Rozpoznawanie obraźliwych treści poszerzone o podejścia wielomodalne

Alicja Figas, Marcin Gruza, Daria Puchalska
Wydział Informatyki i Zarządzania
Politechnika Wrocławska
Wyb. Wyspiańskiego 27, 50-370 Wrocław
{238442, 256875, 234800}@student.pwr.edu.pl

1 Wstęp

W dzisiejszych czasach mamy powszechny dostęp do informacji, a tworzenie nowych treści jest łatwe i dostępne dla wszystkich. Internet daje nam możliwość anonimowego wyrażania opinii lub stwarza iluzję anonimowości. Podczas dyskusji w mediach społecznościowych często nie widzimy twarzy i reakcji innej osoby. Takie środowisko sprzyja obrażaniu innych i łatwej publikacji nienawistnych treści, co widać w rosnącej liczbie takich postów [4, 5]. Nie ma formalnej definicji obraźliwej i nienawistnej mowy, ale często jest ona opisywana jako mowa skierowana do określonych grup społecznych w sposób, który jest dla nich raniący [16]. Problem wypowiedzi obraźliwej jest różnie traktowany w różnych miejscach na świecie. Na przykład w Stanach Zjednoczonych mowa nienawiści jest chroniona na mocy Pierwszej Poprawki, ponieważ zdaniem wielu konstytucjonalistów nie ma jasnego sposobu na odróżnienie tego typu wypowiedzi od innych form wyrazu politycznego [11]. W tym podejściu wolność słowa jest zakorzeniona w autonomii i godności człowieka. Jednak te wartości również mogą być naruszane przez niewłaściwe wypowiedzi, a w kręgach akademickich trwa debata na temat przepisów dotyczących mowy nienawiści.

W tej pracy zajmujemy się obszarem subiektywnych zjawisk obserwowanych w tekście. Niektóre teksty mogą być odbierane przez określone osoby jako wspierające lub usprawiedliwiające nienawiść. Takie teksty mogą spowodować, że odbiorca poczuje się wykluczony, urażony lub skrzywdzony. Tego typu teksty określamy zbiorczo jako *obraźliwa treść*. Traktujemy je jako zjawisko, które nie jest przez wszystkich postrzegane jednakowo. Istnieją różne metody i dostępne zbiory danych służące do identyfikacji obraźliwej, toksycznej i społecznie niedopuszczalnej mowy, a także jej określonych typów, takich jak mowa nienawiści i cyberprzemoc. Wszystkie te problemy mają jedną wspólną cechę: mają umiarkowany poziom zgodności adnotacji. Oznacza to, że mierząc, jak dobrze dwie osoby postrzegają i podejmują decyzje dotyczące adnotacji dotyczących tej samej treści (np. agresja lub brak agresji w tekście), możemy rozpoznać osobiste uprzedzenia. Wiedząc, że na odbiór obraźliwych treści wpływa wiele czynników (jak przykładowo relacja między nadawcą i odbiorcą), poleganie na samej treści może nie wystarczyć, aby jednoznacznie sklasyfikować ją jako obraźliwą lub nie.

2 Powiązane prace

2.1 Obraźliwe, agresywne, toksyczne treści, mowa nienawiści i cyberprzemoc

Jeśli chodzi o zdefiniowanie terminologii treści obraźliwych, wydaje się to równie skomplikowane, jak określenie, co może być obraźliwe dla danej osoby. W klasycznych podejściach do adnotacji jasna specyfikacja byłaby również ważna z punktu widzenia zgodności annotatora. Dla dostawców zbiorów danych kluczowe jest, aby pojęcia były rozumiane w ten sam sposób podczas procedur adnotacji treści.

Istnieje kilka przeglądów dotyczących identyfikacji obraźliwych treści, które zawierają terminologię z tej dziedziny [9, 38, 1, 29]. Fortuna [9] w automatycznym wykrywaniu mowy nienawiści zebrał

definicje mowy nienawiści z różnych źródeł, w tym z artykułów naukowych, przepisów Komisji Europejskiej, terminów z mediów społecznościowych oraz organizacji mniejszościowych, których celem jest ochrona przed mową nienawiści.

Proponują również własną definicję *mowy nienawiści*: *"Mowa nienawiści to język atakujący lub umniejszający, podżegający do przemocy lub nienawiści wobec grup, w oparciu o określone cechy, takie jak wygląd fizyczny, religia, pochodzenie, narodowość lub pochodzenie etniczne, orientacja seksualna, tożsamość płciowa lub inne, i może występować w przypadku różnych stylów językowych, nawet w subtelnych formach lub z użyciem humoru."*[9]

Schmidt [38] podaje w swojej recenzji nieco inną definicję, w której *mowa nienawiści* jest zdefiniowana jako *„powszechnie definiowana jako każda komunikacja dyskredytująca osobę lub grupę na podstawie pewnych cech, takich jak rasa, kolor skóry, pochodzenie etniczne, płeć, orientacja seksualna, narodowość, religia lub inne cechy."*

W ramach terminologii obraźliwej mowy istnieje również koncepcja *uwłaczającego języka*, czyli języka, który może być bolesny dla czytelnika, w tym obraźliwe słowa i wulgaryzmy [25].

Te formy przemocy lub dyskryminacji mogą być bardzo subtelne i wyrażone w postaci żartu lub stereotypu. Oprócz mowy nienawiści, istnieje również definicja *cyberprzemocy*, która uważana jest za czyn, który jest zamierzony i dokonywany w sieci przez osobę lub grupę, o powtarzającym się charakterze, dokonywany w kierunku młodych ludzi [6].

Istnieje również kilka innych powiązanych terminów, takich jak *trolling, toksyczne treści, obsceniczna mowa*. Również bardziej szczegółowe *rasistowskie treści* lub *mizoginia* mogą być traktowane jako obraźliwe [26]. W [29] przedstawiono hierarchię pojęć związanych z obraźliwością, w której wszystkie te pojęcia w pewien sposób się pokrywają.

Wszystkie powyższe terminy można zebrać w ramach ogólnego tematu - **obraźliwych treści**. Co więcej, wszystkie te rodzaje treści są subiektywne i mogą być postrzegane jako obraźliwe przez jedną osobę lub grupę, a nie jako takie przez inną grupę. Dowodem na tę podmiotowość jest brak precyzyjnej i wspólnie uzgodnionej definicji takich treści.

2.2 Metody detekcji obraźliwej mowy

W dziedzinie przetwarzania języka naturalnego (ang. *natural language processing*, NLP) identyfikacja obraźliwych treści jest problemem badanym od kilku lat. Ostatnio była nawet przedmiotem otwartych zadań na konferencjach związanych z NLP, takich jak:

- *SemEval 2019 Task 5 (HatEval) & 6 (OffensEval)* na NAACL HLT 2019: wielojęzykowa detekcja mowy nienawiści przeciw imigrantom i kobietom na Twitterze [3] & rozpoznawanie i kategoryzacja obraźliwego języka w mediach społecznościowych[49],
- *GermEval 2018 Task* na Swiss-Text & KONVENS 2018: identyfikacja obraźliwego języka dla niemieckiego [45],
- *HASOC track* na FIRE 2019: rozpoznawanie mowy nienawiści i obraźliwych treści w językach indoeuropejskich [20],
- *PolEval 2019 Shared Task 6*: automatyczne wykrywanie cyberprzemocy na polskim Twitterze [31].

W kontekście reprezentacji tekstu najbardziej klasyczne metody opierają się na wektorach reprezentujących występowanie słów w tekście (jako informacja o wystąpieniu lub jego częstotliwości), bez uwzględniania kolejności słów, np. model *bag-of-words* lub TF-IDF [7, 36, 39]. Często są one wzbogacane dodatkowymi pojęciami z podanych ontologii lub WordNetów i używane razem z metodami klasycznymi, takimi jak SVM [39, 34] lub regresja logistyczna [7, 36, 44].

Wraz z zwiększającą się zdolnością przetwarzania i dostępnością dużych zbiorów danych, pojawiły się również nowe metody reprezentacji oparte na dużych korpusach tekstowych. Prostsze z nich zakładają konstrukcję wektorowych reprezentacji słów w oparciu o prostą analizę kontekstów w dokumentach, tzw. wektory osadzeń słów [45, 20], wraz z bardziej zaawansowanymi metodami, np. CNN [48, 27], LSTM [48, 47] czy GRU [30, 14]. W ostatnich latach obserwowaliśmy rozwój modeli opartych na uczeniu głębokim (ang. *deep learning*, DL), głównie modelach językowych opartych na Transformerach. Ich wygląd spowodował znaczną poprawę jakości w wielu zadaniach przetwarzania

języka naturalnego. Możemy tu wyróżnić trzy główne grupy. Pierwsza z nich obejmuje modele DL zbudowane dla jednego języka naturalnego, np. BERT i jego modyfikacje, takie jak RoBERTa i ALBERT. Modele oparte na BERT były już wykorzystywane do rozpoznawania mowy nienawiści i obraźliwych treści [22, 24]. Druga grupa obejmuje modele wielojęzyczne, np. XLM-RoBERTa, MultiFiT, mBERT [2], LASER [20, 18] i XLM. Modele te są również stosowane do zadań identyfikacji obraźliwych treści w wielu językach [3], a także do zadań wielojęzycznej analizy sentymentu [21]. Ostatnią grupą modeli, która jest obecnie intensywnie rozwijana, są modele wielomodalne. Większość badań w tej dziedzinie koncentruje się na wspólnej reprezentacji języka naturalnego i obrazów. Przykładowe rozwiązania to: Unicoder-VL, ViLBERT i UNITER.

Jednym z wyzwań w zadaniu identyfikacji obraźliwych treści jest wysokie poziom niezbalansowania klas. Problem ten można rozwiązać już na etapie zbierania danych, stosując różne strategie w celu utrzymania rozkładu klas na zadowalającym poziomie [48]. Alternatywnie można zastosować undersampling lub oversampling. Pierwsza z metod usuwa niektóre przypadki należące do klasy większościowej [30], podczas gdy druga dodaje nowe obserwacje do klasy mniejszościowej [13]. Dodatkowe sztuczne teksty można przykładowo wygenerować za pomocą tłumacza Google [18]. Niektórzy badacze uważają jednak to za niepotrzebne, argumentując, że zbiór danych powinien w jak największym stopniu odzwierciedlać rzeczywistość [44, 43].

Większość badań dotyczących automatycznej identyfikacji treści obraźliwych koncentruje się na cechach tekstowych [7, 36, 39, 48, 27, 8]. Tekst reprezentuje jednak tylko jedno źródło informacji (modę), które można wykorzystać. Pozostałe to: obrazy [46, 10, 41, 40], metadane użytkownika [44, 35, 33], czy kombinacja obu [50, 12]. Wykorzystanie obrazu może zwiększyć dokładność modeli, ponieważ dodaje dodatkowy kontekst wizualny do dyskusji pomiędzy ludźmi w mediach społecznościowych jak Instagram [40], Facebook [46], a także do przetwarzania memów [41]. Z drugiej strony, również metadane użytkownika mogą pomóc modelowi wskazać potencjalnych *agresywnych użytkowników*. Nawet prosta reprezentacja relacji między użytkownikami w sieci społecznościowej (kto kogo obserwuje) daje nam pewien kontekst. Autorzy [32] pokazali, że wykorzystanie tego jako dodatkowe wejście dla modelu (4-elementowy wektor *one-hot* reprezentujący relację między autorem tweeta a odbiorcą) potrafi znacznie poprawić jakość predykcji.

2.3 Istniejące zbiory danych

Angielski jest najpopularniejszym językiem wśród publicznie dostępnych zbiorów danych związanych z wykrywaniem obraźliwych treści i mowy nienawiści. Jednym z często eksplorowanych jest *Offensive Language Identification Dataset (OLID)*, który był przedmiotem Zadania 6 na SemEval 2019 [49] i został szczegółowo opisany w [48]. Wykorzystuje nowy trójpoziomowy hierarchiczny schemat adnotacji, aby najpierw określić, czy tekst jest obraźliwy, czy nie, a następnie zdecydować o jego (1) typie (*skierowany* lub *nieskierowany*) i (2) celu (*osoba, grupa* lub *inny*). Źródłem danych był Twitter, a ich zebranie opierało się na frazach często spotykanych w obraźliwych tekstach, takich jak *you are* lub *gun control*. Zbiór zawiera 14 100 tweetów, oznaczonych za pomocą platformy crowdsourcingowej i pytań testowych w celu odrzucenia adnotacji od osób, które nie osiągnęły określonego progu poprawności. Około 33% tekstów zostało oznaczonych jako obraźliwe.

Zbiór danych opisany w [7] jest oparty na słowach i frazach z leksykonu mowy nienawiści *Hatebase.org*. Zawiera prawie 25 000 tweetów, sklasyfikowanych przez annotatorów *CrowdFlower* na trzy możliwe kategorie: (1) mowa nienawiści, (2) obraźliwa, ale nie mowa nienawiści, lub (3) żadna. Annotatorzy podejmowali decyzje na podstawie definicji podanej przez autorów. Ostateczna etykieta klasy została przypisana przy użyciu głosowania większościowego, a wynik zgodności między etykietującymi, wyznaczony przez platformę crowdsourcingową, wyniósł 92%. Ten zasób wyróżnia się na tle innych, ponieważ klasa ofensywna jest dominującą (76% tweetów), co prawdopodobnie wynika z przyjętej metody zbierania danych.

Angielski jest językiem dominującym wśród zbiorów danych w tej dziedzinie, jak stwierdzono w [38] w 2017 r. Jednak od tego czasu publikowanych jest coraz więcej zbiorów danych w języku innym niż angielski. Na przykład zbiór obraźliwych tweetów w języku niemieckim z jednego z zadań na GermEval 2018 [45]. Składa się z 8541 tekstów ręcznie oznaczonych przez jednego z trzech organizatorów zadań (native speakerów z Niemiec). Aby zapewnić wysoką jakość zbioru danych, autorzy sprawdzili zgodność między annotatorami poprzez wspólną annotację 300 przykładów. Niektóre teksty musiały zostać usunięte ze względu na dużą liczbę błędów gramatycznych uniemożliwiających zrozumienie tekstu lub z innych powodów. W pozostałych 240 tweetach zmierzono zgodność $\kappa = 0.66$, która jest

uważana za znaczną [19]. Przyjęta metoda zbierania danych umożliwiła uzyskanie rozkładu klasy obraźliwej w stosunku 1:2 do tekstów nieobraźliwych, w zbiorze treningowym i testowym.

Nowo utworzone zbiory danych dotyczące obraźliwych treści, mowy nienawiści, cyberprzemocy i podobnych problemów obejmują wiele języków innych niż angielski, takich jak arabski [23], indonezyjski [15], włoski [37], polski [31], i hiszpański [28]. W [42] autorzy przedstawiają przegląd 50 zbiorów danych w tej dziedzinie, w tym 30 zbiorów danych w języku innym niż angielski, co ułatwiło badania nad metodami wielojęzycznymi. Na przykład dziewiętnaście zbiorów danych w dziewięciu różnych językach zostało przeanalizowanych w [2], zarówno w trybie jedno-, jak i wielojęzycznym.

3 Zbiór danych

Aby wypełnić obecną lukę w dostępności danych z analizowanego obszaru dla języka polskiego, postanowiliśmy stworzyć własny zbiór. Zebrane przez nas teksty oraz ich metadane pochodzą z serwisu `www.wykop.pl`. Dla okresu od 09.11.2019 do 21.11.2020 pobranych zostało ponad 500 tysięcy komentarzy umieszczonych na stronie, wraz z informacją o nazwie użytkownika, do którego kierowany był komentarz oraz cechami profilu autora tekstu. Na podstawie zebranych danych zbudowano graf kontaktów między użytkownikami, który został ograniczony do najbardziej aktywnych użytkowników, odrzucono teksty krótsze niż sześć słów, a następnie wybrano największy komponent, składający się z około sześciu tysięcy komentarzy, na których przeprowadziliśmy proces anotacji.

Każdy tekst został oznaczony przez trzech anotatorów jako obraźliwy lub nieobraźliwy, a sama anotacja odbyła się przy pomocy przygotowanego przez nas środowiska opartego o narzędzie open-source *doccano*. Niektóre teksty musiały zostać pominięte i wykluczone ze zbioru ze względu na brak zrozumiałego tekstu (posty składające się z samych emotikon, znaków) albo były napisane w innym języku niż polski. Za definicję obraźliwości została przyjęta poniższa: *„Tekst jest uznawany za obraźliwy, jeśli może lub ma na celu sprawić, że ktoś poczuje się urażony, zdenerwowany lub pokrzywdzony lub też w wyraźny sposób atakuje jednostkę lub grupę”*. W procesie anotacji zorganizowano wielokrotne spotkania, aby przedyskutować sporne przykłady i poprawić zgodność na etykietowanym zbiorze. Ostatecznie udało się osiągnąć zgodność na poziomie $\kappa = 0.89$, wyrażonej za pomocą kappy Fleissa. Ostatecznie każdemu przykładowi przydzielona została etykieta na podstawie głosu większościowego. Finalnie zbiór składa się z 5821 komentarzy, a stosunek liczby tekstów obraźliwych do nieobraźliwych wyniósł 10.75%.

4 Analiza i przetwarzanie zbioru

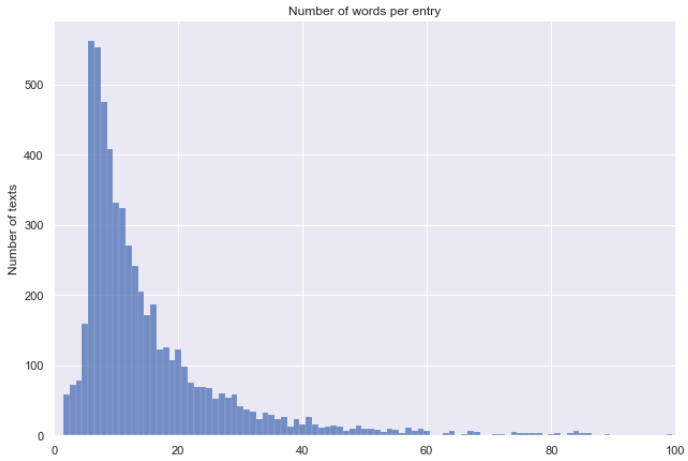
Aby lepiej zrozumieć charakterystykę zbioru, dokonano jego wstępnej analizy eksploracyjnej. Najpierw każdy z tekstów został odpowiednio przetworzony, aby usunąć niepotrzebny szum w danych:

- usunięcie loginów użytkowników zawartych w treści postów,
- usunięcie odnośników (linków, `url`),
- usunięcie emotikon oraz znaków interpunkcyjnych,
- ujednolicenie zapisu do małych liter.

Po dokonaniu czyszczenia charakterystyka zbioru uległa zmianie, część tekstów została bowiem znacząco skrócona. Zbiór w większości składa się z bardzo krótkich tekstów, których długość najczęściej jest mniejsza niż 20 wyrazów. Jedynie 40 tekstów z niemal sześciu tysięcy złożone były z więcej niż stu słów. Rozkład długości wpisów, w oparciu o podział na podstawie spacji w tekście, został zaprezentowany na Rysunku 1.

Aby móc przybliżyć tematykę analizowanych treści, z postów wcześniej przy pomocy wyrażeń regularnych wyciągnięte zostały użyte tagi. Na ich podstawie stworzono chmurę tagów, w której wielkość słów odwzorowuje ich popularność w danych (jak często były używane). Część z nich dotyczy aktualnie poruszanych w społeczności tematów jak `#koronawirus`, `#wybory` czy `#protesty`, obserwujemy jednak kilka tagów ściśle politycznych jak `#kononowicz`, `#polityka`, `#bekazpisu`.

Ponieważ poza samą treścią komentarzy zostały również dodatkowe informacje, jak m.in. liczba głosów, które uzyskał dany post, poddane te wartości analizie, aby przekonać się, czy treści obraźliwe mogą cieszyć się większą popularnością (dostawać więcej głosów) niż treści nieobraźliwe.



Rysunek 1: Rozkład liczby słów dla tekstów z analizowanego zbioru.

Rysunek 2: Chmura najpopularniejszych tagów używanych w analizowanych tekstach.

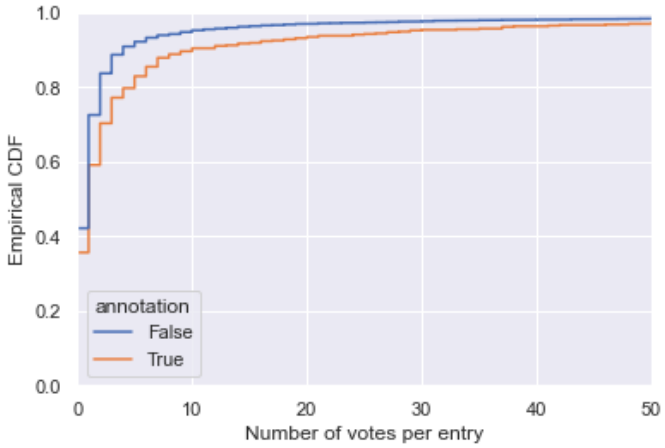
Dystrybuanta empiryczna dla rozkładu liczby głosów w zależności od przydzielonej etykiety dla tekstu została przedstawiona na Rysunku 3. Prawdopodobieństwo uzyskania mniej niż x głosów jest zawsze większe w przypadku treści nieobraźliwych, niezależnie od przyjętej wartości x, co wyraźnie sugeruje, że obraźliwe posty przyciągają większą uwagę użytkowników.

Średnia liczba głosów dla tekstu oznaczonego jako obraźliwy wyniosła 5.81, natomiast dla treści obraźliwych było to 8.01. Zbadano również, jak często teksty te zawierały słowa ze zdefiniowanego przez nas słownika, złożonego z wyrażeń występujących w `Hatebase.org` oraz listy przekleństw i wyzwisk. Tylko 7% tekstów nieobraźliwych zawierało przynajmniej jedno ze zdefiniowanej listy, a w przypadku tekstów oznaczonych jako obraźliwe było to 31% tekstów.

5 Modele tekstowe

W pierwszej kolejności do analizy obraźliwości zebranego zbioru zastosowano modele oparte wyłącznie na zawartości tekstowej postów. Dalej podejście to zostanie rozszerzone o metadane postu i użytkownika, a także relację między autorem i odbiorcą tekstu w sieci społecznej serwisu utworzonej na podstawie oznaczeń (ang. *mentionów*, @) w treści. Aby móc porównać ze sobą wyniki poszczególnych modeli tekstowych, zastosowano 5-foldową stratyfikowaną walidację krzyżową.

Jako proste podejście bazowe stworzono dwa modele bazujące na słownikach słów: model bazujący tylko na słowach w języku polskim ze strony *Hatebase.org*, która zawiera obraźliwe słowa dla wielu języków, a także model wykorzystujący ten sam słownik, poszerzony jednak o zdefiniowaną przez nas listę wulgaryzmów w oparciu o źródła internetowe. Zasada przydzielania etykiety do tekstu była prosta: model słownikowy oznaczał tekst jak *obraźliwy*, jeśli zawiera co najmniej jedno



Rysunek 3: Dystrybuanta empiryczna liczby głosów pod postem w zależności od przydzielonej etykiety.

słowo ze zdefiniowanego słownika, w przeciwnym wypadku przydzielana etykieta to *nieobraźliwy*. Modele te nie uzyskały dobrych rezultatów. Pierwszy z nich, opierający się tylko na słowach z bazy *Hatebase*, uzyskał średni wynik na poziomie F1-Score równy 0.13, co jest bardzo słabym rezultatem. Drugi z modeli, oparty na poszerzonym słowniku wulgaryzmów, osiągnął średni F1-Score z 5 foldów około 0.3.

Ze względu na charakter tekstów, w których często pojawiają się literówki, błędy ortograficzne lub celowe zamiany liter (np. zamiana „o" na „0"), nie poddawano szczegółowym badaniom klasycznym podejściom do problemów tekstowych jak *Bag of Words* czy TF-IDF. Zbadano wprawdzie kilka kombinacji parametrów dla modelu złożonego z TF-IDF w połączeniu z klasyfikatorem SVM, jednak wyniki były znacznie gorsze nawet w porównaniu z modelem słownikowym *Hatebase*.

Ze względu na szybkość działania oraz algorytm oparty na podsłowach, co może być użyteczne przy problemie literówek w tekście, zdecydowano wykorzystać wektory zanurzeń słów FastText (wersji KGR10 dla języka polskiego [17]). Wybrano wersję, która zwraca 300-wymiarowy wektor dla całego tekstu podanego jako wejście do modelu wektor. Tworzony jest on na podstawie wektorów zanurzeń dla poszczególnych tokenów w tekście, zanim jednak FastText zsumuje wektory dla każdego słowa, są on dzielone przez ich normę L2, a następnie proces uśredniania bierze pod uwagę tylko wektory o dodatniej wartości normy.

Otrzymane wektory osadzeń dla każdego tekstu zostały potraktowane jako cechy obiektu, na podstawie których dokonywano klasyfikacji z wykorzystaniem klasyfikatora SVM (za jądro przekształcenia przyjęto radialną funkcję bazową). Sprawdzono różne wartości parametru regularyzacji, jednak gdy był on zbyt mały (co oznacza większą regularyzację), model przewidywał w każdym przypadku etykietę klasy większościowej. Wprawdzie osiągał on tym sposobem dokładność na poziomie 89% ze względu na wysoki poziom niezbalansowania danych, jednak inne miary jakości modelu, jak precyzja lub czułość, wynosiły zero (lub były niezdefiniowane ze względu na wartość zerową pojawiającą się w mianowniku wyrażenia). Najlepsze wyniki średnie z 5-foldowej kroswalidacji udało się uzyskać przyjmując wartość parametru $c = 10$ i zostały one przedstawione w Tabeli 1. Ponadto, aby zbadać wpływ balansu klas w danych zaimplementowany został algorytm podpróbkowywania przykładów z klasy większościowej w sposób losowy tak, aby w zbiorze treningowym stosunek klasy nieobraźliwej do obraźliwej wynosił 1:1 lub 2:1 (przypadki te zostały oznaczone w tabeli wyników jako US 1:1 oraz US 2:1). Analogiczne eksperymenty przeprowadzono używając wektorów zanurzeń z modelu BERT (konkretnie wersji HerBERT dla języka polskiego). Co zaskakujące, wyniki klasyfikacji były znacząco gorsze przy użyciu zwracanego przez model tokena CLS niż gdy zastosowano średni wektor osadzeń dla tokenów. Ostatecznie zrezygnowano z wykorzystanie tokena CLS, a wszystkie wyniki w Tabeli 1 odnoszą się do modelu HerBERT, w którym do reprezentacji tekstu wykorzystano uśredniony wektor dla wszystkich tokenów w tekście.

Biorąc pod uwagę większy wymiar wektorów osadzeń tekstu dla HerBERTa w porównaniu z FastText, zdecydowano sprawdzić, czy zastosowanie wielowarstwowego perceptronu jako

klasyfikatora pozwoli poprawić wyniki predykcji. Sprawdzono różne kombinacje wartości hiperparametrów i architektur sieci neuronowej. Najlepsze wyniki ze stratyfikowanej walidacji krzyżowej, które zostały umieszczone w tabeli, osiągnięto przy rozmiarach kolejnych warstw 768 (warstwa wejściowa), 300, 150, 2 (warstwa wyjściowa). Pomiędzy każą parą warstw dodano warstwę typu dropout z parametrem 0.5, a jako funkcję aktywacji wybrano funkcję Softplus. Model uczony był przy wykorzystaniu optymalizatora *Adam* ze współczynnikiem uczenia równym 0.0005, rozmiarem *mini-batch* równym 150, przez 60 epok. Za funkcję straty przyjęto ważoną entropię krzyżową, aby model zwracał większą uwagę na przypadki pozytywne (wagi 0.25 dla klasy większościowej (nieobraźliwej) oraz 0.9 dla klasy mniejszościowej (obraźliwej)). Podejście to zostało rozszerzone poprzez dodatkową augmentację danych: generowanie nowych, sztucznych przykładów dla klasy obraźliwej poprzez konkatenację tekstów z obydwu klas, który w rezultacie uznawano za obraźliwy. Zabieg ten umożliwił nieznaczną poprawę w precyzji modelu, spadła jednak jego czułość, co ostatecznie poskutkowało gorszym wynikiem miary F1.

Model	**Dokładność**	**Precyzja**	**Czułość**	**F1**	**Macro-F1**
słownikowy (*Hatebase*)	0.886	0.353	0.083	0.133	0.536
słownikowy (wulgaryzmy)	0.861	0.325	0.231	0.300	0.611
FT + SVM($c = 10$)	0.897	0.548	0.230	0.322	0.633
FT + SVM($c = 10$) + US 1:1	0.756	0.276	0.781	0.408	0.627
FT + SVM($c = 1$) + US 1:1	0.754	0.279	0.808	0.415	0.630
FT + SVM($c = 1$) + US 2:1	0.875	0.436	0.565	0.492	0.710
BERT + SVM($c = 10$)	0.893	0.005	0.400	0.009	0.476
BERT + SVM($c = 10$) + US 1:1	0.758	0.812	0.283	0.419	0.633
BERT + SVM($c = 1$) + US 1:1	0.684	0.831	0.231	0.361	0.575
BERT + SVM($c = 1$) + US 2:1	0.897	0.299	0.556	0.382	0.663
BERT + SVM($c = 3$) + US 2:1	0.868	0.605	0.422	0.497	0.710
BERT + NN	0.880	0.441	0.569	0.496	0.710
BERT + NN + DA	0.890	0.478	0.427	0.449	0.690

Tabela 1: Porównanie średnich wyników modeli tekstowych w 5-foldowej stratyfikowanej walidacji krzyżowej. Oznaczenia: FT - FastText, US X:Y - undersampling na zbiorze treningowym w stosunku X:Y klasy nieobraźliwej do obraźliwej, NN - sieć neuronowa, DA - augmentacja danych.

W Tabeli 1, podsumowującej wyniki wszystkich modeli, w każdej kolumnie kolorem czerwonym zaznaczono najmniejszą osiągniętą wartość danej miary, natomiast kolorem zielonym oznaczono wartości najwyższe. Poza wymienionymi modelami, testowano również wiele możliwych architektur i kombinacji hiperparametrów dla sieci neuronowej, która dokonywałaby klasyfikacji na podstawie wektorów FastText. W trakcie badania nie udało się jednak uzyskać zadowalających wyników, dlatego też modele te zostały pominięte w porównaniu.

Co zaskakujące, model słownikowy oparty na wyrazach z *Hatebase* nie wypada najgorzej – słabsze od niego wyniki osiągnął klasyfikator SVM działający na wektorach zanurzeń słów z HerBERTa. Jego wyniki są wyraźnie gorsze niż gdy zastosowano wektory z modelu FastText, należy jednak zwrócić uwagę na różnicę w wymiarach wektorów zwracanych przez oba modele. Zastosowany wariant FastText zwraca wektory o wymiarowości 300, podczas gdy dla HerBERTa są one niemalże trzykrotnie większe (768 wartości), zatem porównywanie wyników klasyfikatora SVM przy zachowaniu tych samych parametrów może być nie miarodajne. Warto także zauważyć, że zastosowanie metody *undersampling* na zbiorze treningowym powoduje wyraźny spadek w osiąganej na zbiorze testowym dokładności. Lepszym rozwiązaniem jest zachowanie proporcji w stosunku 2:1 przykładów oznaczonych jako nieobraźliwe do obraźliwych, co też jest bliższe rzeczywistym proporcjom. Przyjęcie takiego balansu pozwala uzyskać najlepszy wynik miary *macro F1-Score* zarówno w przypadku zanurzeń słów z modelu FastText, jak i HerBERTa.

W następnym kroku sprawdzony został model sekwencyjny – dwukierunkowy LSTM, który analizuje następstwo słów po sobie w tekście. Ze względu na ograniczone zasoby pamięci, konieczne było zastosowanie 100-wymiarowych wektorów osadzeń dla słów FastText [17]. Tekst został podzielony na tokeny przy użyciu tokenizatora z biblioteki *TensorFlow*, przyjmując rozmiar słownika równy 10 tysięcy słów. Dodatkowo teksty zostały przycięte do pierwszych 60 słów, w oparciu o wykres długości komentarzy widoczny na Rysunku 1. Zastosowano *padding* na końcu tekstu, nie ma to

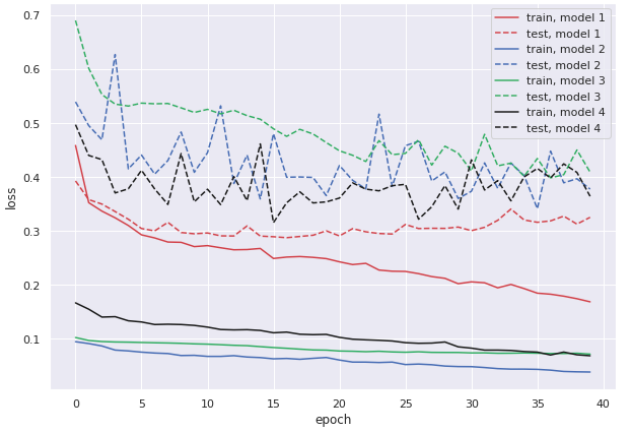
jednak wpływu na wyniki, ponieważ w uczeniu modeli wykorzystano maskowanie zer. Dla wszystkich modeli przyjęto jednakową architekturę, która składała się z poniższych:

- warstwa embedująca słowa do wektorów o wymiarze 100,
- warstwa dwukierunkowa LSTM o rozmiarze 50 z konkatenacją wyjścia (100-wymiarowy wektor na wyjściu),
- warstwa w pełni połączona o rozmiarze 32 z funkcją aktywacji *relu*,
- warstwa wyjściowa z sigmoidalną funkcją aktywacji.

W trakcie uczenia wykorzystano optymalizator *Adam* z parametrem zanikania wag (ang. *weight decay*) równym 10^{-6}, rozmiar *mini-batcha* ustalono na 250, a wszystkie modele uczone były przez 40 epok. Dla każdego z modeli przyjęto także wartości *dropout* i *recurrent dropout* jako 0.3. Za funkcję straty przyjęto binarną entropię krzyżową. Ze względu na długi czas uczenia modeli sekwencyjnych w tym przypadku nie stosowano walidacji krzyżowej, a zbiór danych został w sposób stratyfikowany podzielony na część treningową i testową w stosunku 80:20. Przy pomocy metody przeszukiwania siatki parametrów zbadano wiele możliwych kombinacji wartości hiperparametrów, w raporcie zaprezentowane zostaną jedynie cztery przykłady sieci BiLSTM, które umożliwią porównanie wpływu poszczególnych parametrów na proces uczenia modelu i jego jakość predykcji. Zbadane i przedstawione zostały wyniki modeli z następującymi parametrami:

- wersja 1: współczynnik uczenia równy 0.001,
- wersja 2: współczynnik uczenia równy 0.001, ważenie klas w funkcji straty - waga 0.1 dla klasy 0 (większościowej, przykłady nieobraźliwe) i 0.5 dla klasy 1 (mniejszościowej, przykłady obraźliwe),
- wersja 3: współczynnik uczenia równy 0.0001, ważenie klas w funkcji straty - waga 0.1 dla klasy 0 i 0.5 dla klasy 1,
- wersja 4: współczynnik uczenia równy 0.001, ważenie klas w funkcji straty - waga 0.2 dla klasy 0 i 0.8 dla klasy 1.

Przebieg procesu uczenia modeli w postaci zmian wartości funkcji straty na przestrzeni kolejnych epok został przedstawiony na Rysunku 4. Na zbiorze treningowych dla każdej z wersji modelu funkcja straty monotonicznie maleje, przy czy nachylenie krzywej jest największe w przypadku modelu niewykorzystującego ważenia klas, a najmniejsza dla modelu o rząd mniejszej wartości współczynnika uczenia.

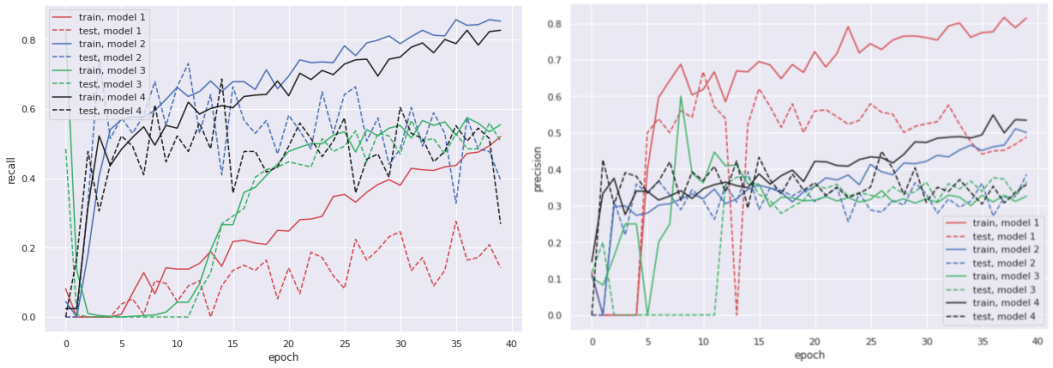


Rysunek 4: Wykres funkcji straty na zbiorze treningowym i testowym dla porównywanych wersji modeli BiLSTM w zależności od epoki uczenia.

Analizując wykresy funkcji straty na zbiorze testowym zauważamy ciekawe własności. Po pierwsze, wykres dla pierwszego z modeli (wersji, w której nie stosowano ważenia błędów) jest najbardziej gładki, wykazuje najmniejsze odchylenia, jednak model ten już od około dwudziestej epoki uczenia zaczyna się przeuczać – wartość funkcji straty zaczyna rosnąć na zbiorze testowym, podczas gdy

wciąż spada na treningowym, pomimo zastosowania *dropoutu*. Krzywe dla pozostałych modeli są znacznie bardziej postrzępione, ich przebieg jest nie przewidywalny, nieco mniejsze odchylenia obserwowane są dla modelu o mniejszym współczynniku uczenia. Pomimo tego wydaje się, że zachowana jest tendencja spadkowa.

W trakcie procesu uczenia badano również zmieniające się wartości miar precyzji i czułości, zarówno na zbiorze treningowym, jak i testowym, a wyniki przedstawiono na Rysunku 5. Na wykresie miary czułości (lewa część) widzimy, że najsłabiej wypada model bez ważenia klas – wprawdzie czułość na zbiorze treningowym osiąga około 50% na końcu procesu uczenia, jednak na zbiorze testowym oscyluje w granicach zaledwie 20%. Wyniki dla modeli 2 i 4, które różniły się jedynie wartościami wag, są do siebie bardzo zbliżone na obu zbiorach, można więcej przypuszczać, że drobne zmiany w przydzielonych wagach nie mają znaczącego wpływu na wyniki modelu. Czułość na zbiorze treningowym w obu przypadkach stale rośnie na przestrzeni 40 epok, jednak na zbiorze testowym obserwujemy tendencję spadkową po około 15 epokach. Co ciekawe, tym razem wahania na wykresach są podobne w przypadku modeli z wagami, jak i bez wag. Inaczej wygląda tylko wykres dla sieci BiLSTM ze zmniejszoną wartości parametru uczenia. Dla tej wersji krzywe czułości na zbiorze treningowym i testowym niemal się pokrywają. Obserwujemy dynamiczny wzrost pomiędzy 10 a 20 epoką uczenia, później jest on wolniejszy, jednak wahania są wyraźnie mniejsze w porównaniu z pozostałymi wersjami.



Rysunek 5: Wykresy wartości precyzji i czułości na zbiorze treningowym i testowym dla porównywanych wersji modeli BiLSTM w zależności od epoki uczenia.

Wnioski można wyciągnąć podobne, analizując wykres dla precyzji (prawa część Rysunku 5). Tutaj również uczenie jest najbardziej stabilne w przypadku modelu o mniejszym współczynniku uczenia, bowiem krzywe dla zbioru treningowego i testowego są sobie najbliższe. Nie obserwujemy jednak wzrostu tej miary na zbiorze testowym dla żadnej wersji modelu, wykresy dla modeli 2,3 i 4 oscylują wokół wartości 0.35. Dla modelu bez ważenia są wyraźnie wyższe, jednak bardziej niestabilne, dodatkowo wykazują również tendencję spadkową po 25 epokach uczenia. Warto zwrócić uwagę, analizując jednocześnie wykres czułości i wykres precyzji, że żaden z modeli sekwencyjnych nie osiąga lepszych wyników niż proste, wcześniej omawiane modele jak SVM. Być może konieczne byłoby przeprowadzenie większej liczby eksperymentów, aby znaleźć takie parametry, które pozwolą poprawić rezultaty. Powodem też może być zbyt mała liczby danych – zwróćmy uwagę, że modele sekwencyjne trenowane były na nie więcej niż 5000 tekstów, co przy tylu parametrach do optymalizacji może być zbyt małym zbiorem.

5.1 Analiza błędów

W celu analizy błędów popełnianych przez modele i identyfikacji przykładów problematycznych, wybrano stosunkowo najlepszy model wymieniony w Tabeli 1: połączenie wektorów FastText z klasyfikatorem SVM($c = 1$) i *undersamplingiem* 2:1. Zbiór danych został podzielony w sposób stratyfikowany w stosunku 80:20 na część treningową, na podstawie której uczony był model, i testową, która będzie poddawana analizie. Na takim podziale model osiągnął nieco lepsze wyniki niż przy walidacji krzyżowej, osiągając dokładność na poziomie 0.887, precyzję 0.506, czułość 0.619, F1-Score 0.557 oraz macro F1-Score 0.746.

Na zbiorze testowym zidentyfikowano 51 przykładów, które model ten niepoprawnie zaklasyfikował jako nieobraźliwe (*false negatives*) oraz 81 przykładów niepoprawnie uznanych za obraźliwe (*false positives*). Teksty, które zostały oznaczone przez nas jako nieobraźliwe, a które model niepoprawnie sklasyfikował jako obraźliwe, to m.in.:

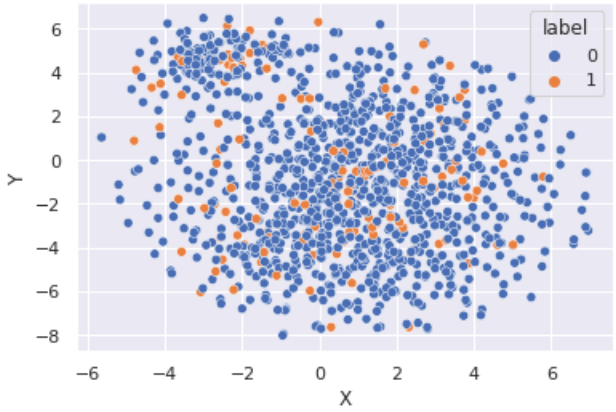
- „Marian meh prawdę mówiąc mam nieco wyrzuty sumienia bo kolejny dzień spędziłem na niczym, pora się kurwa ogarnąć",
- „najgorsze że to gówno odrasta zaraz, ja bym tego nie kosil w pisdu ale każą to czasem skosze",
- „chce Ci sie jebać z tą panierką? bo za pewne liczysz z niej kalorie",
- „nie no fotele gejmingowe to scam jak chuj jest xD",
- „wiem że to mem, ale mam nadzieję że nie karmisz kota kiełbasą, lubi tylko groszek taki dla kotów, z biedronki nie lubi a jest od niego uzależniony, bo tu nie ma co lubić, to jest straszne gówno".

Warto zwrócić uwagę, że w wielu z tych przykładów pojawiają się wulgaryzmy lub inne społecznie nieakceptowane słowa, które bardzo często pojawiają się przy okazji celowego obrażania innych. Zgodnie jednak z definicją, teksty te nie są na nikogo negatywnie nakierowane, nikogo nie obrażają, otrzymały więc negatywną etykietę. Ponieważ jednak używają niekulturalnego słownictwa, mogą sprawiać problemy modelom tekstowym. Z kolei teksty zaanotowane jako obraźliwe, które nie zostały uznane za takie przez model, to n.p.:

- „co robi murzyn na pasach - pojawia sie i znika dabum ts",
- „Jedynym debilem jesteś ty. Pringlesy to jest rozdrobniona masa ziemniaczana i jest to chrupek co masz na każdej tubie napisane. Chipsy są produktem z ciętego ziemniaka a nie z masy ziemniaczanej uformowanej na płasko",
- „widać i słychać że psychiczna, nie wiadomo czy by nawet jej psychiatryk pomogl",
- „wloncz w telefonie pogoda ten tydzien i spisz sobie gunwniarzu",
- „wmawiaj sobie te szanse na zmiane wmawiaj, tylko uważaj żebyś sie nie zesrał przypadkiem",
- „a twoja stara pierze w rzece".

Wymienione teksty obrażają grupę społeczną, osobę lub odbiorcę tekstu, w większości jednak nie używają tak oczywistych wulgaryzmów. Słowa takie jak *psychiczna* czy *stara* często występują w wypowiedziach jako neutralne, przykłady te mogły być zatem nieoczywiste dla modelu.

Biorąc pod uwagę błędy, jakie popełnił model oraz wartości osiąganych do tej pory miar, aby przekonać się, że zadanie faktycznie nie jest proste, stworzona została wizualizacja zbioru testowego przy pomocy algorytm t-SNE, przedstawiona na Rysunku 6.



Rysunek 6: Wizualizacja zbioru testowego przy pomocy algorytmu t-SNE.

Okazuje się, że powstała wizualizacja prezentuje mieszaninę punktów, w której niemożliwe jest wydzielenie odizolowanych grup przykładów pozytywnych. Zaskakującym jest, że model SVM radził sobie z tym problemem do tej pory najlepiej. Potwierdza to również tezę, że detekcja obraźliwych treści nie jest zadaniem prostym i najprawdopodobniej z tego względu pracę tę ręcznie wykonują moderatorzy, a nie oprogramowanie oparte na sztucznej inteligencji.

5.2 Sprawdzenie modeli na zbiorze PolEval

Dla języka polskiego istnieje tylko jeden zbiór danych z tej tematyki, który powstał na potrzeby PolEval 2019 [31]. Jest to zbiór ponad 10000 tweetów w zbiorze treningowym oraz ponad 1000 w zbiorze testowym. Dla zbioru zostały stworzone dwa zadania – klasyfikacji binarnej (tekst oznaczony jako szkodliwy, *harmful*, lub nie, *non-harmful*) oraz wieloklasowej (*cyberbulling*, *hate speech* lub żadno z nich). W zadaniu klasyfikacji binarnej zbalansowanie zbioru wynosi odpowiednio 8.48% dla części treningowej i 13.40% dla części testowej.

Etykiety powstały w wyniku głosowania większościowego wśród grupy anotatorów, a następnie zostały sprawdzone przez nadzorującego anotatora, który w przypadkach skrajnych miał prawo przypisania ostatecznej, zmienionej etykiety. Nie została podana wartość miary zgodności pomiędzy anotatorami. Został przedstawiony jedynie procent przypadków, w których anotatorzy byli zgodni (91.38%), jednak autorzy przyznają, że wynika to głównie ze zgodności dla przypadków oznaczonych negatywnie.

W konkursie obejmującym tylko zadanie klasyfikacji binarnej udział wzięło 14 zespołów, których wyniki na zbiorze testowym zostały przedstawione w tabeli na Rys. 7. Najlepszy z modeli, w klasyfikacji według uzyskanej miary F1-Score, uzyskał wynik na poziomie 0.59, najgorszy ze zgłoszonych modeli natomiast osiągnął wartość F1-Score tylko około 0.23. Postanowiono zbadać proponowane przez nas architektury modeli, aby sprawdzić, czy osiągną one lepsze wyniki na zbiorze PolEval niż na naszym autorskim.

Submission author(s)	Affiliation	Submitted system	Precision	Recall	F-score	Accuracy
Piotr Czapla, Marcin Kardas	**n-waves**	**n-waves ULMFiT**	**66.67%**	**52.24%**	**58.58%**	**90.10%**
Marcin Ciura	independent	Przetak	66.35%	51.49%	57.98%	90.00%
Tomasz Pietruszka	Warsaw University of Technology	ULMFiT + SentencePiece + BranchingAttention	52.90%	54.48%	53.68%	87.40%
Sigmoidal Team (Renard Korzeniowski, Przemysław Sadowski, Rafał Rolczyński, Tomasz Korbak, Marcin Możejko, Krystyna Gajczyk)	Sigmoidal	ensamble spacy + tpot + BERT	52.71%	50.75%	51.71%	87.30%
Sigmoidal Team	Sigmoidal	ensamble + fastai	52.71%	50.75%	51.71%	87.30%
Sigmoidal Team	Sigmoidal	ensenble spacy + tpot	43.09%	58.21%	49.52%	84.10%
Rafał Proński	CVTimeline	Rafal	41.08%	56.72%	47.65%	83.30%
Rafał Proński	CVTimeline	Rafal	41.38%	53.73%	46.75%	83.60%
Maciej Biesek	independent	model1-svm	60.49%	36.57%	45.58%	88.30%
Krzysztof Wróbel	AGH, UJ	fasttext	58.11%	32.09%	41.35%	87.80%
Katarzyna Krasnowska-Kieraś, Alina Wróblewska	IPI PAN	SCWAD-CB	51.90%	30.60%	38.50%	86.90%
Maciej Biesek	independent	model2-gru	63.83%	22.39%	33.15%	87.90%
Maciej Biesek	independent	model3-flair	81.82%	13.43%	23.08%	88.00%
Jakub Kuczkowiak	UWr	Task 6: Automatic cyberbullying detection	17.41%	32.09%	22.57%	70.50%

Rysunek 7: Wyniki zespołów biorących udział w zadaniu binarnym detekcji cyberprzemocy w ramach PolEval 2019.

Poza prostymi podejściami słownikowymi, sprawdzono również modele oparte na połączeniu wektorów FastText z klasyfikatorem SVM. Pod uwagę wzięto również możliwość dokonania *undersamplingu* na zbiorze treningowym. Jak widzimy w Tabeli 2, właśnie ten zabieg okazał się kluczowy do uzyskania zadowalających rezultatów. Większość prezentowanych przez nas modeli wykazała widoczną niestabilność, jeśli chodzi o miary precyzji i czułości – zazwyczaj jednak z nich była wysoka, podczas gdy druga była bliska zeru. Zastosowanie *undersamplingu* w stosunku 2:1 klasy *non-harmful* do *harmful* sprawiło, że obie miary wypadają podobnie i wynoszą odpowiednio 0.629

i 0.619. Plasuje to nasz model na pierwszym miejscu w klasyfikacji opartej na mierze F1, która w naszym przypadku wynosi 0.624. W jednej z prac [18] wykorzystano interesujące metody *oversamplingu*, wykorzystujące Tłumacz Google do parafrazowania tekstów z klasy mniejszościowej, nikt jednak nie wspominał o podejściu z wykorzystaniem *undersamplingu*. Zaskakują również końcowe wyniki w tabeli na Rys. 7, ponieważ trzy ostatnie proponowane modele osiągnęły gorsze wyniki niż zastosowanie deterministycznego algorytmu opartego na słowniku wulgaryzmów (patrz Tabela 2).

Model	Dokładność	Precyzja	Czułość	F1	Macro-F1
słownikowy (*Hatebase*)	0.863	0.429	0.067	0.116	0.521
słownikowy (wulgaryzmy)	0.880	0.625	0.261	0.368	0.651
FT + SVM($c = 10$)	0.886	0.857	0.179	0.296	0.617
FT + SVM($c = 1$) + US 1:1	0.841	0.446	0.769	0.564	0.733
FT + SVM($c = 1$) + US 2:1	0.900	0.629	0.619	0.624	0.783

Tabela 2: Porównanie wyników modeli tekstowych na zbiorze testowym PolEval.

6 Podejścia wielomodalne

Jedynie cechy tekstu mogą nie być wystarczające dla modelu, aby ocenić, czy tekst rzeczywiście jest obraźliwy. Aby człowiek mógł to jednoznacznie stwierdzić, często potrzebny jest kontekst wypowiedzi: kto mówi do kogo i w jakiej sytuacji, w odpowiedzi na co. W trakcie zbierania danych zwracaliśmy uwagę, aby zbierać nie tylko treść postów, ale także dane o autorach i odbiorcach, takie jak liczba udostępnionych linków, liczba postów na blogu, liczba komentarzy do innych użytkowników. Dodatkowo, z ponad 500,000 zebranych postów do anotacji wybrano te, które pozwolą stworzyć spójny graf pomiędzy użytkownikami, gdzie krawędź łącząca daną parę (autora i odbiorcę) to tekst kierowany od jednej osoby do drugiej. Graf ten więc może posiadać wiele krawędzi między jedną parą wierzchołków. Informacje te zostały wykorzystane, aby sprawdzić jakość modelu opartego na reprezentacji wielomodalnej: wykorzystującej nie tylko wektor osadzenia tekstu, ale też cechy autora i odbiorcy. Przykładowo, autorzy [32] pokazali, że już połączenie wektora reprezentacji tekstu z 4-elementowym wektorem *one-hot*, który będzie reprezentował relację między użytkownikami (kto kogo obserwuje), pozwala znacząco poprawić wyniki modelu.

W tej części sprawdzone zostały trzy możliwości wyboru cech, które wykorzystywane będą w zadaniu klasyfikacji. W każdej wersji zastosowano taki sam model, tj. sieć neuronową o architekturze jak wspominana w Rozdziale 5, a do reprezentacji tekstu zastosowano uśrednione po tokenach wektory z modelu HerBERT. Porównanie wyników zostało przedstawione w Tabeli 3, które w tym przypadku również pochodzą ze stratyfikowanej 5-foldowej walidacji krzyżowej. Pierwszy z modeli, bazowy, biorący pod uwagę tylko cechy obu rozmówców, bez reprezentacji tekstu, jak można było się spodziewać, nie poradził sobie z zadaniem. W każdym przypadku model trenowany na tych cechach przewidywał klasę większościową, a z powodu wysokie niezbalansowania danych uzyskał dokładność na poziomie 89%, jednak wartości miary precyzji i czułości wynoszą zero. Jednakże, połączenie tych cech z reprezentacją tekstu pozwoliło poprawić wyniki wcześniej prezentowanych modeli tekstowych. Wartości miary F1-Score są nieznacznie większe (osiągane macro F1-Score jest najwyższe do tej pory), jednak różnica może być statystycznie nieznacząca.

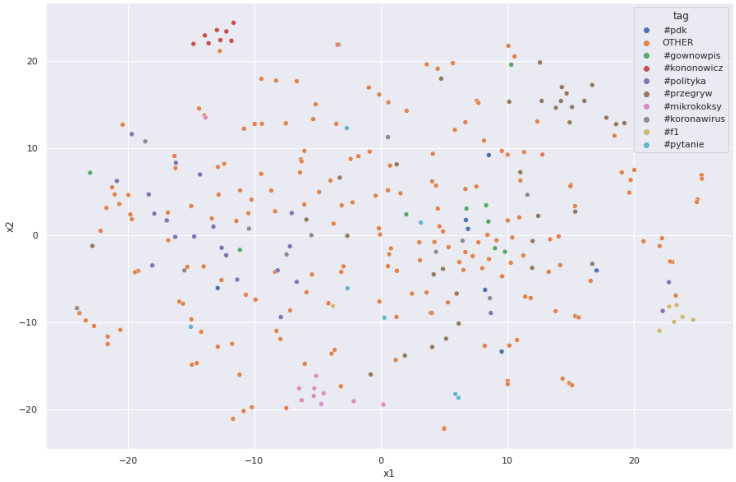
Wykorzystane cechy	Dokładność	Precyzja	Czułość	F1	Macro-F1
cechy użytkowników	0.890	0.000	0.000	0.000	0.470
tekst + cechy użytkowników	0.880	0.445	0.583	0.504	0.720
tekst + wektory node2vec	0.890	0.483	0.511	0.495	0.720

Tabela 3: Porównanie wyników modeli wielomodalnych.

Ostatni z wykorzystanych modeli wielomodalnych opierał się na reprezentacji użytkowników w sieci społecznej stworzonej przy pomocy algorytmu *node2vec*. Na podstawie komentarzy pisanych przez użytkowników w odpowiedzi do postów innych osób, stworzony został skierowany graf prezentujący relacje między internautami. Z każdego wierzchołka poprowadzonych zostało 100 spacerów losowych o długości 16, które później analizowane były oknem o szerokości 10. W procesie uczenia modelu, dla każdego wierzchołka w sieci (użytkownika serwisu) stworzony został 32-wymiarowy *embedding*, który odwzorowuje rolę i pozycję danej osoby w grafie. Tak przygotowane wektory dla autora

i odbiorcy tekstu zostały skonkatenowane z reprezentację tekstu, aby wytrenować model klasyfikujący. Jego wyniki okazały się porównywalne z modelem opartym na cechach internautów, co widać w Tabeli 3.

Aby dodatkowo sprawdzić, czy wektory stworzone algorytmem *node2vec* są rzeczywiście reprezentatywne, stworzona została dwuwymiarowa wizualizacja t-SNE, rzutująca wektory na dwuwymiarową płaszczyznę. Dodatkowo, dla każdego internauty wyznaczony został tag, który najczęściej pojawiał się w publikowanych przez niego tekstach. Na wykresie kolororem zostały oznaczone najpopularniejsze z tagów, te rzadziej pojawiające się zostały zebrane w jedną kategorię *other*. Stworzone reprezentacje, zwizualizowane przy pomocy tego algorytmu, wydają się sensowne – jesteśmy w stanie wyróżnić grupy użytkowników, którzy udzielają się w tych samych tagach (jak np. *#kononowicz*, *#polityka*, *#przegryw*), a ich reprezentacje są zbliżone.



Rysunek 8: Wizualizacja reprezentacji *node2vec* użytkowników przy pomocy algorytmu t-SNE wraz z oznaczeniem najczęściej używanego tagu.

7 Podsumowanie

Zarówno przegląd literatury jak i wykonany projekt pokazał, że zadanie klasyfikacji tekstów z obszaru zagadnień subiektywnych, jakim jest obraźliwość, nie jest zadaniem łatwym. Stąd też próby wykorzystania innych modalności oprócz tekstu, aby wspomóc działanie systemu. Na potrzeby projektu zostały przebadane różne modele, zarówno jednomodalne oparte o tekst, jak również wielomodalne uwzględniające cechy użytkowników.

Użycie modeli głębokich pozwoliło uzyskać przedstawione wyniki zarówno dzięki uczeniu już samego zadania klasyfikacji, ale również przygotowania embeddingów - wykorzystane modele wektorów osadzeń były uczone w oparciu o sieci głębokie.

Wyniki dla badanego zagadnienia są zadowalające. Otrzymane wartości metryk nie są wartościami, jakich można by się spodziewać po klasyfikacji binarnej, jednak należy pamiętać o subiektywnym charakterze zagadnienia rozpoznawania obraźliwości. Na jakość klasyfikacji wpływa również jakość danych - źródło danych z którego korzystaliśmy posiada pewną specyfikę - obecność sporej ilości literówek, błędów - często robionych umyślnie, znacznie utrudnia klasyfikacje. Na wynik wpływa również niezbalansowanie zbioru. Dzięki wykorzystaniu metody *undersamplingu* udało się zmniejszyć niezbalansowanie i poprawić rezultaty. Badanie podejścia zostały również zweryfikowane przez zestawienie wyników z wynikami na danych z konkursu PolEval i model osiągnął rezultaty porównywalne do wystawionych w konkursie modeli.

Sam proces anotacji danych oraz późniejsza dyskusja nad wynikami pokazała subiektywność obraźliwości treści, a poprawa wyników przy użyciu cech multimodalnych pozwalają sądzić, że personalizacja modeli do predykcji zagadnień jest istotna.

Literatura

[1] A. Alrehili. Automatic hate speech detection on social media: A brief survey. In *2019 IEEE/ACS 16th International Conference on Computer Systems and Applications (AICCSA)*, pages 1–6, 2019.

[2] S. S. Aluru, B. Mathew, P. Saha, and A. Mukherjee. Deep learning models for multilingual hate speech detection, 2020.

[3] V. Basile, C. Bosco, E. Fersini, N. Debora, V. Patti, F. M. R. Pardo, P. Rosso, M. Sanguinetti, et al. Semeval-2019 task 5: Multilingual detection of hate speech against immigrants and women in twitter. In *13th International Workshop on Semantic Evaluation*, pages 54–63. Association for Computational Linguistics, 2019.

[4] P. J. Breckheimer. A haven for hate: the foreign and domestic implications of protecting internet hate speech under the first amendment. *S. Cal. L. Rev.*, 75:1493, 2001.

[5] A. Brown. What is so special about online (as compared to offline) hate speech? *Ethnicities*, 18(3):297–326, 2018.

[6] Y. Chen, Y. Zhou, S. Zhu, and H. Xu. Detecting offensive language in social media to protect adolescent online safety. In *2012 International Conference on Privacy, Security, Risk and Trust and 2012 International Confernece on Social Computing*, pages 71–80, 09 2012.

[7] T. Davidson, D. Warmsley, M. Macy, and I. Weber. Automated hate speech detection and the problem of offensive language. In *Proceedings of the Eleventh International AAAI Conference on Web and Social Media (ICWSM 2017)*, pages 512–515. Association for the Advancement of Artificial Intelligence, 2017.

[8] N. Djuric, J. Zhou, R. Morris, M. Grbovic, V. Radosavljevic, and N. Bhamidipati. Hate speech detection with comment embeddings. In *Proceedings of the 24th International Conference on World Wide Web*, WWW '15 Companion, page 29–30. Association for Computing Machinery, 2015.

[9] P. Fortuna and S. Nunes. A survey on automatic detection of hate speech in text. *ACM Computing Surveys (CSUR)*, 51:1 – 30, 2018.

[10] R. Gomez, J. Gibert, L. Gomez, and D. Karatzas. Exploring hate speech detection in multimodal publications. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 1459–1467, March 2020.

[11] S. J. Heyman. Hate speech, public discourse, and the first amendment. *Oxford University Press, Forthcoming*, 2008.

[12] H. Hosseinmardi, S. A. Mattson, R. I. Rafiq, R. Han, Q. Lv, and S. Mishra. Detection of cyberbullying incidents on the instagram social network, 2015.

[13] X. Huang, L. Xing, F. Dernoncourt, and M. J. Paul. Multilingual Twitter corpus and baselines for evaluating demographic bias in hate speech recognition. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 1440–1448, Marseille, France, May 2020. European Language Resources Association.

[14] M. Ibrahim, M. Torki, and N. El-Makky. Alexu-backtranslation-tl at semeval-2020 task [12]: Improving offensive language detection using data augmentation and transfer learning. In *Proceedings of the International Workshop on Semantic Evaluation (SemEval)*, 07 2020.

[15] M. O. Ibrohim and I. Budi. Multi-label hate speech and abusive language detection in Indonesian Twitter. In *Proceedings of the Third Workshop on Abusive Language Online*, pages 46–57, Florence, Italy, Aug. 2019. Association for Computational Linguistics.

[16] J. B. Jacobs. Hate crime: Criminal law and identity politics: Author's summary. *Theoretical Criminology*, 6(4):481–484, 2002.

[17] J. Kocoń and M. Gawor. Evaluating KGR10 Polish word embeddings in the recognition of temporal expressions using BiLSTM-CRF. *Schedae Informaticae*, 27, 2018.

[18] K. Krasnowska-Kieraś and A. Wróblewska. A simple neural networkfor cyberbullying detection. In *Proceedings of the PolEval 2019 Workshop*, pages 161–163. Institute of Computer Science, Polish Academy of Sciences, 2019.

[19] J. Landis and G. Koch. The measurement of observer agreement for categorical data. *Biometrics*, 33 1:159–74, 1977.

[20] T. Mandl, S. Modha, P. Majumder, D. Patel, M. Dave, C. Mandlia, and A. Patel. Overview of the HASOC Track at FIRE 2019: Hate speech and offensive content identification in indo-european languages. In *Proceedings of the 11th Forum for Information Retrieval Evaluation*, FIRE '19, page 14–17, New York, NY, USA, 2019. Association for Computing Machinery.

[21] A. Misiaszek, P. Kazienko, M. Kulisiewicz, Ł. Augustyniak, W. Tuligłowicz, A. Popiel, and T. Kajdanowicz. Belief propagation method for word sentiment in wordnet 3.0. In *Asian Conference on Intelligent Information and Database Systems*, pages 263–272. Springer, 2014.

[22] M. Mozafari, R. Farahbakhsh, and N. Crespi. A bert-based transfer learning approach for hate speech detection in online social media. In *International Conference on Complex Networks and Their Applications*, pages 928–940. Springer, 2019.

[23] H. Mulki, H. Haddad, C. Bechikh Ali, and H. Alshabani. L-HSAB: A Levantine Twitter dataset for hate speech and abusive language. In *Proceedings of the Third Workshop on Abusive Language Online*, pages 111–118, Florence, Italy, Aug. 2019. Association for Computational Linguistics.

[24] A. Nikolov and V. Radivchev. Nikolov-radivchev at semeval-2019 task 6: Offensive tweet classification with bert and ensembles. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 691–695, 2019.

[25] C. Nobata, J. Tetreault, A. Thomas, Y. Mehdad, and Y. Chang. Abusive language detection in online user content. In *Proceedings of the 25th International Conference on World Wide Web*, WWW '16, page 145–153. International World Wide Web Conferences Steering Committee, 2016.

[26] E. W. Pamungkas, V. Basile, and V. Patti. Misogyny detection in twitter: a multilingual and cross-domain study. *Information Processing & Management*, 57(6):102360, 2020.

[27] J. H. Park and P. Fung. One-step and two-step classification for abusive language detection on Twitter. In *Proceedings of the First Workshop on Abusive Language Online*, pages 41–45, Vancouver, BC, Canada, Aug. 2017. Association for Computational Linguistics.

[28] J. C. Pereira-Kohatsu, L. Quijano-Sánchez, F. Liberatore, and M. Camacho-Collados. Detecting and monitoring hate speech in twitter. *Sensors (Basel, Switzerland)*, 19(21), October 2019.

[29] F. Poletto, V. Basile, M. Sanguinetti, C. Bosco, and V. Patti. Resources and benchmark corpora for hate speech detection: a systematic review. In *LREC 2020*, 2020.

[30] R. Prońko. Simple bidirectional lstm solution for text classification. In *Proceedings of the PolEval 2019 Workshop*, pages 111–119. Institute of Computer Science, Polish Academy of Sciences, 05 2019.

[31] M. Ptaszyński, A. Pieciukiewicz, and P. Dybala. Results of the poleval 2019 shared task 6: First dataset and open shared task for automatic cyberbullying detection in polish twitter. In *Proceedings of the PolEval 2019 Workshop*, pages 89–110. Institute of Computer Science, Polish Academy of Sciences, 2019.

[32] B. Radfar, K. Shivaram, and A. Culotta. Characterizing variation in toxic language by social context. In *Proceedings of the Fourteenth International AAAI Conference on Web and Social Media, ICWSM 2020, Held Virtually, Original Venue: Atlanta, Georgia, USA, June 8-11, 2020*, pages 959–963. AAAI Press, 2020.

[33] E. Raisi and B. Huang. Weakly supervised cyberbullying detection using co-trained ensembles of embedding models. In *2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pages 479–486, 08 2018.

[34] A. H. Razavi, D. Inkpen, S. Uritsky, and S. Matwin. Offensive language detection using multi-level classification. In A. Farzindar and V. Kešelj, editors, *Advances in Artificial Intelligence*, pages 16–27, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg.

[35] M. Ridenhour, A. Bagavathi, E. Raisi, and S. Krishnan. Detecting online hate speech: Approaches using weak supervision and network embedding models, 2020.

[36] M. Sahlgren, T. Isbister, and F. Olsson. Learning representations for detecting abusive language. In *Proceedings of the 2nd Workshop on Abusive Language Online (ALW2)*, pages 115–123, Brussels, Belgium, Oct. 2018. Association for Computational Linguistics.

[37] M. Sanguinetti, F. Poletto, C. Bosco, V. Patti, and M. Stranisci. An Italian Twitter corpus of hate speech against immigrants. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, pages 2798–2805, Miyazaki, Japan, May 2018. European Language Resources Association (ELRA).

[38] A. Schmidt and M. Wiegand. A survey on hate speech detection using natural language processing. In *Proceedings of the Fifth International Workshop on Natural Language Processing for Social Media*, pages 1–10, Valencia, Spain, Apr. 2017. Association for Computational Linguistics.

[39] Y. Senarath and H. Purohit. Evaluating semantic feature representations to efficiently detect hate intent on social media. In *2020 IEEE 14th International Conference on Semantic Computing (ICSC)*, pages 199–202, 2020.

[40] V. K. Singh, S. Ghosh, and C. Jose. Toward multimodal cyberbullying detection. In *Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems*, CHI EA '17, page 2090–2099. Association for Computing Machinery, 2017.

[41] S. Suryawanshi, B. R. Chakravarthi, M. Arcan, and P. Buitelaar. Multimodal meme dataset (MultiOFF) for identifying offensive content in image and text. In *Proceedings of the Second Workshop on Trolling, Aggression and Cyberbullying*, pages 32–41, Marseille, France, May 2020. European Language Resources Association (ELRA).

[42] B. Vidgen and L. Derczynski. Directions in abusive language training data: Garbage in, garbage out, 2020.

[43] Z. Waseem. Are you a racist or am I seeing things? annotator influence on hate speech detection on Twitter. In *Proceedings of the First Workshop on NLP and Computational Social Science*, pages 138–142, Austin, Texas, Nov. 2016. Association for Computational Linguistics.

[44] Z. Waseem and D. Hovy. Hateful symbols or hateful people? predictive features for hate speech detection on Twitter. In *Proceedings of the NAACL Student Research Workshop*, pages 88–93, San Diego, California, June 2016. Association for Computational Linguistics.

[45] M. Wiegand, M. Siegel, and J. Ruppenhofer. Overview of the germeval 2018 shared task on the identification of offensive language. In *Proceedings of GermEval 2018, 14th Conference on Natural Language Processing (KONVENS 2018)*, pages 1–10, 2018.

[46] F. Yang, X. Peng, G. Ghosh, R. Shilon, H. Ma, E. Moore, and G. Predovic. Exploring deep multimodal fusion of text and photo for hate speech classification. In *Proceedings of the Third Workshop on Abusive Language Online*, pages 11–18, Florence, Italy, Aug. 2019. Association for Computational Linguistics.

[47] H. Yenala, A. Jhanwar, M. Chinnakotla, and J. Goyal. Deep learning for detecting inappropriate content in text. *International Journal of Data Science and Analytics*, 12 2017.

[48] M. Zampieri, S. Malmasi, P. Nakov, S. Rosenthal, N. Farra, and R. Kumar. Predicting the type and target of offensive posts in social media. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1415–1420, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[49] M. Zampieri, S. Malmasi, P. Nakov, S. Rosenthal, N. Farra, and R. Kumar. SemEval-2019 task 6: Identifying and categorizing offensive language in social media (OffensEval). In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 75–86, Minneapolis, Minnesota, USA, June 2019. Association for Computational Linguistics.

[50] H. Zhong, H. Li, A. Squicciarini, S. Rajtmajer, C. Griffin, D. Miller, and C. Caragea. Content-driven detection of cyberbullying on the instagram social network. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, IJCAI'16, page 3952–3958. AAAI Press, 2016.