowe.

Adres kontaktowy:

Instytut Podstaw Informatyki PAN
ul. Jana Kazimierza 5, 01-248 Warszawa
e-mail: axw@ipipan.waw.pl

Gwałtowny rozwój komunikacji zapośredniczonej przez komputer, a w szczególności tak zwanych mediów społecznościowych, sprawia, że w dziedzinie badań nad treścią przekazów medialnych obserwujemy znaczące rozszerzenie dostępnego obszaru badawczego. W przestrzeni publicznej pojawił się zbiór tekstów pisanych przez tradycyjnie rozumianych „odbiorców" (tzw. *User Generated Content* – UGC). Badacze stanęli wobec istotnego problemu – jak analizować teksty tworzone przez nieprofesjonalistów, które charakteryzują się różnorodnością języka, stylów wypowiedzi, konwencji, wykorzystywanych socjolektów, gwary czy sformułowań potocznych. Co ważniejsze, są to wypowiedzi liczone w setkach tysięcy czy milionach tekstów.

W odpowiedzi na ten problem chcemy przedstawić metodę analizy danych (analizy treści), która, trwając przy jakościowym podejściu do kodowania (przypisywanie kodów w oparciu o znaczenia zawarte w tekście), pozwala jednocześnie na ilościowy opis dużego korpusu różnorodnych tekstów (np. zapisu dyskursu z mediów społecznościowych). Metoda została wypracowana w trakcie projektu badawczego, którego celem było oznaczenie mowy nienawiści wobec mniejszości (narodowych, etnicznych, seksualnych) w polskim Internecie.

We wspomnianym projekcie, chcąc monitorować mowę nienawiści, musieliśmy przygotować narzędzia badawcze, które pozwolą na skuteczne oznaczanie wielkiej liczby tekstów. Ponieważ „mowa nienawiści" jest złożonym konstruktem teoretycznym, a jej reprezentacje językowe mogą przyjmować niezliczone formy, nie sposób przeprowadzić analiz ilościowych bazujących na frekwencjach i współwystępowaniu poszczególnych słów. Analiza jakościowa zakłada mozolne kodowanie znaczeń (w tym ładunku emocjonalnego), co w przypadku zbioru kilku tysięcy tekstów również każe nam stawiać pytanie o wykonalność tego zadania. Rozwiązaniem jest połączenie jakościowego kodowania z narzędziami udostępnionymi przez lingwistykę komputerową i metody automatycznego przetwarzania języka naturalnego (*Natural Language Processing* – NLP).

NLP a nauki społeczne

Narzędzia komputerowej, jakościowej (również ilościowej) analizy tekstów wykorzystywane są w socjologii od kilku dziesięcioleci. Początki sięgają pionierskich programów komputerowych dla języka angielskiego, korzystających ze słowników (General Inquirer [Stone i in. 1966] czy LIWC [2007]). W języku polskim wymienić można w tym kontekście słowniki wydźwięku stworzone dla języka polskiego (Wawer, Rogozińska, 2012).

Współczesne podejście jest nieco inne. Zamiast tworzyć słowniki, które operacjonalizują pożądane kategorie badawcze, tworzy się raczej kolekcje tekstów z oznaczeniami (kodowaniem) wybranych zjawisk. Mogą to być oznaczenia na poziomie wyrazów, wielowyrazowych fraz, zdań lub większych fragmentów tekstu – odpowiednio do tego, na jakim poziomie leksykalnym i składniowym dane zjawisko występuje. Następnie, wykorzystując owe kolekcje oznaczonych w tekstach problemów, próbuje się stworzyć komputerowe modele tych zjawisk, korzystając z narzędzi uczenia maszynowego (ML)

(Bishop 2007). Skuteczność (przykładowo: precyzja) narzędzi tworzonych w ten sposób jest wyższa niż metod słownikowych. Pozwalają one na uchwycenie nie tylko koncepcji i znaczeń kodowanych na poziomie słów, ale także ciągów słów, fraz, zdań i całych dokumentów. Wszystkie te metody zgrupowane są pod zbiorczą nazwą uczenia maszynowego z nadzorem.

W przypadku większości metod wymagane jest wstępne przetwarzanie tekstu w celu otrzymania odpowiedniej reprezentacji. Liczba kroków i poziom przetworzenia zależą od algorytmu rozpoznającego pożądane zjawisko. Z reguły jest to podział na słowa (tokenizacja), rozpoznawanie części mowy oraz przetwarzanie składniowe, płytkie lub głębokie. Zestawy narzędzi tego typu zawiera pakiet Stanford NLP (Manning i in. 2014) oraz dla języka polskiego – Multiserwis, utrzymywany w Instytucie Podstaw Informatyki PAN (Ogrodniczuk, Lenart 2013).

Analiza na poziomie frazowym przeprowadzana jest zazwyczaj z wykorzystaniem metody warunkowych pól losowych (ang. *Conditional Random Fields*) (Lafferty, McCallum, Pereira 2001), ostatnio także głębokich sieci neuronowych (Lample i in. 2016). W metodach tych jest brana pod uwagę kolejność występowania słów w tekście oraz kontekst zdaniowy. Są one do pewnego stopnia odporne na zmienny szyk składniowy. Metody te są z sukcesem używane w takich zastosowaniach jak rozpoznawanie dłuższych niż jeden wyraz fragmentów tekstu określonego typu, jak przykładowo frazy zawierające wydźwięk (ang. *sentiment*). Istnieje możliwość zastosowania tych metod również do innych problemów związanych z jakościową analizą tekstu, zwłaszcza takich, w których kodowane przez badaczy (potencjalnie lingwistów) zjawiska i treści mają charakter wielowyrazowy, silnie zależny od kontekstu. Sprawdzą się tam, gdzie nie jest możliwe zastosowanie prostego podejścia słownikowego, czyli takiego, w którym wszystkie znane z góry wielowyrazowe ciągi wpisane są na stałe do słownika, a każde ich wystąpienie w tekście jest tożsame z wystąpieniem badanego zjawiska.

Do rozpoznawania zjawisk na poziomie zdań oraz większych fragmentów tekstu stosowane są metody klasyfikacji z nadzorem oparte na reprezentacji *bag-of-words*, czyli niebiorącej pod uwagę kolejności słów w tekście, a tylko sam fakt ich wystąpienia. Algorytmy tego typu to przykładowo maszyny wektorów podpierających (ang. *Support Vector Machines*, SVM) (Cortes, Vapnik 1995) lub metody oparte na drzewach lub lasach drzew decyzyjnych (Breiman 2001). Przykładowe zastosowania obejmują klasyfikację tematyczną tekstów (czyli przykładowo stwierdzenie, czy tekst dotyczy sportu czy może polityki), analizy stylometryczne (identyfikacja różnego typu cech psychologicznych lub demograficznych piszącej osoby).

Istnieje wiele obszarów badawczych na gruncie szeroko rozumianych nauk społecznych, w których wykorzystywane są komputerowe narzędzia przetwarzania języka naturalnego. W przypadku języka angielskiego, którego dotyczy większość prowadzonych badań, można mówić o takich generalnych kierunkach jak przewidywanie atrybutów osób piszących teksty (przykładowo ich emocje, płeć, wiek, przekonania polityczne i system wartości) czy przewidywanie zjawisk społecznych na podstawie zebranych tekstów (przykładowo wyniki wyborów, epidemie chorób, notowania giełdowe).

Interesującym obszarem zastosowania komputerowego przetwarzania tekstu są badania literackie i stylometryczne. W paradygmacie tym metody komputerowe i algorytmy służą jako narzędzie poznawcze, agregujące i syntetyzujące informacje z dużych zbiorów tekstowych danych literackich. Używane są między innymi informacje statystyczne i frekwencyjne, o współwystępowaniu pojęć oraz określonego typu słów. Analiza taka pozwala na poznanie struktur charakterystycznych dla określonego okresu lub gatunku literackiego i uzyskanie wglądów zupełnie innych niż lektura poszczególnych pozycji. Możliwa jest zwłaszcza eksploracja trendów w czasie lub różnic geograficznych (Jockers 2013). W analizach tego typu wykorzystywane są też wizualizacje współwystąpień słów i sieci pojęć (Moretti 2013).

Wykorzystanie NLP nie zwalnia badaczy społecznych z odpowiedzialności za proces badawczy, w szczególności za jakość kodowania tekstu. Na przykładzie badań nad mową nienawiści chcemy pokazać szerszy problem metodologiczny, który zamyka się w pytaniu – jak można zoptymalizować działania zespołu badawczego (osób odpowiedzialnych za kodowanie tekstów), aby próbować wykorzystać efekty takiego kodowania do „nauczenia" narzędzia (skryptu), które z określoną precyzją oznaczy w ten sposób dowolnie duży zbiór tekstów.

W trakcie trwających od 2012 roku projektów badawczych testowaliśmy różne metody automatyzacji kodowania tekstów zawierających mowę nienawiści, by oszacować zasięg zjawiska i tematy poszczególnych wypowiedzi (Troszyński 2015). Ostatecznie przyjęliśmy metodę określoną przez nas jako półautomatyczna – przeszkolony zespół koderów/anotatorów[1] kodował fragmenty tekstu (przypisując im określone kategorie, np. poziom negatywnych emocji na skali 0–4), by przejść do budowania algorytmu, który automatyzuje tę czynność, wykorzystując techniki uczenia maszynowego. Efekt działań algorytmów był ponownie weryfikowany przez zespół kodujący.

Przedmiot badania – mowa nienawiści

Kluczowy dla zrozumienia złożoności prezentowanego zagadnienia jest przedmiot badań – mowa nienawiści. Mamy tu konstrukt, który nie dość, że jest wytworem określonej teorii, to dodatkowo jest opisywany w różny sposób w różnych dyskursach kulturowych. Spójrzmy na wybrane definicje tej kategorii pisane z perspektywy prawa, językoznawstwa i socjologii.

Najwięcej uwagi kategorii „mowy nienawiści" poświęca się w obrębie nauk prawnych (Wieruszewski i in. 2010; Siwicki 2011; Bychawska-Siniarska, Głowacka 2013; Heinze 2016). W polskim systemie prawnym nie istnieje powszechnie przyjęta definicja tej kategorii. Jednym z dokumentów obecnych w obiegu prawnym jest raport Biura Studiów i Ekspertyz Kancelarii Sejmu RP, według którego „mowa nienawiści to wypowiedzi ustne i pisemne oraz

[1] Prezentowany tekst łączy perspektywy dwóch dyscyplin – lingwistyki i socjologii. Dlatego będziemy posługiwać się podwójną nazwą kluczowej dla tekstu funkcji w procesie badawczym – osoby, która oznacza teksty, przypisując im odpowiednie kategorie z instrukcji kodowej – anotator/koder.

przedstawienia ikoniczne lżące, oskarżające, wyszydzające i poniżające grupy i jednostki z powodów po części od nich niezależnych – takich jak przynależność rasowa, etniczna i religijna, a także płeć, preferencje seksualne, kalectwo (...). Jest to upubliczniona przemoc werbalna, wyraz nienawiści kolektywnej (...)" (Łodziński 2003: 5).

Próby podsumowania definicji mowy nienawiści z perspektywy systemu prawnego podjęły się Dominika Bychawska-Siniarska i Aleksandra Gliszczyńska-Grabias: „Mowa nienawiści, w potocznym rozumieniu, to słowa, wypowiedzi czy publikacje wyrażające **skrajnie negatywne, nienawistne uczucia i poglądy wobec określonych jednostek lub grup** [wyróżnienie własne]. Najczęściej wypowiedzi takie kojarzymy z rasizmem, ksenofobią, nienawiścią wobec mniejszości seksualnych lub nienawiścią wobec mniejszości religijnych" (2016: 4). Co ważniejsze, mowa nienawiści rozpowszechniana w Internecie nosi „pewne znamiona, których nie posiadają wypowiedzi w tzw. realu, takie jak trwałość, długoterminowość, powtarzalność, anonimowość oraz transgraniczny charakter" (Bychawska-Siniarska, Gliszczyńska-Grabias 2016: 5).

Pojęcie mowy nienawiści jest opisywane w perspektywie językoznawczej. Warto zwrócić uwagę na koncepcję Jadwigi Linde-Usiekniewicz (2015), która do opisu mowy nienawiści wykorzystuje teorię relewancji Sperbera i Wilsona (2011). Efektem jest definicja: „Mową nienawiści jest taka wypowiedź, dla której optymalnego przetwarzania niezbędne są przesłanki (czyli przekonania lub reprezentacje przekonań) dotyczące tego, że jakaś **grupa**, w tej wypowiedzi przywołana i zdefiniowana przez cechy uznawane za stałe, j**est gorsza pod jakimś względem, a przez to zasługuje na gorsze traktowanie** [wyróżnienie własne], i że osoba lub osoby, do których ta wypowiedź się odnosi, jest członkiem tej grupy" (Linde-Usiekniewicz 2015). Podobnie jak w przypadku refleksji prawniczej tak i w tej perspektywie podnoszona jest kwestia przeniesienia cech grupy na przynależące do niej jednostki.

I wreszcie perspektywa nauk społecznych. Temat ten podejmuje Lech Nijakowski, który tak określa to zjawisko: „mowa nienawiści polega na **przypisywaniu szczególnie negatywnych cech i/lub wzywaniu do dyskryminujących działań** [wyróżnienie własne], wymierzonych w pewną kategorię społeczną, przede wszystkim taką, do której przynależność jest postrzegana jako «naturalna» (przypisana), a nie z wyboru" (2008: 132). Podobnie jak w przypadku cytowanych wyżej definicji główny nacisk położony jest na przynależność do grupy oraz „przypisywanie negatywnych cech".

Próbując uwspólnić powyższe definicje, przyjęliśmy rozstrzygnięcia, które pozwoliły dookreślić przyjęty obszar badawczy:

1. wyodrębniliśmy dwie kategorie w obszarze szeroko pojmowanej mowy nienawiści: „mowę nienawiści" oraz „język wrogości". Utworzenie szerszej kategorii „języka wrogości" uznaliśmy za konieczne, by wyjść poza prawno-karne konotacje „mowy nienawiści",

2. przedmiotem analiz uczynimy mowę nienawiści skierowaną przeciw mniejszościom i członkom mniejszości (przynależność do określonej grupy). Oznacza to, że w badaniu pominęliśmy mowę nienawiści wobec polityków, grup większościowych, osób prywatnych.

Konceptualizując powyższe kategorie, przyjęliśmy, że **mowa nienawiści (w wąskim sensie)** to treści zawierające jawne wezwanie do działania oraz skrajnie negatywne emocje (wzmocnione często przez wulgaryzmy) skierowane przeciwko określonej grupie społecznej. W badaniu posługujemy się również pojęciem **język wrogości, który jest kategorią szerszą niż mowa nienawiści.** Określamy w ten sposób wszelkie treści odnoszące się negatywnie, nieprzychylnie lub odwołujące się do negatywnych emocji wobec określonych mniejszości. Mogą to być zarówno wypowiedzi prawdziwe, jak i fałszywe. Kluczowym elementem jest wykorzystanie w tekście mechanizmu stereotypu, który działa poprzez przypisywanie określonych cech i zachowań zbiorowości wszystkim jej członkom.

Pierwsze próby kodowania automatycznego – podejście słownikowe

Pierwszym zadaniem badawczym była budowa korpusu tekstów, które zostaną poddane analizie. W tym celu zostały zrealizowane następujące działania:

- wybraliśmy (w oparciu o dane o liczbie realnych użytkowników) portale horyzontalne, na których zostały wyodrębnione fora internetowe oraz komentarze, jako źródła tekstów do badania (gazeta.pl, wp.pl, onet.pl oraz interia.pl),
- wytypowaliśmy słowa kluczowe – określenia wybranych mniejszości (badanie dotyczyło mniejszości: muzułmańskiej, żydowskiej, LGBT, czeczeńskiej, romskiej, niemieckiej, rosyjskiej, ukraińskiej), w tym również określenia uznawane za obraźliwe.

Następnym krokiem było napisanie crawlera – narzędzia, które skanowało treści postów internetowych na wskazanych portalach i kopiowało do bazy danych te, które zawierały przynajmniej jedną nazwę mniejszości. Zebrane teksty oczyszczono poprzez usunięcie znaczników html i korektę pisowni. W tak opracowanych treściach zidentyfikowano fragmenty zawierające słowa klucze (określenia poszczególnych mniejszości) – zazwyczaj były to zdania lub mniejsze fragmenty wypowiedzi. Te fragmenty poddano dalszemu przetwarzaniu.

Przygotowaliśmy 2 korpusy tekstów[2]:

- 1500 tekstów pochodzących z postów zamieszczanych na forach internetowych, wybranych arbitralnie przez zespół kodujących, uwzględniając różne poziomy natężenia emocji wobec mniejszości – korpus ten wykorzystywany był jako korpus treningowy,

- 11 176 tekstów zebranych automatycznie przez crawler z forów internetowych; jego oznaczenie było głównym zadaniem dla ze-

[2] Przedmiotem analizy były „teksty" zbierane w sposób automatyczny – w pierwszej wersji przez dedykowany crawler, w kolejnych wersjach wykorzystując komercyjne narzędzia. Różne były metody określania długości relewantnego tekstu (odległość od słów kluczowych, spójność gramatyczna). Efektem były wypowiedzi zbliżone w formie i długości do zdania.

społu koderów/anotatorów biorącego udział w projekcie.

Oba korpusy bazowały na tekstach zamieszczonych w sieci w 2012 roku.

Pierwsza wersja narzędzia, którego celem było wykrywanie wśród postów języka wrogości, stworzona była na bazie hipotezy mówiącej, że język wrogości jest w istocie podzbiorem języka negatywnie nacechowanego (ang. *negative sentiment*). Aby zbudować to narzędzie, skorzystaliśmy z dostępnej wówczas wersji słownika wyrażeń negatywnie nacechowanych, tworzonego w Instytucie Podstaw Informatyki PAN (http://zil.ipipan.waw.pl/SlownikWydzwieku). Słownik ten został następnie zmodyfikowany w taki sposób, aby zawęzić go do podzbioru słów (w dużej części były to przymiotniki) i wyrażeń, który jest nacechowany zarówno negatywnymi, jak i wrogimi emocjami. Dzięki analizom korpusu okazało się, że konieczne jest poszerzenie tego zbioru negatywnych słów o słowa i wyrażenia w nim niewystępujące, a charakterystyczne dla języka polskiego Internetu. Dążyliśmy do tego, aby ten zbiór skonstruowany był w taki sposób, aby wystąpienie przynajmniej jednego elementu w danym tekście (poście) pozwalało na skategoryzowanie go jako zawierającego język wrogości.

Dla stworzenia narzędzia, które pozwoli na oznaczanie mowy nienawiści, zaczęliśmy od spisania wyczerpującego zbioru słów (ze względu na charakter szukanej treści są to przeważnie czasowniki oraz wszelkie formy wulgaryzmów), wyrażeń i reguł gramatycznych związanych z nimi, nawołujących do podjęcia działań wymierzonych przeciwko mniejszościom. Każdy z nich był opisywany przez analityka jako zawierający (1) lub niezawierający (0) wezwanie do działania. Następnie po analizie postów oznaczonych jako „1" wypisano słowa i wyrażenia, na podstawie których koderzy/anotatorzy kwalifikowali dany wpis jako mowę nienawiści. Po zebraniu zbioru słów i wyrażeń członkowie zespołu badawczego analizowali formy gramatyczne, w jakich występują poszczególne części wypowiedzi w postach uznanych wcześniej za zawierające wezwanie do działania. Zauważono, że o ile wyrażenia i słowa różne od czasowników występują zazwyczaj w stałych formach (np. „do gazu", „won", „wynocha", „Polska dla Polaków"), o tyle czasowniki przyjmują postać bezokoliczników (np. „zabić", „wykastrować") lub trybu rozkazującego (np. „wybijmy", „odizolujcie"). Wyrażenia wielowyrazowe (pojedyncze słowa są dostępne w zbiorze słów) oraz zdefiniowane gramatyczne formy czasowników zapisano w zbiorze reguł. Dzięki temu w ściśle zdefiniowanych sytuacjach syntaktycznych automatyczny analizator doda bądź usunie znacznik wezwania do działania z określonego fragmentu postu. Ponadto do mowy nienawiści włączane są posty zawierające wulgaryzmy.

Szukając sposobów na odejście od subiektywnych przekonań dotyczących występowania w poszczególnych tekstach negatywnych emocji (związanych z treściami obraźliwymi dla określonych mniejszości), przeprowadziliśmy sondaż, którego celem była ocena wybranych wypowiedzi przez większą grupę respondentów. Badanie było przeprowadzone przez Interaktywny Instytut Badań Rynkowych na próbie reprezentatywnej dla dorosłych internautów (N=800). Respondenci oceniali wyświetlane fragmenty tekstów (po 30 fragmentów dla każdego respondenta) pod kątem ich obraźliwości, wykorzystując skalę 0–4 (brak obraźliwości, bardzo obraźliwe)[3] (por. Troszyński 2015: 207). Analiza wyników badania miała pozwolić nam na intersubiektywizację oceny natężenia negatywnych emocji. Jednak zebrane dane wskazały na bardzo dużą rozbieżność w ocenianiu stopnia obraźliwości poszczególnych wypowiedzi. Pokazuje to, że celem tego typu badania powinno być raczej opisanie zmiennych (cech społeczno-demograficznych, przekonań politycznych itp.), które wpływają na „wrażliwość" na mowę nienawiści. Dzięki temu wyraźnie widzimy czynniki, które wpływają na postrzeganie mowy nienawiści, a co za tym idzie – na interpretację tekstu przez kodujących. W naszym projekcie zdecydowaliśmy się na stosowanie maksymalnie szerokiego rozumienia mowy nienawiści / języka wrogości poprzez bezpośrednie wpisanie do instrukcji kodowej (patrz niżej) postulatu podmiany przedmiotu wypowiedzi na własną grupę (np. „Żydzi won z tego kraju" koder/anotator narodowości polskiej oceniał jak „Polacy won z tego kraju"). Pozwoliło to na częściowe przynajmniej wyrównanie wśród koderów/anotatorów poziomu oceny obraźliwości tekstu.

Pierwsze próby polegające na przybliżeniu wyników automatycznej analizy treści do wyników wzmiankowanego badania (czyli rozkładu odpowiedzi w korpusie 1500 postów) zakończone zostały niepowodzeniem. Jakość przewidywań (czyli kategoryzacji postów jako zawierających język nienawiści) uzyskana tą metodą była niska w stopniu nierokującym na poprawę.

Problem ten, jak się okazało, ma naturę bardziej skomplikowaną niż proste słownikowe podejście, opisane powyżej. Okazało się, że kluczowe jest uchwycenie współwystępowania wielu słów i sensów, a co za tym idzie – wypisanie ich w formie słownika „słów, które uznajemy za wrogie" nie rozwiązuje problemu. Co więcej, wrogość w znaczeniu pewnych słów pojawić się może tylko w wybranych kontekstach leksykalnych, których rozpoznawanie wymaga analizy całego dostępnego tekstu (dyskusji na forum internetowym).

Biorąc pod uwagę wszystkie te doświadczenia, zdecydowaliśmy się na pracę nad wyższym poziomem przeszkolenia zespołu koderów/anotatorów oraz oznaczanie nie tylko w obrębie niewielkiego (1500 tekstów) korpusu treningowego. Korpus, na którym pracowaliśmy, w kolejnym etapie liczył już 11 176 tekstów, z których każdy kategoryzowany był przez jedną osobę. W miejsce rozwijania list słów i wyrazów w słownikach przyjęliśmy inną metodę, polegającą na wykorzystaniu metod uczenia maszynowego.

Budowa instrukcji kodowej

Punktem wyjścia do tej części badania było przekonanie (wynikające z praktyki badawczej), że w procesie badawczym, który uwzględnia metodologię jakościową, jakość kodowania zebranego materiału jest jednym z kluczowych czynników wpływających na cały proces analizy danych. Aby wykorzystać narzę-

[3] Wyświetlanie kolejnych fragmentów tekstów poprzedzone było poleceniem: „Przedstawione teraz zostanie P. kilkadziesiąt autentycznych wypowiedzi polskich internautów. Proszę wskazać, jak bardzo P. zdaniem są one obraźliwe w stosunku do osób, których dotyczą. Odpowiadając, proszę posłużyć się skalą od 0 do 4, gdzie 0 oznacza zdanie niezawierające treści obraźliwych lub neutralne, a 4 oznacza zdanie bardzo obraźliwe. Pozostałe cyfry służą do wyrażenia opinii pośrednich. Odpowiadając, proszę posłużyć się własną oceną".

dzia NLP bazujące na uczeniu maszynowym, proces kodowania musi być tak bardzo ujednoznaczniony jak to tylko (w określonej sytuacji badawczej, warunkowanej harmonogramem i budżetem) możliwe. Rozumiemy przez to pozostawienie jak najmniejszego obszaru na swobodne decyzje koderów/anotatorów, a w konsekwencji nacisk na budowanie jak największej zgodności pomiędzy nimi.

Zgodność między koderami/anotatorami zależy od jakości instrukcji dla anotujących (im bardziej szczegółowe i precyzyjne, tym wyższa zgodność) oraz złożoności semantycznej i syntaktycznej tekstu. Jednak zazwyczaj instrukcje nie są wystarczające: w praktyce okazuje się, że istnieje wiele przypadków brzegowych, noszących cechy kilku kategorii, potencjalnie możliwych do zaklasyfikowania na różne sposoby. Rozwiązaniem tego problemu jest opracowanie (i wspólne szczegółowe omówienie) określonej liczby przypadków tego typu, zidentyfikowanych jako problematyczne, a także przyjęcie wzorcowych rozstrzygnięć, nawet arbitralnych. Dlatego rozwiązaniem jest trening koderów/anotatorów, czyli realizacja w obrębie zespołu badaczy kolejnych zadań polegających na oznaczaniu takich samych partii tekstu i porównywaniu wyników pracy poszczególnych analityków. Istotą treningu nie jest mierzenie zgodności, ale dyskusja i „tłumaczenie się" kodujących z podjętych decyzji. Tylko takie postępowanie może doprowadzić do uwspólnienia rozumienia instrukcji kodowej. W lingwistyce powszechnie przyjętym sposobem rozstrzygania niezgodności jest schemat kodowania, w którym oznaczenia (anotacje) wykonywane są przez dwie osoby o podobnych kompetencjach, a konflikty między nimi rozstrzygane są przez trzecią osobę o najwyższych umiejętnościach i wiedzy. W realizowanym przez nas treningu zadaniem stawianym przed analitykami było osiągnięcie konsensusu w trakcie dyskusji. Celem było zastąpienie arbitralnej decyzji jednego z badaczy merytoryczną argumentacją na rzecz przyjęcia jednego z rozwiązań (przypisania danego kodu).

W analizie treści podstawowe pytanie brzmi: czy badacze posługują się tym samym zestawem pojęć (kategorii języka naturalnego)? Brak zgodności w kodowaniu wskazuje na różnice w rozumieniu pojęć. W przypadku lingwistyki i problemów składniowych niezgodność wynika z nieostrych granic między opisywanymi zjawiskami składniowymi lub braku kompetencji koderów/anotatorów. W przypadku semantyki (również w analizie treści w naukach społecznych) brak zgodności wiąże się także z różnicami w wyznawanych wartościach. Konieczne jest zatem „odkrycie" tych wartości, jasne pokazanie założeń, na których opiera się proces interpretacji.

W kontekście uczenia maszynowego i aplikacji przetwarzających teksty kluczowy jest aspekt rozumienia samego problemu: zgodność między kodującymi jako „górna granica", którą mogą osiągnąć narzędzia automatyczne. Inaczej rozumiemy sukces algorytmów maszynowych, jeśli zgodność między wytrenowanymi anotatorami nie przekracza 0.6 (np. kappa), a inaczej, gdy osiąga 0.95. Zgodność pokazuje, jak bardzo skomplikowane jest badane zjawisko. Z drugiej strony negatywnym punktem odniesienia jest czysto losowy wybór anotacji (oznaczeń). Od jakości anotacji wykonanych przez zespół koderów/anotatorów zależy także zdolność algorytmów do „wyuczenia się" charakterystyki zjawiska. Spójne i konsekwentne oznaczanie badanych znaczeń w tekstach jest kluczowe: algorytmy uczące nie są odporne na sprzeczności lub nieciągłości w danych treningowych.

Aby osiągnąć jak największą zgodność kodujących, a co za tym idzie – kodowanie automatyczne o wysokiej precyzji, należy zrealizować dwa kroki procedury:

1. stworzyć wyczerpujący i kompletny klucz kodowy (opisujący znaczenia i sposoby użycia poszczególnych kodów), przedyskutować znaczenie poszczególnych kategorii z członkami zespołu badawczego.

2. przeprowadzić wieloetapowe szkolenie członków zespołu ankieterskiego, na każdym etapie porównując rezultaty kodowania, i co ważniejsze – wymusić „obronę" danego sposobu kodowania poprzez werbalizację zasad językowych, które pozwoliły koderowi/anotatorowi zastosować określony kod.

Poniżej opisujemy stosowane w omawianym projekcie kategorie kodowe i ich znaczenia. Część tych kategorii to kategorie „techniczne", przygotowane na potrzeby automatyzacji procesu kodowania, zatem ich omówienie ma na celu zaprezentowanie specyfiki omawianego tu procesu. Te właśnie elementy odróżniają nasz klucz kodowy od typowego narzędzia jakościowego. Trudność polegała na konieczności całkowitej algorytmizacji działań koderów. Nie tylko powinni oni kodować „tak samo" (z jak największą zgodnością), ale również powinni być w stanie opisać mechanizmy, które skutkują przypisaniem konkretnego kodu do wypowiedzi. To jest kluczowy czynnik – wyjście poza implicite przyjmowane zasady nadawania sensów bazujące na kompetencji językowej kodujących. Jeśli chcemy skutecznie wyuczyć automat przyjętych przez nas zasad kodowania, konieczna jest ich werbalizacja i algorytmizacja.

Zestawienie użytych kodów:

1. Sensowność (oznaczana jako: S1 lub S0). Teksty, które zostały zakwalifikowane do badania, zostały zapisane w bazie w sposób automatyczny (jako wynik działania skryptu). Co oznacza, że pewna część z nich to fragmenty, których nie można sensownie zinterpretować. Są to wycinki wypowiedzi, pojedyncze litery lub słowa, fragmenty, którym kodujący nie potrafi przypisać jednoznacznego sensu. Zmienną tę kodowaliśmy zerojedynkowo (1, 0). Była to zmienna, która pozwalała na odfiltrowanie nierelewantnych tekstów. Dzięki temu mogliśmy znacząco zmniejszyć poziom szumów w analizowanych wypowiedziach.

2. Temat wypowiedzi (TW_nazwa). Z perspektywy celów badawczych to jedna z najważniejszych zmiennych. Pozwoli nam oszacować, o czym „mówią" zebrane teksty. Przyjęliśmy zestawienie tematów wzorowane na General Inquirer, odwołujące się do dużych systemów społecznych: edukacja, nauka, sztuka, ekonomia, sport, prawo, militaria, polityka, społeczeństwo, religia (por. Stone i in. 1966).

3. Mniejszość (M_nazwa): przypisanie tekstu do określonej mniejszości. Poza nazwami mniejszości konieczne było zakodowania całej gamy określeń wskazujących na przynależność do danej grupy, zarówno tych neutralnych, jak i negatywnych.

Przypadki, gdy sama nazwa mniejszości może być użyta w negatywnym sensie, znacząco ułatwiały właściwe kodowanie sensów (np. „czarnuch").

4. Autoopis (A0 lub A1): określenie, czy autor wypowiedzi mówi sam o sobie. Zmienna ważna dla przypisania właściwego sensu do wypowiedzi.

5. Negatywne emocje (Neg_wartość): określenie poziomu występowania w tekście negatywnych emocji wobec mniejszości. Przyjęliśmy skalę 0–4, gdzie: 0 oznacza brak negatywnych emocji (treść jest albo neutralna, albo pozytywna wobec mniejszości); 1 – wartość pośrednia, 2 – oznacza obecność negatywnych emocji, ale nie są one bardzo silne; 3 – wartość pośrednia; 4 – oznacza obecność skrajnie negatywnych emocji (często występuje razem z wulgaryzmami).

6. Pozytywne emocje (Poz_wartość) – analogicznie kodowaliśmy pozytywne emocje w tekście.

7. Wezwanie do działania negatywnego (W_neg): nawoływanie innych do podjęcia negatywnie nacechowanych działań wobec danej mniejszości.

8. Wezwanie do działania pozytywnego (W_poz): analogicznie, jak powyżej, ale działania o pozytywnych skutkach.

9. Ironia/sarkazm (IS): określenie, czy w danej wypowiedzi znajdują się elementy ironiczne bądź sarkastyczne.

10. Typ ramki (Typ_nazwa): określenie źródła naszej wiedzy o tym, czy fragment zawiera negatywne emocje wobec mniejszości, gdzie: „Leksykalna" oznacza, że wiemy to ze słów użytych w treści (możemy wskazać słowo/słowa, które jednoznacznie odsyłają do negatywnych emocji, np. debile); „Kontekstowa" – oznacza, że „znak" (rodzaj emocji/nastawienia) rozpoznajemy nie z konkretnych słów, a z szerszego kontekstu, zrozumienia wypowiedzi. W pierwszej kolejności sprawdzamy, czy są słowa lub frazy, które są obraźliwe. Jeśli istnieją takie, to oznaczamy ramkę jako „leksykalna" (i kończymy tu kodowanie), jeśli nie – jako „kontekstowa".

11. Kontekstowość (Kon_nazwa): jeśli typ ramki zostanie określony jako kontekstowy, to określamy źródło kontekstu; „Wewnętrzna" oznacza, że kontekst znajduje się w treści analizowanego fragmentu (np. „bo skończycie tak jak oni" [a wcześniej jest napisane, jak skończyli oni]); „Zewnętrzna" – znajomość kontekstu wynika z wiedzy, którą mamy spoza tekstu (np. „(…) bo skończy się jak w '68").

Przyjęliśmy, że aby zakwalifikować tekst jako mowę nienawiści, to musiał on być opisany kodami "Neg_4" oraz "W_neg". Teksty kwalifikowane jako język wrogości to te oznaczone kodami "Neg_1", "Neg_2", "Neg_3", co kwalifikowało nam znacznie większy zbiór tekstów do dalszych analiz. Pozostałe oznaczenia kodowe pełniły funkcję zmiennych niezależnych w badaniu, co pozwalało na pogłębienie analizy zjawiska.

Poniżej przedstawiamy przykłady kodowania z wykorzystaniem tak zbudowanej instrukcji kodowej. Są to wybrane rekordy z bazy danych, które przywołujemy w tym miejscu tekstu, by stanowiły ilustrację naszych rozważań.

Tabela 1.

ID	ID mniejsz	fragment do zakodowania	kody
5243	51	długość ich jest krótsza o 20 lat od długości przeciętnego Kanadyjczyka pederaści i lesbijki częściej popełniają samobójstwa 14 razy częściej niż pozostali przedstawiciele populacji trzykrotnie częściej nadużywają nikotyny częściej popadają w alkoholizm	S1; TW_zdrowie; M_homoseks; A0, neg2; poz0; tr_leks
7439	71	W omawianej publicystyce spotkać się można ze stwierdzeniem aczkolwiek chyba nie powtarzanym tak często i z takim naciskiem jak wspomniane poprzednio Hitler doszedł w roku 1933 do władzy przede wszystkim dlatego zakulisowo wsparli go Żydzi bez ich aproba	S1; tw_historia; m_żydzi; a0; neg1, tr_konteks, kon_wewn
7026	72	polacy są wybitnie inteligentnym narodem potrafią w tak ekstremalnych warunkach jakie są w Polsce bo niemiec jakby do nas przyjechał i miał z tego co zarabia i opłacić mieszkanie samochód nowy z salonu kupiony na raty to by się zes akał w majty	S1, tw_ekonomia; m_niemcy; a0; neg1; tr_konteks; kon_zewn
7472	71	Można zatem głosić Żydzi skazali się sami na to co ich spotkało bo są pozbawieni wszelkiej społecznej solidarności utrzymując zarazem stanowią jedną wielką sitwę solidarnie dbającą wyłącznie o swoje własne interesy i działającą na niekorzyść innych przed	S1; tw_obyczaje; m_zydzi; a0; neg3; tr_leks

Źródło: badania własne.

Dla zrozumienia koncepcji „ramki kontekstowej" warto zwrócić uwagę na 2 teksty:

- tekst nr 7026: jest to wypowiedź o Niemcach, zaklasyfikowana jako negatywna (1). Ale sens tej negatywności zbudowany jest na przekonaniu o powszechnej znajomości faktów dotyczących różnic w sytuacji ekonomicznej Polaków i Niemców. To oznacza, że nie możemy uzasadnić tego kodu (negatywności) poprzez wskazanie konkretnego słowa lub grupy słów;
- tekst nr 7439: wypowiedź o Żydach, zaklasyfikowana jako negatywna (1). Jej sens jest jednoznaczny pod warunkiem, że rozpoznajemy postać historyczną Adolfa Hitlera wraz z wszystkimi jej konotacjami.

Trening zespołu kodującego

Na potrzeby projektu zebraliśmy czteroosobowy zespół koderów/anotatorów, w którym uczestniczyli studenci socjologii zainteresowani analizą treści.

Pierwszym etapem szkolenia było szczegółowe omówienie poszczególnych kategorii kodowych. Po takim wprowadzeniu zespół rozpoczął trwający 4 tygodnie trening. Polegał on na cyklicznych spotkaniach, w trakcie których omawialiśmy wyniki ich pracy, szczególną uwagę poświęcając fragmentom tekstów, które były niejednoznaczne dla koderów. Każde takie spotkanie kończyło się przydzieleniem nowego zbioru tekstów do kodowania. W czasie treningu anotatorzy posługiwali się przygotowanym na użytek tego procesu arkuszem MS Excel (z wpisanymi w poszczególne wiersze fragmentami tekstu i predefinowanymi kategoriami kodowymi w kolejnych kolumnach).

Postępy na tym etapie kodowania mierzyliśmy na oznaczonym równolegle przez wielu kodujących zbiorze 1500 tekstów miarą zgodności – alfą Krippendorffa (Krippendorf 1980). Zbiorem, na którym przeprowadzaliśmy obliczenia, były oznaczone fragmenty tekstów, a wartościami zmiennych – etykiety oznaczonych zjawisk (kodów). Na początku procesu niewytrenowany zespół koderów osiągał dla wszystkich anotowanych kodów zgodność na poziomie α<0,2. Efektem treningów było zbliżenie się zespołu na większości kodów do zgodności na poziomie α=0,6. Taki poziom uznaliśmy za wystarczający, by zakończyć szkolenie i przejść do kodowania całości materiału. Rzecz jasna trening należałoby kontynuować, by dążyć do osiągnięcia jeszcze większej zgodności, ale nie pozwolił nam na to harmonogram prac. Wartość α=0,6 znajduje swe uzasadnienie w literaturze przedmiotu: „Za Lombardem i wsp. (2004) można przyjąć, iż zmienna została rzetelnie stworzona, jeśli wartość wskaźnika przekracza poziom (...) 0,5" (Krejtz, Krejtz 2005: 249).

Dla usprawnienia właściwego kodowania (przygotowania zbioru tekstów jako próby uczącej do kodowania maszynowego) zostało zbudowane narzędzie online pozwalające, po zalogowaniu się, na przypisywanie kodów do fragmentów tekstów i bezpośrednie zapisywanie całości w bazie danych. Narzędzie charakteryzowało się maksymalnie uproszczonym interfejsem, w którym koderzy/anotatorzy, korzystając z list rozwijanych, oznaczali kolejne fragmenty tekstu. Efektem pracy zespołu było ręczne zakodowanie około 12 000 fragmentów tekstów. Interfejs umożliwiał zarządzanie tekstami, szybkie odnajdywanie nieoznaczonych jeszcze fragmentów, edycję wcześniejszych zmian.

Próby automatyzacji kodowania

W tej części opisujemy wyniki eksperymentów, w których do procesu automatycznego kodowania tekstów stosujemy metody komputerowe, określane jako algorytmy uczenia z nadzorem (ang. *supervised learning*). W przypadku tej klasy metod decyzja na temat kodowania tekstu podejmowana jest przez model uczenia maszynowego (ML), wytrenowany uprzednio na podstawie zbioru przykładów, dzięki którym został on nauczony interesującego nas zjawiska. Algorytm testowany przez nas w niniejszym eksperymencie to maszyny wektorów podpierających (SVM), jedna z lepszych i bardziej uznanych metod stosowanych między innymi do automatycznej klasyfikacji tematyki tekstów.

W eksperymentach korzystaliśmy z pakietu oprogramowania ML o nazwie scikit-learn (Pedregosa i in. 2011). Oferuje on nie tylko możliwość elastycznego trenowania i ewaluacji modeli ML, ale także wykonywania innych czynności, takich jak przygotowanie przestrzeni cech (ang. *feature space*), selekcji cech (ang. *feature selection*). Pakiet scikit-learn zawiera między innymi oba algorytmy klasyfikacyjne wspomniane w poprzedniej części artykułu, czyli zarówno maszyny wektorów podpierających (SVM), jak i lasy losowe. W przeprowadzonych eksperymentach korzystaliśmy (za pośrednictwem scikit-learn) z implementacji SVM opartej o bibliotekę liblinear, korzystającą z liniowej funkcji jądrowej (ang. *linear kernel*). Według dostępnej literatury oraz zgodnie z doświadczeniem autorów niniejszego artykułu jest to optymalna funkcja jądrowa dla klasyfikacji danych tekstowych. Głównym powodem jest jej wydajność obliczeniowa przy pracy z rzadkimi macierzami i wysokowymiarowymi danymi. Przestrzeń cech dla danych tekstowych konstruowana jest poprzez wektoryzację (ang. *vectorization*) tekstów, czyli przypisanie poszczególnym słowom tekstu liczbowych identyfikatorów, wykorzystywanych po znormalizowaniu i konwersji TF-IDF jako cechy wejściowe (ang. *input features*) modeli klasyfikujących ML. Reprezentacja tego typu nosi nazwę przestrzeni wektorowej (ang. *vector space*) i efektem jej zastosowania jest właśnie rzadka, wysokowymiarowa macierz cech, której przetwarzanie algorytmem SVM możliwe jest w praktyce tylko za pomocą liniowych funkcji jądrowych. Warto także nadmienić, że korzystaliśmy z domyślnych wartości parametrów wywołania metody SVM pakietu scikit-learn (w szczególności domyślnej wartości stałej C oraz domyślnej regularyzacji).

Algorytmy klasyfikujące działają w dwóch „trybach": uczącym i klasyfikującym. W tym pierwszym trybie budowane są modele interesujących nas zjawisk. Do tego celu konieczne jest „pokazanie" algorytmom dwóch zbiorów przykładowych tekstów, przykładowo obraźliwych i nieobraźliwych (neutralnych). Na podstawie wystąpień słów i kombinacji słów w tekstach lub fragmentach tekstów algorytmy „uczą się" dystynktywnych cech obraźliwego języka. Dzięki pewnym własnościom języka oraz samych algorytmów możemy mówić o zdolnościach generalizacyjnych modeli:

- nieco uproszczonym przykładem takiego generalizującego mechanizmu językowego jest synonimia: wystarczy „pokazać" uczącym się modelom słowo drań, a dzięki synonimii również jest uwzględniane: kreatura, bydlak, łobuz i tak dalej,

- generalizujące własności algorytmów dobrze opisuje w tym przypadku intuicja podobieństwa, mająca swoje matematyczne sformułowanie. Dzięki niemu możemy uznawać za obraźliwe teksty tylko częściowo przypominające, nawet w odległy sposób, teksty już widziane. W trybie uczenia algorytm ustala między innymi próg podobieństwa, czyli jak bardzo „odległe" teksty (względem znanych) można nadal uznawać za obraźliwe i w jakich kontekstach.

W drugim trybie, klasyfikującym, zbudowane uprzednio modele działają analogicznie do anotatorów – ludzi: na podstawie analizy tekstów (nawet niewidzianych w trybie uczącym) podejmują decyzję o ich obraźliwości. Modele te mogą skutecznie rozpoznawać teksty *trochę* podobne do tych oglądanych w trybie uczącym.

W obszarze kodowania tekstów można powiedzieć, że model ML analizuje (tryb uczenia) oraz próbuje powielać (tryb oznaczania lub tagowania) decyzje, które analityk podejmował podczas procesu kodowania. ML w decyzjach tych identyfikuje to, co powtarzalne. Stosuje się tu zatem kryterium frekwencyjne. Celem tego procesu jest zbudowanie modelu opartego na zidentyfikowanych kryteriach decyzyjnych analityka. Jakość korpusu tekstów oznaczonych przez analityków, związana z metodologią procesu kodowania tekstów, jest ściśle związana z jakością wyników uzyskiwanych później przez modele uczenia maszynowego.

Typowymi miarami oceny jakości automatycznego kodowania są precyzja (ang. *precision*) oraz pełność (ang. *recall*). Precyzja związana jest z odsetkiem błędów popełnianych przez model na skutek błędnej identyfikacji fragmentu tekstu jako przynależącego do danej klasy. W naszym przypadku precyzja mówi o tym, jak dużo tekstów zostało poprawnie rozpoznanych przez nasz model (np. jako obraźliwe). Pełność natomiast opisuje, jak wiele tekstów z ich całkowitej liczby (np. tekstów obraźliwych) zostało oznaczonych przez model.

Jakość modeli ML trenowanych na pierwszej wersji korpusu, stworzonej na podstawie początkowej, niezbyt rozbudowanej wersji instrukcji kodowania oraz bez przeprowadzenia intensywnych treningów dla anotujących, nie była zadowalająca. Nieznacznie tylko przekraczały one czysto losowy wybór. Dobrym przykładem ilustrującym to zjawisko były eksperymenty z modelem uczenia maszynowego przewidującym, czy dany fragment tekstu zawiera treści obraźliwe. Mamy tu do czynienia z przewidywaniem 2-wartościowej zmiennej o równej liczebności obu klas (zbiór tekstów, 1500 oznaczonych fragmentów – zazwyczaj zdań – zawierał dokładnie po połowie tekstów obraźliwych i nieobraźliwych, co przy losowym wyborze, np. rzucając monetą, daje precyzję 0.5). Wyniki, jakie uzyskaliśmy, można opisać następująco:

- Algorytmy trenowane na tekstach oznaczanych przez niewytrenowanych anotatorów: precyzja równa 0,57.

- Algorytmy trenowane na tekstach oznaczanych przez wytrenowanych anotatorów: precyzja równa 0,80.

Jak widać, poprawa instrukcji kodowania oraz przeprowadzenie serii treningów anotujących osób znalazło odzwierciedlenie w poprawie jakości kodowania automatycznego o 23 pp, mierzonej liczbą błędów popełnianych przez model ML. Wzrost tego typu niezmiernie rzadko można uzyskać za pomocą metod informatycznych, takich jak na przykład zastosowanie lepszego algorytmu.

Innym pytaniem, jakie należy postawić w tym miejscu, jest kwestia przyczyn błędów popełnianych przez modele uczenia maszynowego przewidujących obraźliwość tekstów, stworzonych na korpusach oznaczonych przez wytrenowanych anotatorów. Przeprowadzona przez nas analiza błędnie rozpoznanych przypadków (precyzja, jak pamiętamy, 0.8) pokazała, że najczęstsze źródła błędów związane są z następującymi powodami:

- Obraźliwość zakodowana na poziomie leksykalnym, ale przy wykorzystaniu nieznanych dotychczas słów. Błąd ten polegał na pojawieniu się słów i fraz „nieoglądanych" przez algorytm w fazie treningowej. Można zatem założyć, że liczebność tego typu błędów można w pewnym stopniu ograniczyć, zwiększając rozmiar danych treningowych.

- Obraźliwość zakodowana na poziomie ponadleksykalnym:

  – Język figuratywny (metafory), niekompozycyjność znaczeń (znaczenie frazy wykracza poza sumę znaczeń słów). Rozpoznawanie struktur tego typu jest dość trudnym zadaniem, od niedawna jednak podejmowanym z umiarkowanym sukcesem przez obiecujący nurt badań związanych z semantyką dystrybucyjno-kompozycyjną (Gutierrez i in. 2016). Zaadresowanie tego typu zjawisk wymaga specjalnych zbiorów treningowych i dedykowanych, dość złożonych rozwiązań algorytmicznych.

  – Odniesienia do kontekstu społecznego i rzeczywistości pozatekstowej. W przypadku tego typu błędów nie istnieją obecnie automatyczne metody przynoszące zadowalające efekty. W dużym stopniu jest to kwestia doświadczeń, indywidualnej wrażliwości i orientacji podmiotu kodującego.

Generalną konkluzją płynącą z tych eksperymentów jest stwierdzenie, że tworzenie skutecznych narzędzi automatycznych, rozpoznających wybrane aspekty znaczenia tekstów, takich jak ich obraźliwość względem mniejszości, wymaga zapewnienia, że badacze posługują się tym samym zestawem pojęć. Wysoka niezgodność pokazuje na różnice w rozumieniu pojęć, co odbija się bezpośrednio na jakości automatycznych modeli komputerowych. Niezgodność ta często wynika z nieostrych granic między opisywanymi zjawiskami lub brakiem kompetencji językowych osób kodujących. W analizie treści w naukach społecznych brak zgodności wiąże się także z różnicami w wartościach między anotującymi.

Wszystkie te obserwacje prowadzą nas do stwierdzenia, że kodowanie automatyczne nie jest metodą samodzielną; jest propozycją przypisania określonych kategorii, która powinna być ponownie weryfikowana przez badaczy.

Podsumowanie

W prezentowanym tekście staraliśmy się prześledzić kolejne kroki na drodze do owocnej współpracy pomiędzy naukami społecznymi a lingwistyką komputerową. Analizowany projekt był realizowany według metody typowej dla jakościowej analizy treści, znaczącą zmianą jest zastosowanie uczenia maszynowego, by zastąpić kodowanie przez grupę koderów/anotatorów kodowaniem maszynowym.

Wracając do postawionego powyżej pytania: jak zoptymalizować działania zespołu badawczego pod kątem przygotowania zakodowanych tekstów, które staną się próbą uczącą dla algorytmów uczenia maszynowego?

Bazując na doświadczeniach przy realizacji monitoringu mowy nienawiści, wskazujemy następujące elementy optymalizacji procesu badawczego:

1. Cały proces musi być przygotowany wyjątkowo starannie, konieczna jest werbalizacja jak największej liczby założeń (przyjmowanych najczęściej implicite przez członków zespołu), szczegółowa algorytmizacja kolejnych działań przewidzianych w poszczególnych etapach.

2. Konieczna jest precyzyjna konceptualizacja i operacjonalizacja kategorii teoretycznych oraz pokazanie na przykładach brzegowych (trudnych, niejednoznacznych), jak te kategorie stosować w praktyce.

3. Konieczne jest, przy budowie korpusu, uwzględnienie na poziomie słów kluczowych zarówno określeń neutralnych, pozytywnych, jak i negatywnych (obraźliwych).

4. Szczególnej uwagi wymaga proces kodowania zebranych fragmentów tekstu. Dla maksymalizacji zgodności kodujących konieczne jest:

   a. wprowadzenie rozbudowanej instrukcji kodowej – przygotowanej przez doświadczonych badaczy, którzy w pełni rozumieją cele badania oraz brali udział we wszystkich jego etapach (konceptualizacji i operacjonalizacji);

   b. stworzenie zespołu koderów/anotatorów – osób o zbliżonych kompetencjach językowych, które gotowe są przyjąć i przyswoić sposób interpretacji tekstu narzucony przez badaczy;

   c. przeprowadzenie treningu koderów/anotatorów – „próbne" kodowanie tekstów wybranych (wylosowanych) z korpusu; proces ten powinien być oparty o kilka iteracji. Kluczowa jest werbalizacja przesłanek stojących za wyborem poszczególnych kodów, a następnie ujednolicenie tych przesłanek dla całego zespołu, osiągnięcie konsensusu w procesie racjonalnej dyskusji;

   d. „kody techniczne" – konieczne jest wprowadzenie kodów, które pozwolą rozdzielić różne sposoby nadawania sensu wypowiedzi. W klasycznym kodowaniu te rozróżnienia są elementem kompetencji językowej kodujących i pozostają najczęściej niezwerbalizowane.

Poprawa jakości danych, będąca skutkiem zastosowania zbioru niniejszych zaleceń, ma widoczny i znaczący efekt w postaci lepszej jakości modeli uczenia maszynowego. Oczywiście nie wyczerpuje się w ten sposób możliwości popełnienia zarówno pomyłek ludzkich – na etapie tworzenia danych treningowych – oraz pomyłek maszyny, na etapie testowania modeli algorytmów. Tematyka automatycznego kodowania treści wymaga dalszych badań, kierunek wyznacza rozwijająca się infrastruktura Clarin (narzędzia NLP na potrzeby nauk społecznych i humanistycznych, http://clarin-pl.eu/pl/czym-jest-clarin/).

W ocenie autorów niniejszego tekstu dalszy rozwój metod przetwarzania języka naturalnego umożliwi komputerowe rozpoznawanie coraz bardziej złożonych pojęć, wykraczających poza obszar lingwistyki. Pojęć społecznie konstruowanych, które tradycyjnie stanowiły domenę jakościowych badań nad tekstem. Z technicznego punktu widzenia stanie się to możliwe dzięki coraz lepszym metodom reprezentacji znaczenia słów i tekstu (ang. *embeddings*) oraz głębokiemu uczeniu maszynowemu.

To przybliża nas do zrozumienia celu opisywanego procesu – skutecznej współpracy pomiędzy przedstawicielami NLP a nauk społecznych. Wydaje się, że na obecnym etapie specjalizacji poszczególnych dyscyplin konieczne jest budowanie multidyscyplinarnych zespołów naukowych. Dzięki temu badacze społeczni zyskają narzędzia pozwalające skutecznie analizować duże korpusy tekstów, co między innymi przybliża nas do zrozumienia zjawisk społecznych reprezentowanych przez dyskurs internetowy, a lingwiści komputerowi otrzymują możliwość zderzenia wypracowanych algorytmów i metod analizy danych z realnymi, nietrywialnymi problemami społecznymi.

Bibliografia

Bishop Christopher (2006) *Pattern Recognition and Machine Learnin*g. Secaucus: Springer-Verlag.

Breiman Leon (2001) *Random Forests*. „Machine Learning", vol. 45, no. 1, s. 5–32.

Bychawska-Siniarska Dominika, Głowacka Dorota, red., (2013) *Mowa nienawiści w internecie: jak z nią walczyć*. Warszawa: Helsińska Fundacja Praw Człowieka.

Bychawska-Siniarska Dominika, Gliszczyńska-Grabias Aleksandra (2016) *W stronę sieci tolerancji. Prawnomiędzynarodowe instrumenty walki z mową nienawiści* [dostęp 14 maja 2017 r.]. Dostępny w Internecie ‹http://www.siectolerancji.pl/aktualnosc/w-strone-sieci-tolerancji-publikacja-w-module-prawnym›.

Cortes Corinna, Vapnik Vladimir (1995) *Support-Vector Networks*. „Machine Learning", vol. 20, no. 3, s. 273–297.

Gutierrez Dario i in. (2016) *Literal and Metaphorical Senses in Compositional Distributional Semantic Models*. Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, (ACL) 2016, August 7-12, 2016, Berlin, Germany, vol. 1 [dostęp 14 maja 2017 r.]. Dostępny w Internecie: ‹http://aclweb.org/anthology/P/P16/P16-1018.pdf›.

Heinze Eric (2016) *Hate Speech and Democratic Citizenship*. Oxford: Oxford University Press.

Jockers Matthew (2013) *Macroanalysis: Digital Methods and Literary History*. Champaign: University of Illinois Press.

Krejtz Izabela, Krejtz Krzysztof (2005) *Wybrane statystyki zgodności między sędziami w analizie treści* [w:] Katarzyna Stemplewska-Żakowicz, Krzysztof Krejtz, red., *Wywiad psychologiczny. Wywiad jako postępowanie badawcze*. Warszawa: Pracownia Testów Psychologicznych Polskiego Towarzystwa Psychologicznego, s. 231–249.

Krippendorff Klaus (1980) *Content Analysis: An Introduction to Its Methodology*. Newbury Park, CA: Sage.

Lafferty John D., McCallum Andrew, Pereira Fernando C. N. (2001) *Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data*. Proceedings of the Eighteenth International Conference on Machine Learning (ICML '01), San Francisco, USA, Morgan Kaufmann Publishers Inc., s. 282–289.

Lample Guillaume i in. (2016) *Neural Architectures for Named Entity Recognition*. Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics. The Association for Computational Linguistics, s. 260–270.

Linde-Usiekniewicz Jadwiga (2015) *Teoria relewancji jako narzędzie opisu mowy nienawiści*. „Studia Pragmalingwistyczne", t. 7, s. 53–68.

Lombard Matthew, Snyder-Duch Jennifer, Bracken Cheryl Campanella (2004) *A Call for Standardization in Content Ana-*

lysis Reliability. „Human Communication Research", vol. 30, s. 434–437.

Łodziński Sławomir (2003) *Problemy dyskryminacji osób należących do mniejszości narodowych i etnicznych w Polsce*. Warszawa: Kancelaria Sejmu, Biuro Studiów i Ekspertyz.

Manning Christopher D. i in. (2014) *The Stanford CoreNLP Natural Language Processing Toolkit*. Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics. The Association for Computational Linguistics. ACL, System Demonstrations.

Moretti Franco (2013) *Distant Reading*. London: Verso Books.

Nijakowski Lech (2008) *Mowa nienawiści w świetle teorii dyskursu* [w:] Anna Horolets, red., *Analiza dyskursu w socjologii i dla socjologii*. Warszawa: Wydawnictwo Adam Marszałek, s. 113–133.

Ogrodniczuk Maciej, Lenart Michał (2013) *A Multi-Purpose Online Toolset for NLP Applications*. Proceedings of the 18th International Conference on Applications of Natural Language to Information Systems, vol. 7934 of Lecture Notes in Computer Science, Springer-Verlag. Springer Berlin Heidelberg, s. 392–395.

Pedregosa Fabian i in. (2011) *Scikit-Learn: Machine Learning in Python*. „Journal of Machine Learning Research", vol. 12, s. 2825–2830.

Siwicki Maciej (2011) *Nielegalna i szkodliwa treść w Internecie. Aspekty prawnokarne*. Warszawa: Oficyna Wolters Kluwer.

Sperber Dan, Wilson Deidre (2011) *Relewancja. Komunikacja i poznanie*. Przełożyły Magdalena Charzyńska i n.. Kraków: Wydawnictwo Tertium.

Stone Philip J. i in. (1966) *The General Inquirer: A Computer Approach to Content Analysis*. Cambridge: MIT Press.

Troszyński Marek (2015) *Hate Speech. Towards a Research Standard* [w:] Jacek Sobczak, Jędrzej Skrzypczak, red., *Professionalism in Journalism in the Era of New Media*. Berlin: Logos, s. 199–208.

Wawer Aleksander, Rogozińska Dominika (2012) *How much supervision? Corpus-based lexeme sentiment estimation*. IEEE 12th International Conference on Data Mining Workshops (SENTIRE 2012), Los Alamitos, USA, IEEE Computer Society, s. 724–730

Wieruszewski Roman i in., red., (2010) *Mowa nienawiści a wolność słowa. Aspekty prawne i społeczne*. Warszawa: Wolters Kluwer.

Cytowanie

Troszyński Marek, Wawer Aleksander (2017) *Czy komputer rozpozna hejtera? Wykorzystanie uczenia maszynowego (ML) w jakościowej analizie danych*. „Przegląd Socjologii Jakościowej", t. 13, nr 2, s. 62–80 [dostęp dzień, miesiąc, rok]. Dostępny w Internecie: ‹www.przegladsocjologiijakosciowej.org›.

Can a Computer Recognize Hate Speech? Machine Learning (ML) in Qualitative Data Analysis

Abstract: The purpose of this article is to present the process of automatic tagging of hate speech in social media. The implementation of this process allows for quantitative treatment of qualitative methods: analysis on the corpora of hundreds thousands of texts based on their meaning. The process is possible through algorithms of machine learning (ML).

The example of the hate speech designation project in texts from Polish online forums is presented. The key issue is the precise of conceptualization and operationalization of category "hate speech." This allows for preparing specific instructions and conducting the training code unit. As a result we get higher rates of inter-coder agreement. Marked texts will be used as training data for automated categorization methods based on ML algorithms. Then we describe the course of machine coding. This article also seeks to establish problems associated with automatic coding of hate speech and propose solutions. In summary, we point the factors that are crucial to the research process that uses machine learning.

Keywords: machine learning, qualitative data analysis, hate speech, intercoder agreement