Politechnika Warszawska

WYDZIAŁ ELEKTRONIKI
I TECHNIK INFORMACYJNYCH

Instytut Informatyki

Praca dyplomowa magisterska

na kierunku Informatyka
w specjalności Sztuczna Inteligencja

Uczenie ze wzmocnieniem modeli generatywnych na podstawie danych sprzedażowych

Michał Wiszenko
Numer albumu 300285

promotor
dr hab. inż. Mariusz Kamola

WARSZAWA 2025

Uczenie ze wzmocnieniem modeli generatywnych na podstawie danych sprzedażowych

Streszczenie. Szybki rozwój modeli językowych opartych na sztucznej inteligencji stawia przed badaczami wyzwanie ich optymalizacji pod kątem zgodności z ludzkimi preferencjami, co jest kluczowe w kreatywnych zadaniach, takich jak generowanie nazw domen internetowych. Przemyślany wybór nazwy domeny wpływa na rozpoznawalność i profesjonalny wizerunek w internecie, jednak proces ten jest skomplikowany i wymaga uwzględnienia wielu czynników.

Niniejsza praca magisterska omawia użycie metody uczenia ze wzmocnieniem z informacją zwrotną (RLHF) oraz jej alternatyw jako mechanizmów dopasowania modeli generatywnych do specyfiki tego zadania. W pracy zbadano i porównano trzy wiodące metody uczenia preferencyjnego – RLHF, Direct Preference Optimization (DPO) oraz Offline Reinforcement Learning Policy Optimization (ORPO). Wykorzystano rzeczywisty zbiór danych obejmujący opisy stron internetowych i odpowiadające im zarejestrowane domeny, uzupełniony o sygnał sprzedażowy, co umożliwiło strojenie pod preferencje rynkowe i obiektywne porównanie metod. Do optymalizacji wykorzystano model oparty na architekturze BART, a jakość generowanych nazw oceniono za pomocą metryki odległości Levenshteina.

Eksperymenty wykazały, że metody RLHF i DPO osiągają porównywalne, wysokie rezultaty, jednak to DPO wyróżnia się optymalną równowagą między niską złożonością obliczeniową a wysoką jakością wyników. W przypadku metody ORPO zaobserwowano, że do uzyskania podobnej skuteczności konieczne byłoby wykorzystanie znacznie większego zbioru danych. Wyniki te sugerują, że DPO jest najbardziej obiecującym podejściem do automatyzacji procesu tworzenia nazw domen internetowych.

Słowa kluczowe: sztuczna inteligencja, uczenie maszynowe, uczenie ze wzmocnieniem, modele generatywne, domeny internetowe, dane sprzedażowe, RLHF, DPO, ORPO, optymalizacja preferencji

Reinforcement Learning for Generative Models Based on Sales Data

Abstract. The rapid development of language models presents a challenge for researchers in optimizing them for human preferences, which is crucial in creative tasks such as generating internet domain names. A well-chosen domain name impacts brand recognition and online presence, yet the generation process is complex and multifaceted.

This master's thesis explores the use of Reinforcement Learning from Human Feedback (RLHF) and its alternatives as mechanisms for aligning generative models to this specific task. The study investigates and compares three leading preference learning methods—RLHF, Direct Preference Optimization (DPO), and Offline Reinforcement Learning Policy Optimization (ORPO). The research uses a real-world dataset of website descriptions and corresponding registered domains, augmented with a sales signal, enabling market-aligned tuning and objective comparison of the methods. A BART-based architecture was optimized, and the quality of generated names was assessed using the Levenshtein distance metric.

The experiments demonstrated that RLHF and DPO achieve comparable, high performance, with DPO offering an optimal balance between low computational complexity and high-quality results. The ORPO method was found to require a significantly larger dataset to achieve similar effectiveness. These findings suggest that DPO is the most promising approach for automating the process of internet domain name creation.

Keywords: artificial intelligence, machine learning, reinforcement learning, generative models, internet domains, sales data, RLHF, DPO, ORPO, preference optimization

Spis treści

1. Wstęp 7
1.1. Tło i motywacja 7
1.2. Cel i zakres pracy 7
1.2.1. Przykład działania modelu generującego nazwy domen internetowych 8
1.3. Przegląd literatury 8
1.3.1. Rozwój architektur generatywnych 8
1.3.2. Ewolucja paradygmatów uczenia maszynowego 9
1.3.3. Współczesne kierunki w strojeniu modeli 9
1.3.4. Standardy ewaluacji w badaniach nad generacją tekstu 10
1.3.5. Generacja nazw i uczenie na danych biznesowych 11
1.4. Struktura pracy 11
2. Fundamenty teoretyczne optymalizacji modeli językowych 13
2.1. Architektury generatywnych modeli językowych 13
2.1.1. Architektura Transformer 13
2.1.2. Model BART i jego polska adaptacja 13
2.1.3. Modele oparte na architekturze decoder-only 14
2.2. Uczenie ze wzmocnieniem 15
2.3. Uczenie ze wzmocnieniem z informacją zwrotną 17
2.4. Alternatywne metody uczenia preferencyjnego 18
2.4.1. Metoda Direct Preference Optimization 18
2.4.2. Metoda Offline Reinforcement Learning Policy Optimization 18
2.4.3. Metoda Reinforcement Learning from AI Feedback 19
2.5. Miary jakości w ewaluacji modeli generatywnych 19
2.5.1. Metryki automatyczne oparte na podobieństwie 19
2.5.2. Wewnętrzne metryki treningowe 20
2.6. Proces rejestracji domen internetowych 21
3. Metodyka i środowisko badawcze 22
3.1. Definicja zadania badawczego 22
3.2. Zbiór danych i jego przygotowanie 23
3.2.1. Dane do dostrajania nadzorowanego 24
3.2.2. Dane preferencyjne 25
3.3. Architektura i implementacja modeli 26
3.3.1. Model generujący: Polish BART 26
3.3.2. Model nagrody: Bielik-1.5B 26
3.3.3. Model referencyjny 27
3.4. Proces treningu i ewaluacji 27
3.4.1. Przygotowanie środowiska treningowego 27

3.4.2. Proces ewaluacji modeli 27
4. Prezentacja i analiza wyników 29
4.1. Wyniki ilościowe 29
4.1.1. Wyniki dla podejścia RLHF 29
4.1.2. Wyniki dla podejścia DPO 29
4.1.3. Wyniki dla podejścia ORPO 31
4.2. Analiza jakościowa 32
4.3. Dyskusja 35
4.3.1. Istotność danych sprzedażowych 36
5. Podsumowanie i wnioski 37
5.1. Wnioski końcowe 37
5.2. Znaczenie praktyczne i implikacje zastosowań 37
5.3. Ograniczenia pracy i kierunki dalszych badań 38
Bibliografia 41
Spis rysunków 44
Spis tabel 44

1. Wstęp

1.1. Tło i motywacja

W ostatnich latach uczenie maszynowe (ang. Machine Learning, ML) stało się fundamentem wielu nowoczesnych rozwiązań informatycznych. Wraz z dynamicznym rozwojem tej dziedziny poszukiwane są nowe rozwiązania pozwalające na udoskonalenie dotychczasowych procesów treningu modeli sztucznej inteligencji (ang. Artificial Intelligence, AI). Tradycyjne podejścia są często niewystarczające, gdyż wymagają precyzyjnie oznaczonych danych i nie pozwalają na efektywne uwzględnienie sytuacji, w których decyzje podejmowane są kolejno, a każda z nich wpływa na dalszy przebieg procesu.

Jednym z interesujących zastosowań metod uczenia maszynowego jest automatyczna generacja nazw domen internetowych na podstawie opisów działalności potencjalnych klientów. Przemyślany wybór nazwy domeny wpływa na zwiększoną rozpoznawalność, wiarygodność oraz profesjonalny wizerunek w środowisku internetowym, co jest istotnym czynnikiem zarówno dla przedsiębiorstw, jak i indywidualnych użytkowników.

Generowanie trafnych nazw domen stanowi jednak wyzwanie ze względu na konieczność uwzględnienia wielu czynników, takich jak związek z charakterem działalności, łatwość zapamiętania czy unikalność. Kluczową trudnością pozostaje dopasowanie wyników generowanych przez modele do ludzkich preferencji i wartości. W tym kontekście istotną rolę odgrywa metoda uczenia ze wzmocnieniem z informacją zwrotną (ang. Reinforcement Learning with Human Feedback, RLHF), która rozwiązuje ten problem poprzez wykorzystanie ludzkiej oceny jako dodatkowy sygnał do optymalizacji polityki modeli.

1.2. Cel i zakres pracy

Głównym celem niniejszej pracy dyplomowej jest zbadanie, implementacja i ocena skuteczności nowoczesnych metod uczenia preferencyjnego w zadaniu optymalizacji dużych modeli językowych do generowania nazw domen internetowych. Praca ma na celu wyjście poza standardowe zastosowania modeli generatywnych i sprawdzenie, jak zaawansowane techniki, takie jak uczenie ze wzmocnieniem z informacją zwrotną, radzą sobie w tym niszowym, ale kreatywnym zadaniu. Dąży ona do udowodnienia, że możliwe jest skuteczne dotrenowanie modelu preferencji na podstawie historycznych danych sprzedażowych, co stanowi alternatywę dla kosztownych i subiektywnych ocen ludzkich.

Zakres pracy obejmuje przeprowadzenie pełnego procesu badawczego, w jego ramach:

- Wykorzystano polskojęzyczny model generatywny dotrenowany w zadaniu generacji nazw domen jako architekturę bazową.
- Zaimplementowano i porównano trzy kluczowe techniki optymalizacji preferencyjnej: RLHF, DPO oraz ORPO.
- Użyto autentycznego zbioru danych do treningu i ewaluacji, co pozwoliło na obiektywną ocenę praktycznej użyteczności modeli.

- Dokonano ewaluacji jakościowej i ilościowej (z użyciem miary odległości Levenshteina), aby ocenić skuteczność poszczególnych podejść.
- Przeanalizowano w jakim stopniu metody uczenia ze wzmocnieniem mogą poprawić jakość modeli generatywnych przy treningu na danych sprzedażowych.

Praca została zrealizowana w ścisłej współpracy z NASK (Naukowa i Akademicka Sieć Komputerowa – Państwowy Instytut Badawczy), który pełni rolę krajowego rejestru domen internetowych. Współpraca ta umożliwiła dostęp do unikalnego, rzeczywistego zbioru danych, zawierającego nazwy zarejestrowanych domen internetowych i powiązanych z nimi opisów działalności, jak również dostęp do historycznych danych sprzedażowych. Dzięki temu możliwe stało się przeprowadzenie badań na prawdziwych przykładach, co przekłada się na praktyczną wartość wyciąganych wniosków.

1.2.1. Przykład działania modelu generującego nazwy domen internetowych

Aby zilustrować zadanie, rozważmy przykładowy opis działalności: "Sklep internetowy z ręcznie robioną biżuterią z naturalnych kamieni". Celem modelu jest zaproponowanie krótkiej, kreatywnej i łatwej do zapamiętania nazwy domeny, jednocześnie unikającej zbędnych separatorów, np. `kamienne-inspiracje.pl`, `naturalna-bizuteria.pl`, `kamykart.pl`. W fazie SFT model uczy się zgodności tematycznej (biżuteria, kamienie, rękodzieło), a w fazie optymalizacji preferencyjnej wzmacniane są propozycje, które lepiej wpisują się w preferencje użytkowników i sygnał rynkowy (np. krótkość, brak nietypowych znaków, naturalne złożenie słów, większa kreatywność).

1.3. Przegląd literatury

Zadanie generacji nazw domen internetowych wpisuje się w szerszy nurt badań nad generowaniem tekstu przez duże modele językowe (ang. Large Language Models, LLM). Rozwój w tej dziedzinie jest bezpośrednio powiązany z ewolucją architektur generatywnych, dostępnością modeli dostosowanych do konkretnych języków oraz metodami optymalizacji i ewaluacji wyników.

1.3.1. Rozwój architektur generatywnych

Podstawą współczesnych modeli do zadań generacyjnych jest architektura Transformer [1], a w szczególności jej warianty typu enkoder-dekoder. Model BART (Bidirectional and Auto-Regressive Transformers) stał się jednym z najistotniejszych w tej dziedzinie. Jego zdolność do przetwarzania sekwencji wejściowej w sposób dwukierunkowy oraz auto-regresywnego generowania nowej sekwencji sprawia, że jest bardzo dobrze przystosowany do zadań wymagających zrozumienia kontekstu wejściowego i generacji na jego podstawie nowych treści [2].

Obok architektur enkoder-dekoder, znaczną popularność zdobyły modele oparte wyłącznie na dekoderze. Przykładem jest Mistral-7B [3], model typu open-source, który mimo relatywnie niewielkiej liczby parametrów (7 miliardów) dorównuje wydajnością znacznie

większym modelom dzięki innowacjom takim jak metody Grouped-Query Attention czy Sliding-Window Attention.

Efektywność modeli językowych jest również bardzo uzależniona od ich dopasowania do specyfiki danego języka. Z tego powodu kluczową rolę w badaniach nad przetwarzaniem języka polskiego odgrywają modele wstępnie wytrenowane na dużych, polskojęzycznych korpusach. W niniejszej pracy jako model bazowy wykorzystano wersję modelu BART dostosowaną do języka polskiego [4], co zapewniło wysokie zrozumienie kontekstu wejściowego. Z kolei jako zaawansowany model oceniający wykorzystano Bielik-1.5B [5] z rodziny polskojęzycznych modeli o architekturze decoder-only, jednocześnie pozwalając na ewaluację generowanych nazw w kontekście języka polskiego.

1.3.2. Ewolucja paradygmatów uczenia maszynowego

Podstawą teoretyczną dla zaawansowanych metod optymalizacji modeli językowych jest uczenie ze wzmocnieniem (ang. Reinforcement Learning, RL), czyli paradygmat uczenia maszynowego, w którym algorytmy uczą się podejmowania optymalnych decyzji poprzez interakcję ze środowiskiem w celu maksymalizacji sygnału nagrody. Fundamentalne koncepcje tej dziedziny, stanowiące punkt wyjścia dla technik takich jak RLHF, zostały szeroko opisane w pracach naukowych, w tym w kluczowej publikacji z 1996 roku [6].

Prawdziwy przełom w praktycznych zastosowaniach nastąpił jednak przy połączeniu uczenia ze wzmocnieniem z głębokimi sieciami neuronowymi, co dało początek dziedzinie głębokiego uczenia ze wzmocnieniem (ang. Deep Reinforcement Learning, DRL) [7]. Podejście to umożliwiło rozwiązywanie jeszcze bardziej złożonych problemów.

1.3.3. Współczesne kierunki w strojeniu modeli

Samo wstępne wytrenowanie modelu nie gwarantuje, że generowane przez niego treści będą zgodne z oczekiwaniami, wartościami czy preferencjami użytkowników. W odpowiedzi na to wyzwanie opracowano metodę uczenia ze wzmocnieniem z informacją zwrotną od człowieka (ang. Reinforcement Learning from Human Feedback, RLHF) [8]. Ta technika pozwala na precyzyjne dostrojenie modelu z wykorzystaniem ludzkich ocen jako sygnału nagrody. Kluczowym elementem RLHF jest optymalizacja polityki modelu, często realizowana za pomocą algorytmów takich jak Proximal Policy Optimization (PPO) [9], które stabilizują proces treningu i zapobiegają drastycznym zmianom w zachowaniu modelu i utratę jakości w zadaniu bazowym.

Jednakże złożoność obliczeniowa i implementacyjna procesu RLHF skłoniła badaczy do poszukiwania prostszych, lecz równie skutecznych alternatyw. Do najpopularniejszych należą:

- Direct Preference Optimization (DPO) [10] - eliminuje potrzebę trenowania osobnego modelu nagrody, zamiast tego bezpośrednio optymalizując politykę modelu na podstawie par preferencji. To uproszczenie znacząco redukuje zasoby potrzebne

do treningu i stabilizuje cały proces, co czyni DPO główną metodą optymalizacji w niniejszym projekcie.
- Offline RL Policy Optimization (ORPO) [11] - jeszcze nowsza metoda, która idzie o krok dalej, integrując standardowe dostrajanie nadzorowane z uczeniem preferencyjnym w jednym etapie, co dodatkowo upraszcza proces treningowy.

1.3.4. Standardy ewaluacji w badaniach nad generacją tekstu

Ocena jakości wyjścia modeli generatywnych jest jednym ze standardowych wyzwań dla badaczy. W praktyce stosuje się metryki automatyczne, takie jak BLEU [12], ROUGE [13], czy odległość Levenshteina, które pozwalają na szybkie porównywanie modeli. W niniejszej pracy do automatycznej weryfikacji wyników zastosowano miarę odległości Levenshteina. Algorytm ten oblicza podobieństwo na poziomie pojedynczych tokenów, porównując liczbę modyfikacji wymaganych do przekształcenia jednego tekstu w drugi [14]. Oprócz zewnętrznych metryk, podczas samego procesu treningu preferencji stosuje się również szereg wewnętrznych wskaźników, które służą do wyboru najlepszej wersji modelu na podstawie tego, jak dobrze uczy się on rozróżniać preferowane odpowiedzi od odrzuconych.

Współczesne metody oceny jakości generowanych odpowiedzi przez duże modele językowe coraz częściej opierają się również na podejściu „model jako sędzia” (ang. model as a judge). Oznacza to, że zamiast korzystać wyłącznie z klasycznych metryk porównujących ciągi znaków, wykorzystuje się inne modele językowe, które na podstawie kontekstu i semantyki potrafią ocenić, na ile dana odpowiedź jest użyteczna. Do najważniejszych przedstawicieli tego podejścia należą:

- G-Eval – polega na zadawaniu dużemu modelowi językowemu specjalnie przygotowanych zapytań o ocenę wygenerowanej odpowiedzi w kontekście konkretnego zadania (np. trafność odpowiedzi na pytanie, styl wypowiedzi, poprawność gramatyczna) [15].
- GPT-Score – metryka wykorzystująca model GPT do obliczenia logarytmicznego prawdopodobieństwa danego tekstu, co pozwala ocenić naturalność języka [16].
- MoverScore – bardziej zaawansowana metryka, która porównuje teksty na poziomie osadzeń semantycznych (ang. embeddings), a nie pojedynczych słów czy znaków. Wykorzystuje dodatkową miarę word mover's distance (WMD), która oblicza minimalny koszt konwersji znaczenia jednego tekstu do drugiego w przestrzeni wektorowej. Dzięki temu MoverScore potrafi wychwycić podobieństwa semantyczne nawet wtedy, gdy teksty nie zawierają identycznych słów, ale są zbliżone pod względem znaczenia [17].

Podejścia te stanowią obecnie "state-of-the-art"w dziedzinie automatycznej ewaluacji odpowiedzi generowanych przez LLM, ponieważ są bardziej elastyczne i zbliżone do ludzkiej oceny niż tradycyjne metryki oparte wyłącznie na podobieństwie statystycznym.

Ich przewaga ujawnia się głównie w dłuższych tekstach, szczególnie w formacie konwersacyjnym.

1.3.5. Generacja nazw i uczenie na danych biznesowych

Zastosowanie dużych modeli językowych w zadaniach kreatywnych, takich jak tworzenie nazw domen internetowych, jest dynamicznie rozwijającą się dziedziną. Chociaż brakuje jeszcze recenzowanych publikacji poświęconych wyłącznie generowaniu nazw domen, prace nad tworzeniem nazw marek i produktów pokazują duży potencjał dużych modeli językowych w tej dziedzinie, wykazując jak ważne jest połączenie możliwości generacyjnych modelu z interaktywną oceną przez człowieka. Badacze oraz firmy eksplorują, w jaki sposób modele mogą generować nazwy, które są nie tylko chwytliwe, ale również oddają kluczowe cechy produktu i rezonują z docelową grupą odbiorców [18]. Niniejsza praca wpisuje się w ten trend, przenosząc te koncepcje na specyficzny temat nazw domen internetowych.

Jednocześnie kluczowym trendem w komercyjnych zastosowaniach LLM staje się ich dostrajanie w oparciu o twarde dane biznesowe. Firmy i badacze coraz częściej wykorzystują niejawne sygnały zwrotne (ang. implicit feedback) pochodzące bezpośrednio z interakcji użytkowników. Nowsze badania idą o krok dalej, dążąc do optymalizacji modeli bezpośrednio pod kątem metryk biznesowych.

Podejście to, polegające na optymalizacji modelu z myślą o celach biznesowych, jest postrzegane jako kolejny krok w ewolucji metod takich jak RLHF. Podejście przyjęte w niniejszej pracy, wykorzystujące dane o zarejestrowanych domenach jako historyczny wskaźnik sukcesu rynkowego, jest praktycznym przykładem zastosowania tej zaawansowanej koncepcji.

Warto zauważyć, że na rynku istnieją komercyjne narzędzia generujące nazwy domen i marek (głównie należące do firm rejestrujących domeny), które wykorzystują proste heurystyki i wyszukiwanie dostępności. Rozwiązania popularnych rejestratorów domen zestawiają słowa kluczowe z opisu a następnie filtrują wyniki pod kątem dostępności i długości [19][20]. Niniejsza praca zamiast heurystyk wykorzystuje proces uczenia na rzeczywistych preferencjach użytkowników i danych sprzedażowych, co pozwala bezpośrednio optymalizować zgodność z rynkiem.

1.4. Struktura pracy

Niniejsza praca została podzielona na pięć rozdziałów, które w sposób uporządkowany przedstawiają tło teoretyczne, metodykę badawczą oraz wyniki przeprowadzonych eksperymentów.

Rozdział pierwszy stanowi wprowadzenie do omawianej problematyki. Nakreślono w nim znaczenie i wyzwania związane z generowaniem nazw domen internetowych, przedstawiono motywację do podjęcia badań oraz sformułowano cel, zakres i strukturę pracy.

Rozdział drugi poświęcony jest fundamentom teoretycznym wykorzystanych metod. Omówiono w nim podstawy uczenia ze wzmocnieniem oraz szczegółowo opisano architekturę i zasady działania kluczowych technik optymalizacji modeli językowych, takich jak uczenie ze wzmocnieniem z informacją zwrotną (RLHF), bezpośrednia optymalizacja preferencji (DPO) oraz optymalizacja polityki metodami ORPO i RLAIF.

Rozdział trzeci szczegółowo opisuje metodykę przeprowadzonego procesu badawczego. Przedstawiono w nim definicję zadania, charakterystykę wykorzystanego zbioru danych oraz proces jego przygotowania. Ponadto, opisano architekturę zaimplementowanych modeli oraz przebieg procesu treningowego i ewaluacji.

Rozdział czwarty zawiera analizę wyników uzyskanych w ramach eksperymentów. Zamieszczono w nim wyniki ilościowe uzyskane za pomocą standardowych metryk oraz przeprowadzono analizę jakościową nazw domen wygenerowanych przez różne modele. Całość uzupełnia dyskusja, w której zinterpretowano uzyskane rezultaty.

Rozdział piąty stanowi podsumowanie całej pracy. Przedstawiono w nim syntezę wyników, sformułowano wnioski końcowe oraz wskazano na ograniczenia przeprowadzonego badania i możliwe kierunki dalszych prac.

2. Fundamenty teoretyczne optymalizacji modeli językowych

2.1. Architektury generatywnych modeli językowych

Współczesne modele językowe zdolne do generowania tekstu o wysokiej jakości opierają swoje działanie na zaawansowanych architekturach sieci neuronowych. Zrozumienie ich budowy jest kluczowe dla kontekstu niniejszej pracy. W tej sekcji omówiono podstawową współczesną architekturę Transformer oraz jej konkretne implementacje wykorzystane w badaniach.

2.1.1. Architektura Transformer

Przełomem w przetwarzaniu języka naturalnego było wprowadzenie architektury Transformer [1]. W odróżnieniu od wcześniejszych modeli opartych na przykład na sieciach rekurencyjnych (ang. Recurrent Neural Network, RNN) lub konwolucyjnych (ang. Convolutional Neural Network, CNN), Transformer w całości bazuje na mechanizmie atencji, a w szczególności na jego wariancie zwanym samoatencją (ang. self-attention).

Mechanizm ten pozwala na automatyczne ważenie istotności konkretnych tokenów w sekwencji wejściowej bezpośrednio w trakcie przetwarzania. Dzięki temu model jest w stanie efektywnie wychwytywać zawiłe zależności w tekście, co stanowiło znaczne ograniczenie starszych architektur [21].

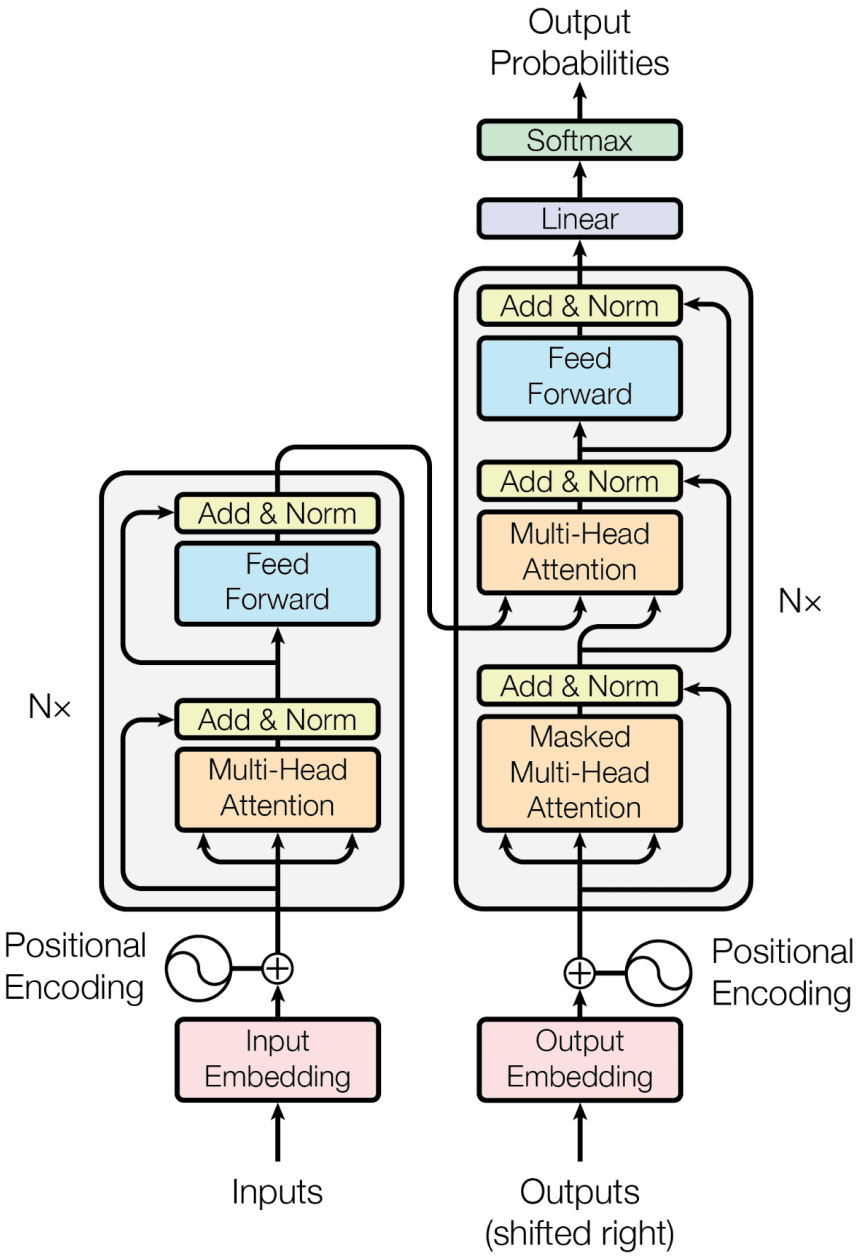
Standardowa architektura przedstawiona na rysunku 2.1 ilustruje dwuczęściowy podział na enkoder i dekoder oraz centralną rolę mechanizmu atencji w obu komponentach.

- Enkoder przetwarza całą sekwencję wejściową (np. opis strony internetowej) i tworzy jej numeryczną reprezentację.
- Dekoder na podstawie tej reprezentacji oraz dotychczas wygenerowanych tokenów kolejno generuje nową sekwencję wyjściową (np. nazwę domeny).

2.1.2. Model BART i jego polska adaptacja

BART (Bidirectional and Auto-Regressive Transformers) jest modelem opartym na architekturze Transformer typu enkoder-dekoder. W ramach jego treningu zastosowano standardową technikę polegającą na odszumowaniu (ang. denoising) uszkodzonego tekstu. W takim procesie model uczy się rekonstruować oryginalny tekst z jego zniekształconej wersji (na przykład z usuniętymi lub poprzestawianymi fragmentami). Taka strategia treningowa sprawia, że BART doskonale sprawdza się w zadaniach generacyjnych, które wymagają zarówno głębokiego zrozumienia tekstu wejściowego, jak i płynnego tworzenia nowych treści [2].

W niniejszej pracy jako model bazowy do generowania nazw domen wykorzystano zmodyfikowaną wersję modelu BART, która została wytrenowana od podstaw na dużym, zróżnicowanym korpusie języka polskiego [4]. Dzięki temu model posiada zdolność do



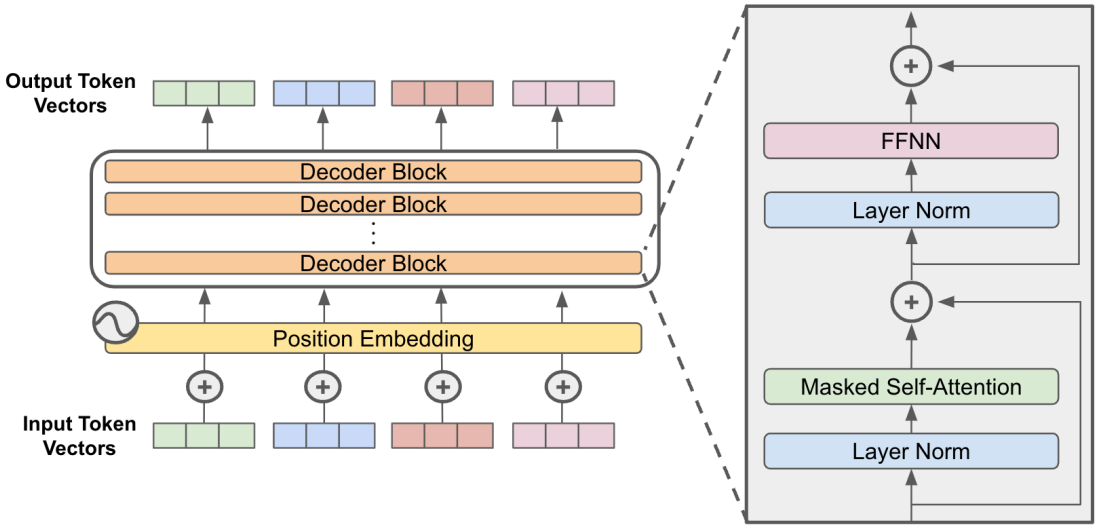
Rysunek 2.1. Architektura Transformer [1]

przetwarzania i generowania tekstu z uwzględnieniem specyfiki języka polskiego, co jest kluczowe dla uzyskania wysokiej jakości wyników w badanym zadaniu.

2.1.3. Modele oparte na architekturze decoder-only

Oprócz modeli typu enkoder-dekoder, duże uznanie zyskała architektura oparta wyłącznie na dekoderze (ang. decoder-only) przedstawiona na rysunku 2.2. W modelach tego typu nie ma oddzielnego enkodera, a zarówno tekst wejściowy, jak i wyjściowy są przetwarzane przez pojedynczy blok dekodera [22]. Takie podejście okazało się niezwykle

skuteczne, szczególnie w przypadku modeli trenowanych na ogromnych zbiorach danych [23]. Do najpopularniejszych modeli tego typu należą:



Rysunek 2.2. Architektura Transformer typu decoder-only [24]

- Mistral-7B - przykład anglojęzycznego, otwartego modelu o relatywnie niewielkiej liczbie parametrów (7 miliardów), który osiąga wydajność porównywalną ze znacznie większymi modelami. Jest to możliwe dzięki nowym rozwiązaniom, które przyspieszają obliczenia i zmniejszają zużycie pamięci, takim jak Grouped-Query Attention (GQA) czy Sliding-Window Attention (SWA) [3].
- Bielik-1.5B - model z rodziny polskojęzycznych modeli Bielik, wytrenowany przez polskich badaczy. W niniejszej pracy został on wykorzystany w roli modelu nagrody do oceny jakości generowanych nazw domen. Jego zastosowanie pozwala na bardziej wiarygodną i kontekstową ocenę propozycji niż w przypadku metryk opartych wyłącznie na statystycznym podobieństwie tokenów [5].
- PLLuM (Polish Large Language Model) [25] - przedstawiciel polskojęzycznych modeli dialogowych udostępniony na licencji otwartej. W odróżnieniu od Bielika, który występuje w konfiguracjach od 1,5 miliarda do 7 miliardów parametrów, modele PLLuM są zazwyczaj potężniejsze i trenowane na korpusie o większej objętości. Dodatkowo twórcy wspierają innowacje w sektorach publicznym i prywatnym poprzez rozwój narzędzi, takich jak inteligentny asystent petenta [26].

2.2. Uczenie ze wzmocnieniem

Uczenie ze wzmocnieniem (ang. Reinforcement Learning, RL) odwzorowuje ludzki proces nauki poprzez metodę prób i błędów, dążąc do wypracowania optymalnego zachowania w danym środowisku. Algorytmy uczenia ze wzmocnieniem opierają się na paradygmacie nagrody i kary, aby poprzez analizę informacji zwrotnej stopniowo udoskonalać sposób podejmowania decyzji. W kontekście uczenia ze wzmocnieniem autonomiczny

system decyzyjny nazywany jest agentem, a strategia postępowania agenta - jego polityką [6].

Metoda ta opiera się na procesie decyzyjnym Markowa (ang. Markov Decision Process, MDP), który jest matematycznym modelem podejmowania decyzji w systemach zmieniających się w czasie. Środowisko jest reprezentowane przez zbiór stanów, a agent podejmuje akcje powodujące przejście do nowych stanów. System nagród i kar dostarcza agentowi informacji zwrotnej, oceniając skuteczność podjętych działań. Każda decyzja jest rozpatrywana w odniesieniu do możliwego stanu, który może nastąpić po jej podjęciu. Nagroda agenta r definiowana jest jako funkcja zależna od określonej akcji a, aktualnego stanu środowiska s oraz potencjalnego kolejnego stanu s' [27]:

$$r_t(s,a) = \sum_{s' \in S} r_t(s,a,s')\, p_t(s' \mid s,a) \tag{1}$$

Tak zdefiniowana funkcja nagrody może zostać wykorzystana jako element polityki sterującej zachowaniem agenta. Wyznaczenie optymalnej polityki uwzględniającej kontekst opóźnionej gratyfikacji stanowi jedno z głównych wyzwań metod programowania dynamicznego w uczeniu ze wzmocnieniem. Wiele algorytmów celowo wprowadza rozwiązania generujące krótkoterminowe straty, aby w dłuższej perspektywie zmaksymalizować łączną nagrodę. Ta właściwość pozwala agentom na wypracowywanie strategii uwzględniających kompromis między natychmiastowymi konsekwencjami a długofalowymi celami [28]. Podstawowym narzędziem w tym kontekście jest równanie Bellmana, które definiuje optymalną funkcję wartości $v_t(s)$ jako maksymalną możliwą oczekiwaną nagrodę, którą agent może uzyskać począwszy od chwili t, będąc w stanie s. Wartość ta składa się z natychmiastowej nagrody $r_t(s,a)$ wynikającej z podjęcia akcji a w stanie s oraz z oczekiwanej sumy nagród w przyszłości, ważonej prawdopodobieństwem przejścia do kolejnych stanów $s' \in S$ [27]:

$$v_t(s) = \max_{a \in A(s)} \left\{ r_t(s,a) + \sum_{s' \in S} p_t(s' \mid s,a)\, v_{t+1}(s') \right\} \tag{2}$$

Najczęściej poruszanym wyzwaniem metody uczenia ze wzmocnieniem jest dylemat eksploracji i eksploatacji. Agent musi dynamicznie balansować między badaniem nieznanych obszarów przestrzeni stanów a wykorzystywaniem dotychczasowej wiedzy, aby skutecznie maksymalizować łączną nagrodę. Nadmierna eksploracja może prowadzić do krótkoterminowych strat, podczas gdy zbyt duża eksploatacja może uniemożliwić odkrycie lepszych strategii. W praktyce stosuje się polityki ϵ-zachłanne (ang. ϵ-greedy), miękką maksymalizację czy eksplorację przez entropię (jak w PPO), które pozwalają kontrolować siłę eksploracji [29].

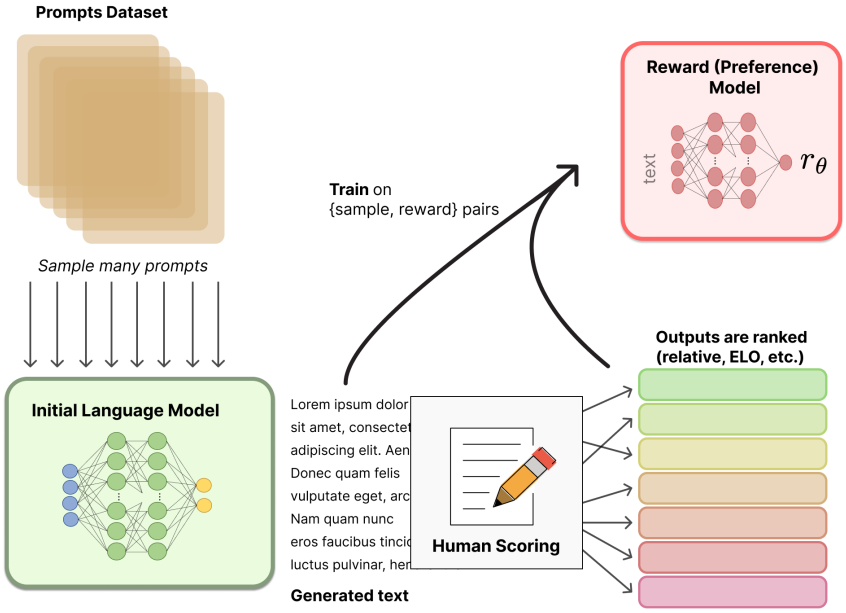
Ponadto skuteczność danej metody jest ściśle skorelowana z jakością zdefiniowanej funkcji nagrody. Problem ten pojawia się zazwyczaj w skomplikowanych zadaniach, w których utrudniona jest jednoznaczna definicja sukcesu oraz łatwiej jest ocenić wynik niż zaprojektować odpowiednią strategię wymaganą do jego osiągnięcia. W rezultacie agent

może zoptymalizować swoje działania w sposób, który maksymalizuje funkcję nagrody, ale niekoniecznie odpowiada oczekiwanym celom lub wartościom użytkownika [7].

2.3. Uczenie ze wzmocnieniem z informacją zwrotną

Uczenie ze wzmocnieniem z informacją zwrotną (RLHF) to zaawansowane podejście do doskonalenia modeli sztucznej inteligencji, które uwzględnia ludzkie preferencje w procesie treningowym. Głównym celem tej metody jest poprawa jakości generowanych odpowiedzi, zapewnienie ich zgodności z oczekiwaniami użytkownika oraz zapobieganie potencjalnie szkodliwym lub nieetycznym wynikom [8].

W pierwszym kroku procesu treningowego RLHF wykorzystuje się model uprzednio wytrenowany na dużych zbiorach danych, dostosowując go do realizacji konkretnego zadania przy pomocy metody uczenia nadzorowanego (patrz rysunek 2.3). Etap ten pozwala na stworzenie podstawowej polityki generowania odpowiedzi.



Rysunek 2.3. Proces treningu RLHF - krok SFT oraz budowa modelu nagrody [30]

Następnie buduje się model nagrody (ang. Reward Model, RM), który jest kluczowym elementem RLHF. Jego zadanie polega na przekształceniu ludzkich preferencji na numeryczny sygnał oceniający jakość odpowiedzi. Przyjmuje on na wejściu tekst wygenerowany przez główny model i przypisuje mu wartość liczbową odwzorowującą preferencje człowieka. Główną funkcją modelu nagrody jest ograniczenie potrzeby ciągłego zaangażowania człowieka w proces oceny, co pozwala na jego trening na relatywnie niewielkim, ręcznie oznaczonym zbiorze danych, a następnie wykorzystanie go do automatycznej oceny znacznie większej liczby przykładów [30].

Za pomocą technik, takich jak Proximal Policy Optimization (PPO), polityka modelu jest dostosowywana w celu maksymalizacji nagrody. PPO wprowadza również ograniczenia w zakresie rozmiaru pojedynczej zmiany polityki przy użyciu powszechnie stosowanej dywergencji Kullbacka-Leiblera [9].

2.4. Alternatywne metody uczenia preferencyjnego

W ramach badań nad ulepszaniem modeli sztucznej inteligencji rozwijane są modyfikacje metody uczenia ze wzmocnieniem z informacją zwrotną. Takie podejścia oferują alternatywne sposoby optymalizacji modeli w celu utrzymania pożądanych rezultatów przy jednoczesnym zmniejszeniu złożoności, nakładu czasowego oraz dodatkowych kosztów.

2.4.1. Metoda Direct Preference Optimization

Metoda Direct Preference Optimization (DPO) stanowi alternatywę dla RLHF, umożliwiając bardziej bezpośrednie modelowanie ludzkich preferencji. Proces rozpoczyna się od dostosowania wstępnie wytrenowanego modelu do konkretnego zadania za pomocą technik klasycznego uczenia nadzorowanego.

Model uzyskany w pierwszym kroku jest następnie bezpośrednio optymalizowany zgodnie z ludzkimi preferencjami. Problem maksymalizacji nagrody jest sformułowany w postaci standardowej funkcji straty, co zapewnia większą efektywność i stabilność algorytmu w porównaniu z poprzednio omawianym podejściem.

Cały proces treningu przy użyciu metody DPO jest stosunkowo zbliżony do metody RLHF z wykorzystaniem PPO, ale redukuje złożoność i podatność na błędy poprzez eliminację oddzielnego modelu nagrody i uproszczenie funkcji optymalizacyjnej dla polityki modelu [10].

2.4.2. Metoda Offline Reinforcement Learning Policy Optimization

Offline Reinforcement Learning Policy Optimization (ORPO) to metoda uczenia preferencyjnego modeli językowych bez konieczności korzystania z modelu referencyjnego. Jej założenie opiera się na prostszym podejściu do optymalizacji polityki, eliminując potrzebę wieloetapowego treningu. Dzięki temu skrótowi ORPO umożliwia dostosowanie modelu w jednym kroku, co zmniejsza złożoność procesu treningowego, jednocześnie zapewniając skuteczność w dopasowywaniu polityki do ludzkich preferencji [11].

Analogicznie do metody DPO, wstępnie wytrenowany model jest dostosowywany do konkretnego zadania przy pomocy metod uczenia nadzorowanego. Natomiast już w ramach tego kroku następuje optymalizacja polityki modelu – ORPO silnie wzmacnia prawdopodobieństwo generowania odpowiedzi zaakceptowanych przez człowieka, a jednocześnie łagodnie obniża prawdopodobieństwo odpowiedzi odrzuconych. Optymalizacja opiera się na funkcji straty, która uwzględnia zarówno czynnik wynikający z preferencji użytkownika, jak i funkcję ujemnej logwiarygodności (ang. negative log-likelihood).

2.4.3. Metoda Reinforcement Learning from AI Feedback

Metoda Reinforcement Learning from AI Feedback (RLAIF) to nowatorska technika, która zastępuje tradycyjny komponent ludzkiej informacji zwrotnej ocenami generowanymi przez inne modele sztucznej inteligencji.

Takie podejście całkowicie eliminuje zapotrzebowanie na czasochłonną i kosztowną ludzką informację zwrotną, jednocześnie zachowując spójność sygnału wyjściowego używanego do uczenia preferencyjnego. Ponadto jest mniej podatna na subiektywne błędy w porównaniu z ludzkim sposobem rankingowania.

RLAIF jest szczególnie atrakcyjną alternatywą w sytuacjach, w których bezpośrednie zaangażowanie człowieka w proces oceniania jest utrudnione lub kosztowne, a jednocześnie istnieje potrzeba dopasowania wyników modelu do specyficznych kryteriów. Ograniczony udział człowieka może utrudniać wychwytywanie niuansów preferencji oraz wartości etycznych, co w niektórych zastosowaniach może stanowić istotne wyzwanie, lecz nie wyklucza użyteczności tej metody. Warto podkreślić, że hybrydowe modele łączące RLAIF z okazjonalną walidacją ekspertów są obecnie przedmiotem intensywnych badań, mających na celu pogodzenie efektywności z zachowaniem kontroli i uwzględnieniem ludzkich wartości [31].

2.5. Miary jakości w ewaluacji modeli generatywnych

Ocena jakości tekstu generowanego przez modele językowe jest jednym z największych wyzwań w dziedzinie przetwarzania języka naturalnego. W przeciwieństwie do zadań klasyfikacyjnych, gdzie poprawność odpowiedzi jest jednoznaczna, w przypadku generacji tekstu kryteria takie jak płynność, kreatywność czy adekwatność są często subiektywne. Z tego powodu stosuje się zróżnicowane podejścia do ewaluacji, obejmujące zarówno metryki automatyczne, jak i kosztowną ocenę ludzką [32].

2.5.1. Metryki automatyczne oparte na podobieństwie

W celu zapewnienia powtarzalnej i obiektywnej oceny powszechnie wykorzystuje się metryki statystyczne, które porównują tekst wygenerowany przez model z jednym lub kilkoma tekstami referencyjnymi (ang. ground truth). Do najpopularniejszych rodzin takich metryk należą odległość Levenshteina (przydatna do porównywania krótkich tekstów niebędących pełnymi zdaniami), BLEU (stosowane głównie w tłumaczeniu maszynowym) [12] oraz ROUGE (Recall-Oriented Understudy for Gisting Evaluation) [13], która jest standardem w zadaniach streszczania i generacji tekstu.

W niniejszej pracy do automatycznej weryfikacji wyników zastosowano miarę odległości Levenshteina, bazującą na minimalnej liczbie operacji edycyjnych, które są potrzebne do przekształcenia jednego ciągu w drugi. Operacje te obejmują:

- Wstawienie znaku (ang. insertion) – dodanie pojedynczego znaku w dowolnym miejscu ciągu.

- Usunięcie znaku (ang. deletion) – usunięcie pojedynczego znaku z ciągu.
- Zamiana znaku (ang. substitution) – zamiana pojedynczego znaku na inny.

Każda z tych operacji ma zazwyczaj taki sam koszt (np. 1 jednostka), choć w rozszerzonych wersjach algorytmu można stosować różne wagi. Standardowy sposób obliczania tej odległości wykorzystuje paradygmat programowania dynamicznego. Tworzona jest macierz, w której każda komórka reprezentuje minimalny koszt przekształcenia fragmentu jednego ciągu w drugi. Algorytm ten działa w czasie $O(n \times m)$, gdzie n i m to długości porównywanych ciągów [33].

Wybór odległości Levenshteina jako główną automatyczną miarę jakości na potrzeby niniejszego zadania uzasadnia szereg czynników:

- Działanie na poziomie znaków, które wpływa na zwiększoną efektywność dla bardzo krótkich ciągów, takich jak nazwy domen.
- Niezależność od tokenizacji i języka, umożliwiająca przetestowanie różnych modeli bazowych.
- Możliwość normalizacji wyników do zakresu [0, 1], pozwalająca na łatwe porównanie wyników.

Alternatywy takie jak BLEU/ROUGE są mniej stabilne dla pojedynczych, krótkich nazw [34], a metryki oparte na osadzeniach trudniejsze do kalibracji dla krótkich ciągów. W konsekwencji miara odległości Levenshteina wyjątkowo skutecznie odzwierciedla jakość pojedynczej wygenerowanej nazwy domeny [35].

Należy jednak podkreślić, że wszystkie metryki oparte na podobieństwie n-gramów mają swoje ograniczenia. Nie są w stanie ocenić poprawności gramatycznej, sensu merytorycznego ani kreatywności wygenerowanego tekstu. W specyficznym zadaniu generowania nazw domen ich użyteczność jest dodatkowo ograniczona przez krótką długość tekstu, co sprawia, że nawet niewielkie różnice mogą drastycznie wpłynąć na wynik [36].

2.5.2. Wewnętrzne metryki treningowe

Oprócz ewaluacji wyników po zakończonym treningu, kluczową rolę odgrywają metryki wykorzystywane w trakcie samego procesu do monitorowania postępów i wczesnego zakończenia w razie wykrycia braku poprawy. W przypadku metod uczenia preferencyjnego, takich jak DPO, metryki te nie oceniają podobieństwa do tekstu referencyjnego, lecz zdolność modelu do rozróżniania między wybranymi a odrzuconymi odpowiedziami [37].

W tej pracy, podczas treningu na danych preferencyjnych, jako kluczowy wskaźnik wyboru najlepszego modelu wykorzystano metrykę reprezentującą średnią różnicę w logarytmicznych prawdopodobieństwach przypisanych przez model do odpowiedzi wybranych i odrzuconych. Wyższa wartość tej miary jakości oznacza zwiększoną zdolność modelu do promowania pożądanych przez użytkownika odpowiedzi.

2.6. Proces rejestracji domen internetowych

Rejestr domen internetowych pełni kluczową rolę w procesie rejestracji nazw domen, utrzymując centralny rejestr nazw oraz infrastrukturę niezbędną do ich obsługi. Rejestracja odbywa się w modelu partnerskim - akredytowani rejestratorzy pośredniczą w procesie, umożliwiając użytkownikom wyszukiwanie, zgłoszenie i utrzymanie domen. Typowy przebieg obejmuje weryfikację dostępności nazwy, złożenie wniosku przez rejestratora oraz wprowadzenie danych do rejestru, co skutkuje formalnym przypisaniem domeny do użytkownika. Analiza danych rejestracyjnych wskazuje na wyraźną sezonowość oraz obecność trendów kształtowanych zarówno przez czynniki zewnętrzne (m.in. uwarunkowania społeczno-ekonomiczne), jak i wewnętrzne (polityki cenowe i operacyjne rejestratorów). Czynniki te wpływają na preferencje nabywców co do cech wybieranych nazw, takich jak długość, obecność słów kluczowych czy użycie łączników, a zatem również na dynamikę całego rynku rejestracji domen [38].

3. Metodyka i środowisko badawcze

3.1. Definicja zadania badawczego

Generacja nazw domen internetowych to zadanie polegające na automatycznym tworzeniu krótkich, unikalnych i adekwatnych adresów internetowych na podstawie opisu działalności realizowanej przez stronę internetową. W praktyce oznacza to, że model językowy otrzymuje informacje dotyczące funkcji oraz tematyki strony i na tej podstawie rekomenduje nazwę domeny, która powinna cechować się atrakcyjnością, łatwością zapamiętania oraz tematycznym powiązaniem z przedstawionym opisem.

Zadanie to wykracza poza klasyczną generację tekstu i podobnie jak tłumaczenie czy streszczanie wymaga od modelu jednoczesnego spełnienia kilku, często przeciwstawnych, kryteriów.

Przedmiotem niniejszej pracy dyplomowej jest weryfikacja skuteczności nowoczesnych metod uczenia preferencyjnego, w której jawna informacja zwrotna od człowieka (ang. human feedback) zostaje uzupełniona przez sygnały pochodzące z danych sprzedażowych. Głównym celem jest zbadanie, jak te zaawansowane techniki radzą sobie z optymalizacją modelu językowego w kreatywnym zadaniu generowania nazw domen internetowych.

W celu realizacji postawionego celu, przeprowadzona zostanie analiza porównawcza trzech wiodących metod optymalizacji preferencyjnej:

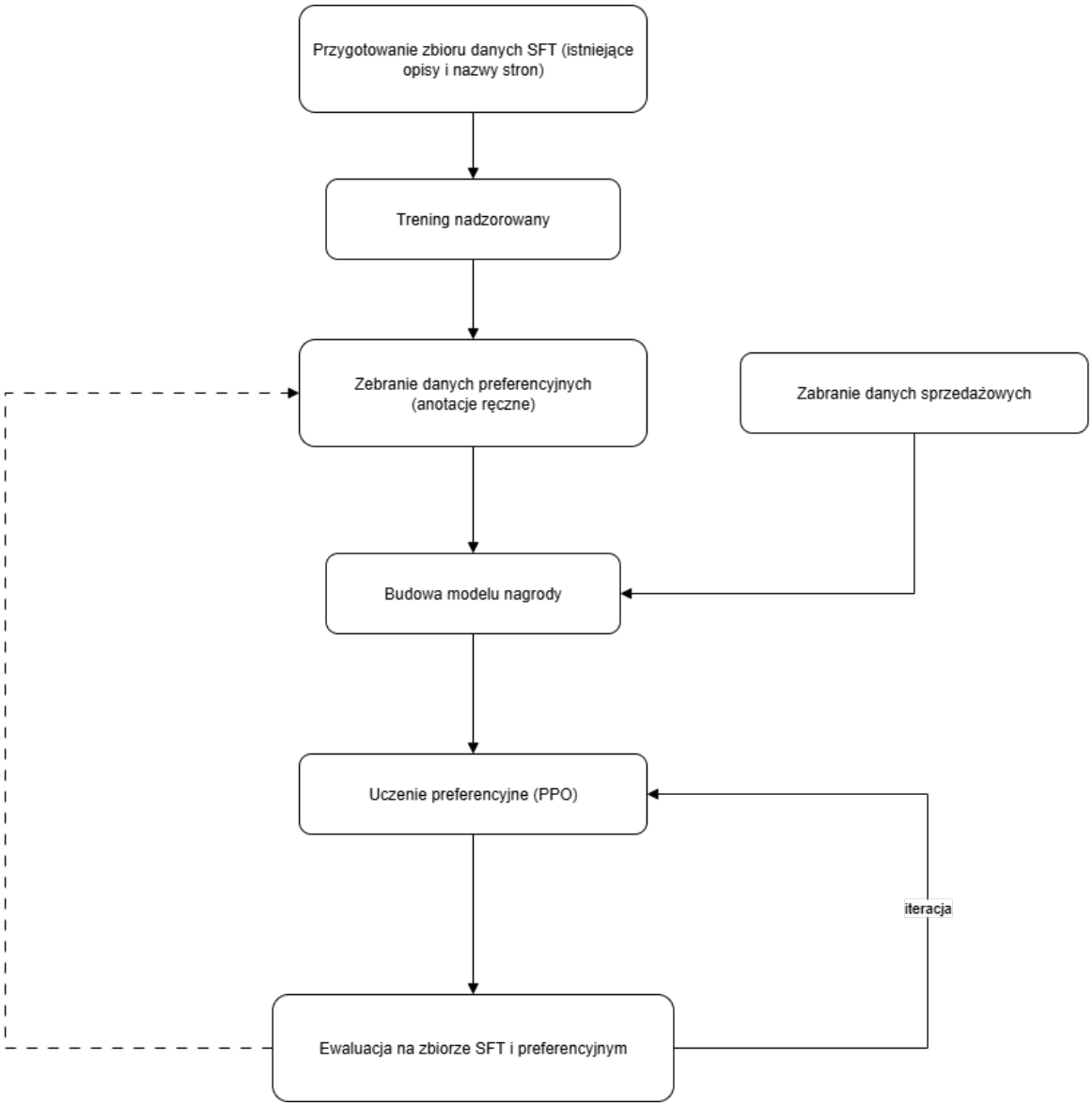
- RLHF - klasyczne, wieloetapowe podejście, stanowiące punkt odniesienia.
- DPO - uproszczona metoda eliminująca potrzebę osobnego modelu nagrody.
- ORPO - jednoetapowe podejście integrujące proces dostrajania i optymalizacji.

Badanie zostało przeprowadzone z wykorzystaniem zróżnicowanych zbiorów danych preferencyjnych, co pozwoliło na zbadanie wpływu dwóch kluczowych aspektów na jakość modelu:

- Źródło danych - porównano skuteczność treningu na zbiorze opartym wyłącznie na ręcznych adnotacjach ze zbiorem zawierającym również dane sprzedażowe.
- Liczebność zbioru danych - przeanalizowano, jak wydajność poszczególnych metod skaluje się w zależności od wielkości dostępnych danych treningowych.

Uproszczony schemat blokowy procedury badawczej dla metody RLHF został przedstawiony na rysunku 3.1. Ukazuje on kolejność i relacje poszczególnych etapów eksperymentu wraz z opcjonalnymi iteracjami. Dla metody DPO schemat prezentowałby się podobnie, byłby jedynie pozbawiony fazy budowy modelu nagrody. Z kolei przy treningu ORPO pominięty jest również etap treningu nadzorowanego.

Celem tak zaprojektowanego eksperymentu jest nie tylko wyłonienie najskuteczniejszej i najbardziej efektywnej metody dla badanego zadania, ale również sformułowanie praktycznych wniosków dotyczących wrażliwości poszczególnych technik na charakterystykę i objętość danych treningowych.



Rysunek 3.1. Uproszczony schemat procedury badawczej dla metody RLHF

3.2. Zbiór danych i jego przygotowanie

W ramach przyjętej metody badawczej wykorzystano rzeczywisty zbiór danych obejmujący istniejące nazwy domen internetowych zarejestrowanych w polskiej przestrzeni adresowej. Każdy przykład w zbiorze zawiera parę: opis strony internetowej oraz aktualnie wykupioną domenę, co pozwala na ocenę trafności propozycji generowanych przez model w odniesieniu do faktycznych wyborów właścicieli stron.

W niniejszej pracy zastosowano hybrydową strategię danych, opierającą się na dwóch fundamentalnie różnych typach zbiorów, które pełniły odmienne role w procesie treningowym modeli. Wszystkie wykorzystane dane pochodzą z polskiej przestrzeni internetowej, co zapewnia ich językową i kulturową adekwatność do badanego zadania.

Przed użyciem w eksperymentach dane poddano podstawowemu przetwarzaniu, które

obejmowało konwersję nazw domen do małych liter oraz standaryzację formatu w celu usunięcia ewentualnych anomalii.

Na szczególną uwagę zasługuje duża różnica w średniej długości opisu między zbiorem SFT a zbiorem danych sprzedażowych (patrz tabela 3.1). Skoro wejścia w środowisku produkcyjnym są znacznie krótsze, warto rozważyć dostosowanie zbioru SFT do podobnych długości opisów. Tym bardziej wartościowy staje się rynkowy sygnał z danych preferencyjnych, który ukierunkowuje model na rzeczywiście wybierane przez użytkowników nazwy.

	Zbiór SFT	Ręczne anotacje	Zbiór danych sprzedażowych
Liczba przykładów	43844	3236	405
Śr. długość opisu	166,99	183,33	25,34
Śr. długość domeny	15,68	13,82	14,29
Źródło danych	NASK	Anotatorzy z NASK i Autor	NASK

Tabela 3.1. Charakterystyka zbiorów danych

3.2.1. Dane do dostrajania nadzorowanego

Pierwszym filarem procesu treningowego jest obszerny zbiór danych do dostrajania nadzorowanego (ang. Supervised Fine-Tuning, SFT). Zbiór ten składał się z ponad 40000 unikalnych wierszy zawierających opis strony i zarejestrowaną nazwę domeny (szczegółowy rozkład zbioru przedstawia tabela 3.2), a jego głównym zastosowaniem jest wstępna adaptacja modelu językowego do specyfiki zadania generowania nazw domen.

- Format - zbiór par (opis strony internetowej, zarejestrowana nazwa domeny).
- Rola w procesie:
  — Uczenie nadzorowane modelu BART w celu dostosowania go do zadania generacji nazw domen, co stanowi fundament dla późniejszych eksperymentów oraz procesu tworzenia zbioru danych preferencyjnych.
  — Aktualizacja polityki modelu (PPO) w podejściu RLHF przy pomocy oddzielnie wytrenowanego modelu nagrody.

	Nazwa domeny	Opis strony
Maksymalna długość (liczba znaków)	64	1364
Minimalna długość (liczba znaków)	6	28
Średnia długość (liczba znaków)	15,68	166,99
Odchylenie standardowe (liczba znaków)	5,05	69,67

Tabela 3.2. Szczegółowa charakterystyka zbioru danych SFT

3.2.2. Dane preferencyjne

Drugim, kluczowym elementem są zbiory danych preferencyjnych. Charakteryzują się one znacznie mniejszą licznością w postaci łącznie 3641 przykładów (patrz tabele 3.1, 3.3 oraz 3.4), ale niosą ze sobą precyzyjny sygnał dotyczący preferencji ludzkich i rynkowych. Dane te są niezbędne w zaawansowanych fazach optymalizacji.

- Format - zbiór trójek (zapytanie, odpowiedź wybrana, odpowiedź odrzucona), gdzie zapytanie to opis strony uzupełniony o odpowiednio spreparowany prompt, a odpowiedzi to dwie różne propozycje nazw domen.
- Rola w procesie:
  — Trening modelu nagrody w klasycznym podejściu RLHF.
  — Druga faza treningu w metodzie DPO, mająca na celu optymalizację polityki wstępnie wytrenowanego modelu.
  — Całościowy trening w podejściu ORPO.

	Wybrana nazwa	Odrzucona nazwa	Opis strony
Maksymalna długość	32,00	33,00	329,00
Minimalna długość	3,00	1,00	45,00
Średnia długość	14,15	13,48	183,33
Odchylenie standardowe	4,26	4,55	69,28

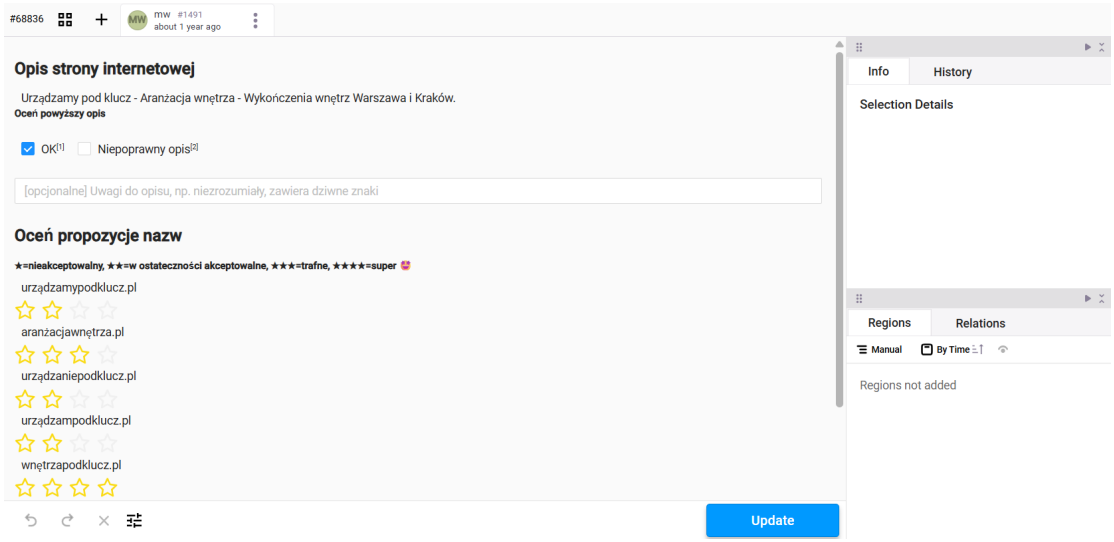
Tabela 3.3. Szczegółowa charakterystyka zbioru preferencyjnego ręcznych anotacji

	Wybrana nazwa	Odrzucona nazwa	Opis strony
Maksymalna długość	24,00	35,00	116,00
Minimalna długość	6,00	5,00	3,00
Średnia długość	13,82	14,75	25,34
Odchylenie standardowe	4,43	4,43	24,07

Tabela 3.4. Szczegółowa charakterystyka zbioru preferencyjnego danych sprzedażowych

Źródłem danych preferencyjnych jest połączenie ręcznych anotacji z danymi sprzedażowymi. Proces tworzenia zbioru był wieloetapowy - model po fazie SFT generował zestaw propozycji nazw domen dla danych opisów, które następnie podlegały ocenie grupy anotatorów przy użyciu programu Label Studio (przykładowa anotacja została przedstawiona na rysunku 3.2). Oceniali oni poszczególne propozycje w skali (1-10) pod kątem kreatywności, trafności, potencjału marketingowego oraz zgodności z zasadami języka. Równolegle, twarde dane sprzedażowe dostarczały binarnej informacji o tym, która z propozycji została ostatecznie zakupiona przez klienta, a które odrzucono.

Ostateczny zbiór danych preferencyjnych został skonstruowany poprzez konwersję obu tych sygnałów do wspólnego formatu par preferencji. Zrezygnowano ze szczegółowej oceny liczbowej (1-10) niedostępnej w danych sprzedażowych na rzecz ujednoliconego binarnego sygnału, co pozwoliło na efektywną integracji obu części zbioru.



Rysunek 3.2. Przykładowa anotacja przy użyciu programu Label Studio [39]

3.3. Architektura i implementacja modeli

W celu przeprowadzenia procesu badawczego wykorzystano złożoną architekturę, składającą się z kilku wyspecjalizowanych modeli językowych, z których każdy pełnił odmienną, kluczową rolę w eksperymencie. Poniżej przedstawiono charakterystykę i zastosowanie poszczególnych komponentów.

3.3.1. Model generujący: Polish BART

Bazowym modelem generatywnym jest polskojęzyczna adaptacja architektury BART (Bidirectional and Auto-Regressive Transformers) [40], która dzięki swojej budowie typu enkoder-dekoder jest naturalnie przystosowana do zadań typu text-to-text. Wybór tego konkretnego modelu był podyktowany jego udowodnioną skutecznością w zadaniu streszczania tekstu, które jest stosunkowo podobne do zadania generacji domen internetowych, gdzie długi opis jest kondensowany w krótką i chwytliwą nazwę. W ramach eksperymentu model ten przechodził dwuetapowy proces treningu:

- Dostrajanie nadzorowane (SFT) - w ramach tego etapu model uczył się standardowej zależności między opisem a nazwą domeny na dużym zbiorze danych.
- Optymalizacja preferencyjna - wytrenowany model SFT stanowił następnie punkt wyjścia do dalszej optymalizacji za pomocą metod RLHF i DPO.

3.3.2. Model nagrody: Bielik-1.5B

W metodologii RLHF wykorzystano model Bielik-1.5B. Jest to polskojęzyczny model o architekturze "decoder-only", który dzięki treningowi na ogromnym korpusie języka polskiego posiada zdolność do rozumienia subtelnych niuansów lingwistycznych.

Jego rola w procesie badawczym skupiała się na pełnieniu funkcji modelu nagrody (ang. Reward Model). Jego zadaniem było przypisanie numerycznej oceny (nagrody) do

par (opis, wygenerowana nazwa), ucząc się w ten sposób, które propozycje są bardziej preferowane na podstawie danych treningowych.

3.3.3. Model referencyjny

Kluczowym komponentem w technikach optymalizacji jest również model referencyjny. W niniejszym projekcie rolę tę pełniła niezmieniona kopia modelu Polish BART po zakończeniu etapu SFT.

Tak spreparowany model nie jest trenowany podczas fazy optymalizacji preferencyjnej. Służy on wyłącznie jako zamrożony punkt odniesienia, a jego celem jest stabilizacja procesu treningu. Algorytmy PPO, DPO oraz ORPO dążą do maksymalizacji nagrody, jednocześnie dbając jednak o to, by polityka optymalizowanego modelu nie oddaliła się zbytnio od polityki modelu referencyjnego. Zapobiega to zjawisku katastrofalnego zapominania i generowaniu odpowiedzi, które sztucznie zawyżają metryki jakości (na przykład są niezrozumiałe lub odbiegają od naturalnego rozkładu języka).

3.4. Proces treningu i ewaluacji

W ramach procesu badawczego przeanalizowano trzy strategie treningowe modelu językowego, zróżnicowane pod względem mechanizmów adaptacji do specyficznych wymagań zadania generowania nazw domen. Zachowanie niezmienionej architektury bazowej we wszystkich przypadkach umożliwiło systematyczną ocenę efektywności poszczególnych metod, zapewniając rzetelność i porównywalność uzyskanych rezultatów.

Trening modeli językowych został przeprowadzony w środowisku Google Colab, co umożliwiło wykorzystanie chmurowych zasobów obliczeniowych w postaci karty graficznej Nvidia A100 oraz łatwą integrację z bibliotekami ekosystemu Hugging Face. Eksperymenty realizowano w oddzielnych notatnikach, dedykowanych kolejno trzem podejściom do uczenia ze wzmocnieniem z wykorzystaniem preferencji: RLHF, DPO oraz ORPO.

3.4.1. Przygotowanie środowiska treningowego

Podstawą implementacji były biblioteki *transformers* oraz *trl* (Transformers Reinforcement Learning), które zapewniają wysokopoziomowe API do trenowania dużych modeli językowych przy użyciu metody RLHF i jej popularnych alternatyw. Wykorzystano model bazowy BART jako punkt wyjścia, a następnie poddano go dalszemu dostrajaniu zgodnie z aktualnie trenowanym algorytmem.

3.4.2. Proces ewaluacji modeli

Ewaluacja trenowanych modeli przebiegała dwutorowo:

- Ewaluacja automatyczna - podczas treningu każdego podejścia użyta została odpowiednia automatyczna metryka pozwalająca na wstępną ocenę jakości oraz kontrolę przebiegu treningu już w jego trakcie.

- Ewaluacja porównawcza - generacje modeli porównywano między sobą w odrębnym notatniku Google Colab pod względem spójności, poprawności językowej oraz zgodności z intencją zadania. W tym celu analizowano pary odpowiedzi w kontekście danych preferencyjnych, co pozwoliło na ocenę, który algorytm skuteczniej uczy model pożądanych zachowań. Ta ewaluacja została dokonana przy pomocy miary podobieństwa bazującej na odległości Levenshteina oraz empirycznej, ręcznej analizie jakościowej generowanego tekstu.

4. Prezentacja i analiza wyników

4.1. Wyniki ilościowe

Przeprowadzono serię eksperymentów dla pięciu liczności zbioru opisów stron: $n = 100$, $n = 200$, $n = 400$, $n = 590$ (pełny zbiór ręcznych anotacji) oraz $n = 657$ (pełny zbiór po włączeniu danych sprzedażowych). Zastosowano także augmentację danych, więc końcowa liczność zbioru treningowego jest odpowiednio większa, ponieważ dla każdego opisu strony wygenerowano wiele par preferencji. Wszystkie modele, w tym oryginalny bazowy model, ewaluowano zarówno na zbiorze SFT (sprawdzając utrzymanie jakości zadania), jak i na zbiorze preferencyjnym (weryfikując zgodność z preferencjami użytkowników). Dane sprzedażowe włączono naiwnie, z tą samą wagą co pozostałe dane preferencyjne.

4.1.1. Wyniki dla podejścia RLHF

Wyniki dla metody RLHF przedstawiono na rysunku 4.1, gdzie oś X reprezentuje jakość na zbiorze preferencyjnym w postaci średniej różnicy znormalizowanej odległości Levenshteina między odpowiedziami preferowanymi a odrzucanymi. Z kolei oś Y oznacza jakość na zbiorze SFT wyrażoną poprzez średnie znormalizowane podobieństwo wyliczone również przy użyciu odległości Levenshteina.

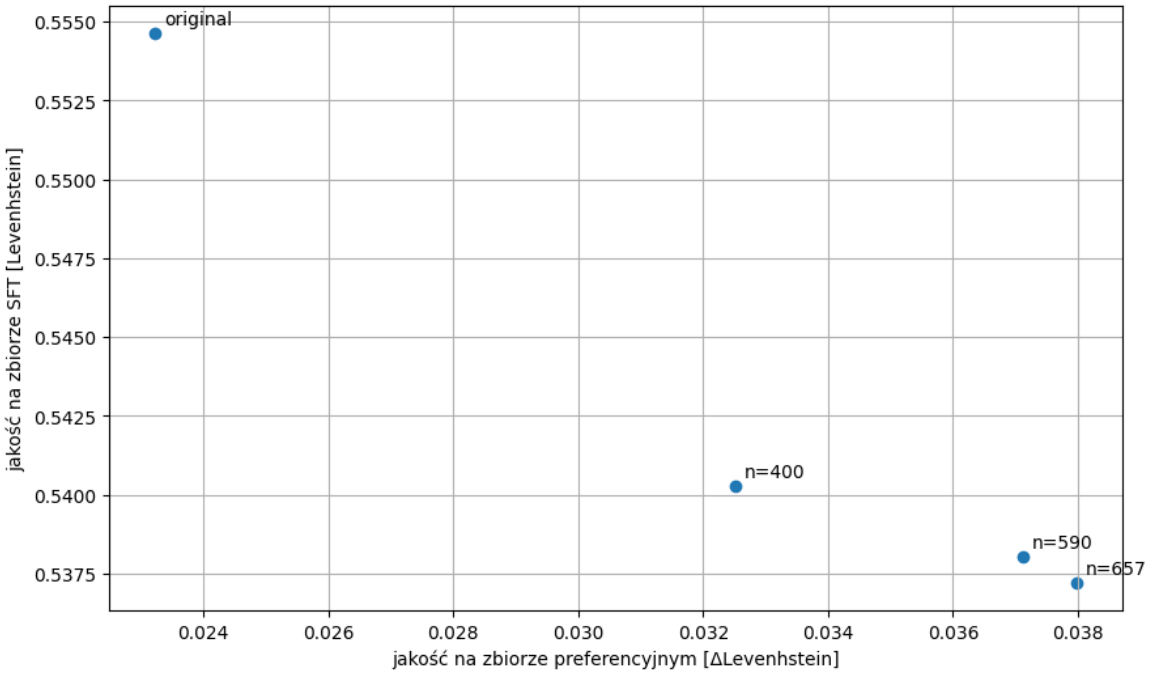
Punkt odniesienia (oznaczony jako "original") reprezentuje model bazowy po fazie SFT. Z definicji ma on najlepszą jakość na zbiorze SFT (najniższa wartość na osi Y), ale jednocześnie najsłabiej dopasowuje się do preferencji (najwyższa wartość na osi X).

W efekcie treningu na zwiększanej liczności danych preferencyjnych, model RLHF staje się coraz lepszy w dopasowywaniu się do preferencji, co widać po przesunięciu punktów w lewo na osi X. Niemniej jednak poprawa w zakresie preferencji odbywa się niewielkim kosztem zapominania wiedzy z fazy SFT. Można zaobserwować lekkie pogorszenie jakości na zbiorze SFT (wzrost wartości na osi Y), co jest często spotykanym kompromisem w procesach optymalizacyjnych.

4.1.2. Wyniki dla podejścia DPO

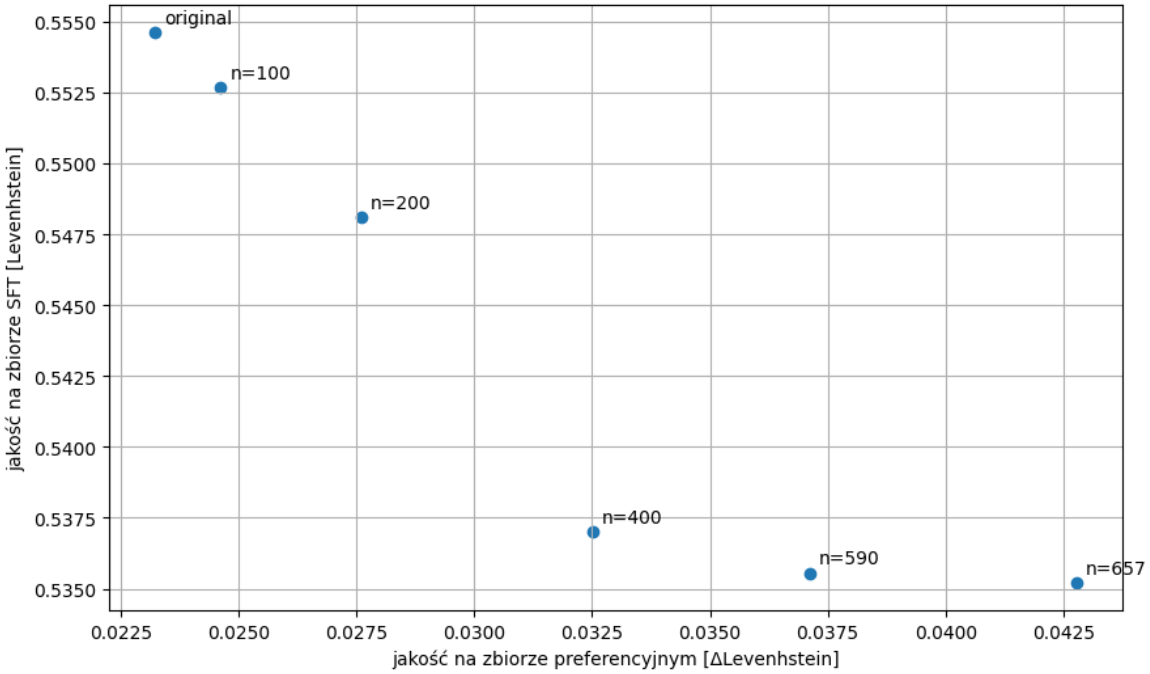
Wykres dla metody DPO umieszczony na rysunku 4.2 ma analogiczną strukturę do wykresu RLHF, ale przedstawia wyniki dla szerszego zakresu liczności zbiorów danych. Podobnie jak w podejściu opisanym powyżej, trening DPO prowadzi do poprawy jakości na zbiorze preferencyjnym (przesunięcie w lewo na osi X) kosztem jakości na zbiorze SFT (przesunięcie w górę na osi Y), natomiast trend jest jeszcze bardziej widoczny dzięki większej liczbie zbadanych zależności.

Porównując z PPO, DPO wydaje się być bardziej agresywną i skuteczną metodą optymalizacji preferencji. Przy największym zbiorze danych DPO osiąga znacznie lepszy wynik na zbiorze preferencyjnym niż PPO, chociaż odbywa się to kosztem nieco większego odchylenia od zadania bazowego.



Rysunek 4.1. Wyniki ilościowe dla metody RLHF

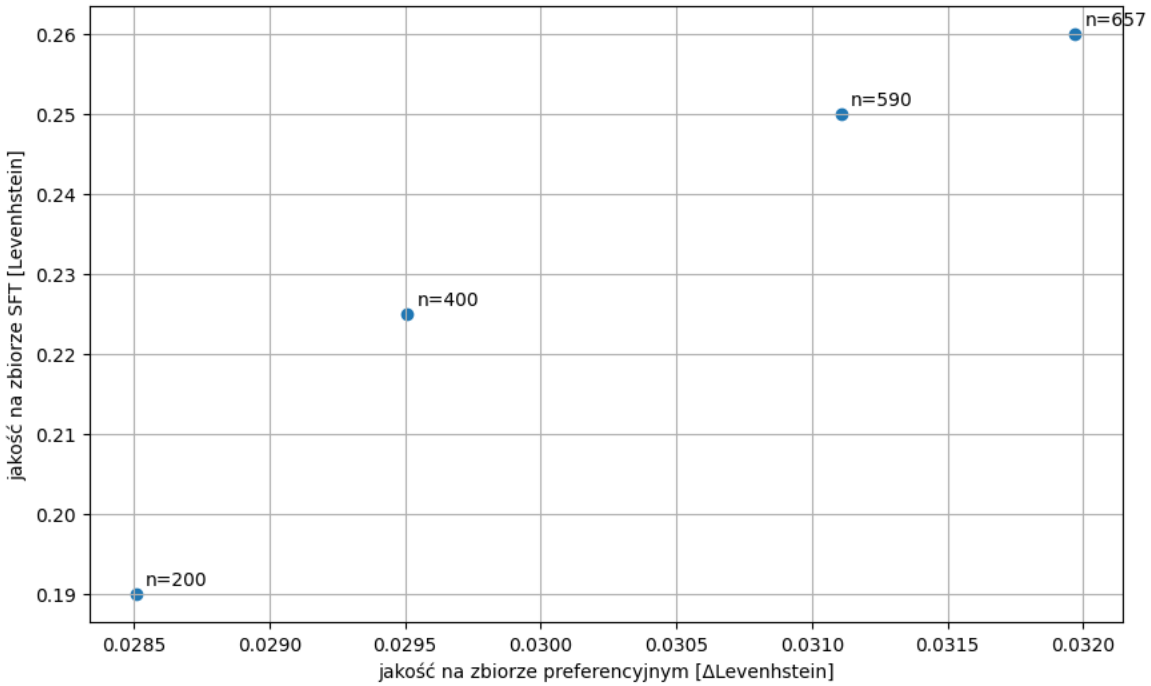
Punkty dla DPO w ramach zwiększania liczności zbioru danych tworzą wyraźną krzywą, wskazującą na trend lepszego dopasowania do preferencji użytkowników. Kształt krzywej nie jest jednak zgodny z przewidywanym - zgodnie z założeniem wyraźniejsza poprawa powinna być zaobserwowana przy niższej liczności zbioru.



Rysunek 4.2. Wyniki ilościowe dla metody DPO

4.1.3. Wyniki dla podejścia ORPO

Wykres podobieństwa Levenshteina dla metody ORPO przedstawiony na rysunku 4.3 pokazuje zupełnie inną dynamikę niż w RLHF i DPO. W miarę zwiększania liczby danych wyniki modelu poprawiają się na obu osiach jednocześnie – zarówno w zakresie dopasowania do preferencji, jak i jakości mierzonej na zbiorze SFT. Jest to zgodne z założeniami ORPO, które ma jednocześnie uczyć się samego zadania i preferencji.



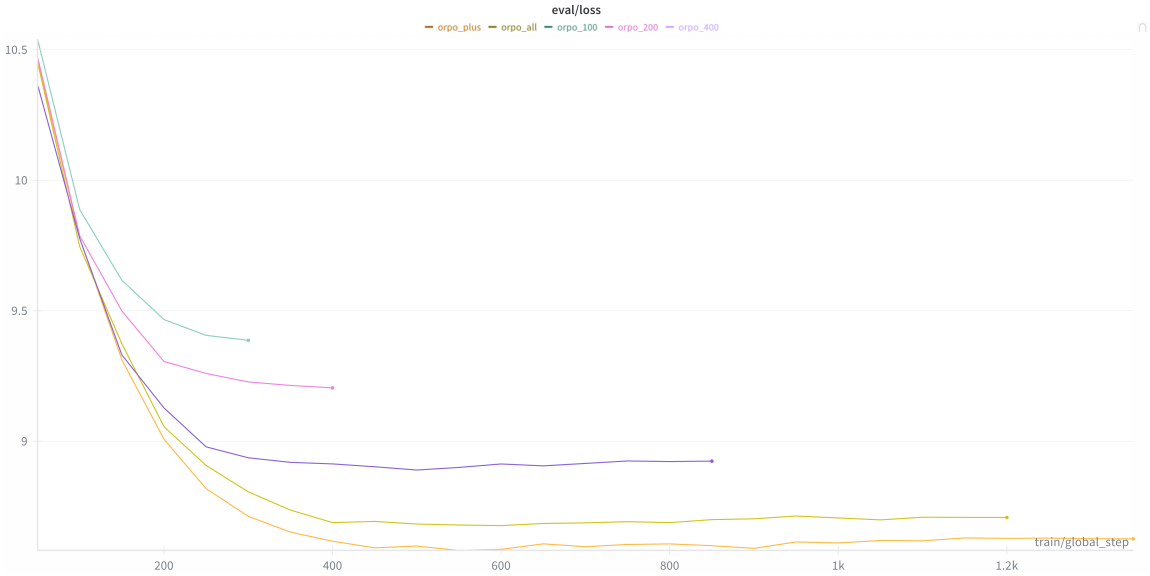
Rysunek 4.3. Wyniki ilościowe dla metody ORPO

Mimo że metoda ORPO skutecznie uczy się sygnału preferencji, jakość generowanych przez nią nazw domen jest wyraźnie gorsza w porównaniu do podejść dwuetapowych. Wygenerowane nazwy charakteryzują się niższą spójnością lingwistyczną, co sugeruje, że w obecnej formie nie osiągnęłyby one pożądanej użyteczności.

Główną przyczyną tego zjawiska jest prawdopodobnie niewystarczająca wielkość zbioru danych. W przypadku jednoetapowego treningu ORPO, zbiór ten musi jednocześnie dostarczyć sygnału do nauki podstawowego zadania generacyjnego oraz do optymalizacji preferencji. Hipotezę tę wspierają wykresy funkcji straty (rysunek 4.4) oraz funkcji nagrody (rysunek 4.5), które potwierdzają, że sam proces optymalizacji preferencji przebiega poprawnie. Wartość funkcji straty systematycznie spada, a wartość nagrody dla preferowanych odpowiedzi rośnie, stabilizując się na wysokim poziomie.

Oznacza to, że model skutecznie uczy się rozróżniać odpowiedzi zgodne z dostarczonymi preferencjami. Jednocześnie, oba wykresy pokazują, że większe zbiory danych prowadzą do osiągnięcia lepszych wyników końcowych (niższej straty i wyższej nagrody), co jest oczekiwanym zjawiskiem. Można zatem wnioskować, że metoda ORPO jest po-

prawna, lecz jej praktyczna skuteczność w tym zadaniu jest silnie uzależniona od dostępu do znacznie większego zbioru preferencji, niż jest to wymagane w przypadku metod dwuetapowych.



Rysunek 4.4. Trening ORPO - wykres funkcji straty

Rysunek 4.5. Trening ORPO - wykres funkcji nagrody

4.2. Analiza jakościowa

Celem analizy jakościowej jest ręczne porównanie wyjść wybranych, wytrenowanych modeli pod kątem ich użyteczności w zadaniu opisu stron. Ocenie podlegają w szczególności poprawność językowa, kreatywność oraz dopasowanie wygenerowanej nazwy

do opisu działalności. Poniżej przedstawiono wybrane przykłady wejść (opisów stron) oraz wygenerowane wyjścia modeli wraz z krótkim komentarzem oraz porównaniem rezultatów.

Wejście 1: E-commerce oferujący zdrową, certyfikowaną żywność bio: warzywa, owoce, produkty zbożowe, miody oraz przetwory. Strona promuje styl życia w zgodzie z naturą, a dodatkowo prowadzi blog z przepisami i poradami dotyczącymi zdrowego odżywiania.

Model referencyjny:

- bio-sklep.pl
- sklep-bio.pl
- eko-sklep.pl
- e-bio.pl

Model po RLHF:

- bio-sklep.pl
- biznesbio.pl
- eko-sklep.pl
- sklep-bio.pl

Model po DPO:

- biosklep.pl
- bio-sklep.pl
- ekobiznes.pl
- naturalne-eko.pl

Model po ORPO:

- e-commerce.pl
- żywnośćbio.pl
- stylżycia.pl
- blog.pl

Wejście 2: Serwis internetowy skupiający się na najnowszych trendach w świecie IT: sztucznej inteligencji, gadżetach, recenzjach sprzętu i oprogramowania. Oprócz artykułów dostępne są wywiady z ekspertami oraz forum dla pasjonatów technologii.

Model referencyjny:

- agencjait.pl
- agencja-it.pl
- agencjadlabiznesu.pl
- instalacjeit.pl

Model po RLHF:

- trendyit.pl
- technologia-online.pl
- agencjait.pl

- trendy-it.pl

Model po DPO:

- biznesit.pl
- inzynierit.pl
- trendyit24.pl
- trendit.pl

Model po ORPO:

- najnowszetrendy.pl
- trendy.pl
- serwistrendy.pl
- trendyit.pl

Wejście 3: Interaktywna strona oferująca gry, quizy i lekcje online z matematyki, języków obcych oraz przyrody. Wszystko zaprojektowane w kolorowej, przyjaznej formie, aby nauka była zabawą. Rodzice mogą śledzić postępy dziecka i korzystać z dodatkowych materiałów edukacyjnych.

Model referencyjny:

- e-shop.pl
- e-game.pl
- e-life.pl
- e-make.pl

Model po RLHF:

- e-shop.pl
- e-game.pl
- e-edu.pl
- mobilnanauka.pl

Model po DPO:

- naukaonline.pl
- zabawkidladzieci.pl
- zwiedzajswiat.pl
- instytut-edukacji.pl

Model po ORPO:

- interaktywnastrona.pl
- quizyilekcje.pl
- nauka.pl
- strona.pl

Na podstawie ręcznej analizy jakościowej można stwierdzić, że model po treningu DPO wykazuje najwyższy poziom kreatywności. Generowane przez niego propozycje często odbiegają od schematu, co sprawia, że są potencjalnie atrakcyjniejsze w kontekście zadań

wymagających innowacyjności. Jednocześnie należy zauważyć, że w części przypadków prowadzi to do nadmiernego oddalenia się od treści wejściowych, a wygenerowane nazwy nie zawsze zachowują ścisłą spójność z opisem. Zupełnie inaczej prezentuje się model po treningu ORPO, który niemal w całości opiera się na słowach kluczowych występujących w wejściowym opisie. Powoduje to wysoką zgodność z treścią, ale jednocześnie ogranicza różnorodność i oryginalność rezultatów. Z kolei model po treningu RLHF generuje wyniki bardzo zbliżone do modelu referencyjnego, lecz zauważalnie bardziej dopracowane – nazwy są spójniejsze, a zgodność z opisem pozostaje na wysokim poziomie.

4.3. Dyskusja

Analiza porównawcza trzech badanych metod pozwala na sformułowanie kompleksowych wniosków dotyczących ich skuteczności i charakterystyki w zadaniu generowania nazw domen internetowych. Zarówno DPO, jak i RLHF pozwoliły na skuteczne dostrajanie modelu bazowego, osiągając porównywalną, wysoką jakość na obu zbiorach danych. Kluczowym czynnikiem sukcesu w obu przypadkach okazało się wykorzystanie wstępnego treningu nadzorowanego, który zapewnił modelowi podstawowe umiejętności generacyjne przed właściwą optymalizacją pod kątem preferencji.

Choć ogólna skuteczność obu metod jest zbliżona, wnikliwa analiza wyników pozwala dostrzec subtelne różnice (patrz tabela 4.1). DPO okazało się metodą bardziej agresywną i skuteczną w strojeniu preferencji. Przy największym zbiorze danych osiągnęło ono najlepsze dopasowanie do danych preferencyjnych, co odbyło się jednak kosztem nieco większego odchylenia od pierwotnej wiedzy modelu SFT. Z kolei PPO stanowi stabilną alternatywę, która również skutecznie poprawia model, ale w mniejszym stopniu ingeruje w jego bazową politykę. Wybór między tymi dwiema metodami zależy więc od tego, czy priorytetem jest maksymalne dopasowanie do preferencji, czy zachowanie jak największej wierności wobec modelu referencyjnego.

Podejście	Jakość na zbiorze SFT	Różnica jakości na zbiorze preferencyjnym
RLHF	**0.537**	0.038
DPO	0.535	**0.043**
ORPO	0.261	0.032

Tabela 4.1. Porównanie wyników trzech analizowanych podejść

W odróżnieniu od opisanych wcześniej metod, podejście ORPO uzyskało w ostatecznej ocenie wyraźnie niższe wyniki. Główną przyczyną jest brak oddzielnej, intensywnej fazy dostrajania na dużym zbiorze danych SFT. Pominięcie tego etapu sprawiło, że model nie zdołał w pełni opanować podstawowych wzorców językowych i generacyjnych, co negatywnie wpłynęło na jego zdolność do generalizacji. Należy jednak podkreślić, że wewnętrzne metryki treningowe (spadek funkcji straty i wzrost funkcji nagrody) oraz trend na wykresie Levenshteina pokazują, że sama metoda ORPO prawdopodobnie działa

poprawnie i skutecznie uczy się preferencji. Jej słabszy wynik końcowy nie świadczy o wadzie algorytmu, lecz o jego silnej zależności od dostępu do dużego i zróżnicowanego zbioru danych. Można przypuszczać, że przy znacznie większym zbiorze preferencyjnym, wyniki dla ORPO mogłyby zrównać się z pozostałymi metodami [41].

Z kolei z perspektywy efektywności zasobowej, DPO pozostaje najbardziej praktyczne - minimalizuje koszty poprzez eliminację modelu nagrody, a zarazem dobrze skaluje się z rosnącą licznością zbioru preferencyjnego.

4.3.1. Istotność danych sprzedażowych

Mimo że dane sprzedażowe były znikome, ich dołączenie do zbioru preferencyjnego przyniosło relatywnie duży wzrost jakości na zbiorze preferencyjnym, co widać na rysunkach 4.1, 4.2 oraz 4.3 po różnicy między konfiguracjami z najliczniejszym zbiorem ($n = 657$) a nieco mniejszym ($n = 590$) — zwiększenie udziału przykładów sprzedażowych znacząco poprawia wynik na zbiorze preferencyjnym. Wynik ten silnie wskazuje, że sygnał sprzedażowy jest niezwykle istotny. Obecnie przykłady oparte na sprzedaży miały taką samą wagę jak pozostałe preferencje i można przypuszczać, że nadanie im wyższej wagi dodatkowo wzmocniłoby pozytywny efekt. Dane sprzedażowe zostały zebrane za pomocą prototypowego modelu rekomendacji NASK wykorzystywanego przez jednego z ich partnerów w jego kanale sprzedażowym, a dalsze zwiększanie wolumenu tych danych powinno przekładać się na poprawę jakości modelu.

Rozszerzona analiza wskazuje, że dane sprzedażowe stanowią wartościowe źródło preferencji, ponieważ odzwierciedlają realne decyzje użytkowników, a nie tylko subiektywne oceny anotatorów. Dzięki temu można je traktować jako wskaźnik wartości rynkowej, który eliminuje poprawne, ale mało atrakcyjne biznesowo propozycje nazw domen. Co więcej, pozyskiwanie takich danych jest znacznie bardziej skalowalne i mniej kosztowne niż ręczne anotacje.

Należy jednak zauważyć, że sygnał sprzedażowy nie jest wolny od ograniczeń. Na decyzję o rejestracji domeny wpływają również czynniki zewnętrzne, takie jak cena, aktualne trendy rynkowe, krótkoterminowe preferencje użytkowników czy dodatkowe usługi sprzedawane w pakiecie. Oznacza to, że dane sprzedażowe mogą być częściowo zaszumione i nie zawsze jednoznacznie przekładają się na jakość samej nazwy. W przyszłości warto pewnie rozważyć nadanie im dynamicznej wagi w procesie treningu, aby zwiększyć odporność modelu na takie zakłócenia.

5. Podsumowanie i wnioski

Niniejsza praca magisterska stanowiła weryfikację skuteczności nowoczesnych metod uczenia preferencyjnego w kreatywnym zadaniu generacji nazw domen internetowych. Przeprowadzone badania i analiza wyników pozwalają na sformułowanie kluczowych wniosków oraz nakreślenie obiecujących kierunków dalszych prac.

5.1. Wnioski końcowe

Przeprowadzone eksperymenty jednoznacznie potwierdzają, że zaawansowane metody optymalizacji, takie jak RLHF, DPO i ORPO, są skutecznymi narzędziami do trenowania modeli językowych pod kątem specyficznych i niesprecyzowanych kryteriów, jakimi cechuje się dobra nazwa domeny. Kreatywność i zgodność z oczekiwaniami użytkowników, trudne do uchwycenia przy treningu nadzorowanym, mogą być dobrze odzwierciedlane przy użyciu sygnału preferencji.

Analiza porównawcza wykazała, że podejście DPO zapewnia optymalny kompromis między jakością generowanych nazw a szybkością i prostotą implementacji. Eliminując potrzebę trenowania osobnego modelu nagrody, DPO znacząco obniża barierę obliczeniową w porównaniu do klasycznego RLHF, jednocześnie osiągając w tym badaniu najwyższą skuteczność w dopasowaniu modelu do preferencji rynkowych. Jest to szczególnie istotne w zastosowaniach komercyjnych, gdzie kluczowe znaczenie mają skalowalność oraz możliwość szybkiego iterowania i wdrażania uzyskanych rezultatów.

Metoda RLHF również okazała się skuteczna, lecz jej wieloetapowa natura i większe wymagania obliczeniowe czynią ją mniej praktyczną alternatywą. Z kolei podejście ORPO w powyższym badaniu nie dorównało jakością pozostałym metodom. Jego głównym ograniczeniem okazała się silna zależność od liczności zbioru danych, który w jednoetapowym procesie musiałby dostarczyć wystarczającego sygnału zarówno do nauki zadania, jak i optymalizacji preferencji.

Szczególnym wkładem niniejszej pracy było uwzględnienie danych sprzedażowych jako części zbioru preferencyjnego. Mimo niewielkiego wolumenu przełożyły się one na wyraźną poprawę jakości modelu, wskazując na ich wysoką wartość jako obiektywnego wskaźnika atrakcyjności nazw domen. Wyniki sugerują, że dalsze zwiększanie udziału danych sprzedażowych w procesie optymalizacji może stanowić kluczowy kierunek rozwoju badań w tej dziedzinie.

5.2. Znaczenie praktyczne i implikacje zastosowań

Otrzymane rezultaty mają istotne implikacje praktyczne. Włączenie danych biznesowych do procesu treningowego wskazuje drogę do tworzenia systemów, które nie tylko spełniają kryteria lingwistyczne, ale również odpowiadają rzeczywistym preferencjom użytkowników. W kontekście generacji nazw domen przekłada się to na rozwiązania wspie-

rające proces rejestracji i marketingu internetowego, podnoszące skuteczność działań firm i instytucji.

Co istotne, to podejście może zostać przeniesione na inne zadania biznesowe, w których mierzalne dane rynkowe (między innymi konwersje takie jak kliknięcia czy zakupy) mogą pełnić rolę sygnału nagrody. Dotyczy to w szczególności takich obszarów jak generowanie nazw produktów, sloganów reklamowych czy treści marketingowych, gdzie atrakcyjność i dopasowanie do odbiorcy są równie trudne do formalizacji jak w przypadku domen internetowych.

W szerszej perspektywie uzyskane wyniki podkreślają, że integracja modeli językowych z realnym środowiskiem biznesowym stanowi naturalny kierunek rozwoju systemów generatywnych. Pozwala to zwiększyć ich użyteczność komercyjną, a jednocześnie tworzy połączenie pomiędzy zaawansowanymi badaniami teoretycznymi a praktycznymi potrzebami rynku.

5.3. Ograniczenia pracy i kierunki dalszych badań

Wyniki i wnioski z niniejszej pracy otwierają drogę do wielu interesujących kierunków dalszych badań, które mogłyby udoskonalić prezentowane podejście. Należą do nich między innymi:

- Rozbudowa i wzbogacenie zbiorów danych - najbardziej bezpośrednią metodą poprawy wyników, zwłaszcza dla metody ORPO, jest dalsze gromadzenie danych preferencyjnych. Można to osiągnąć zarówno poprzez anotację większej liczby ręcznych ocen, jak i integrację z szerszym zakresem danych sprzedażowych. Ciekawym rozwinięciem byłoby również uwzględnienie ważonego współczynnika dla danych sprzedażowych, gdzie preferencja nie byłaby binarna, ale ważona poziomem zaangażowania użytkownika (kliknięcie, dodanie do koszyka, zakup). W praktyce taki współczynnik można bezpośrednio wyznaczyć na podstawie istniejących danych sprzedażowych, lecz nie został on uwzględniony w przeprowadzonych badaniach.
- Zaawansowana ocena jakości z użyciem LLM - obecna ocena opiera się na metryce liczonej na podstawie odległości Levenshteina, która ma swoje ograniczenia. Obiecującym kierunkiem jest stworzenie dedykowanego modelu oceniającego lub użycie już istniejącego rozwiązania z literatury. Taki model mógłby zostać użyty do dyskretnej oceny bardziej abstrakcyjnych kryteriów, dostarczając jeszcze bardziej wymownej metryki jakości.
- Testy A/B i walidacja biznesowa - ostatecznym dowodem skuteczności modelu są wyniki w świecie rzeczywistym. Kolejnym krokiem mogłoby być wdrożenie systemu do testów A/B, w ramach którego najlepsze propozycje generowane przez model byłyby prezentowane prawdziwym klientom. Mierzenie wskaźników takich jak współczynnik klikalności czy konwersja pozwoliłoby na bezpośrednią walidację biznesową i dalszą optymalizację w oparciu o najbardziej wiarygodny sygnał zwrotny.

- Eksploracja nowych podejść i architektur - dziedzina sztucznej inteligencji rozwija się w błyskawicznym tempie. Dalsze prace mogłyby objąć zbadanie nowszych algorytmów optymalizacji preferencyjnej, które potencjalnie mogłyby zaoferować jeszcze lepszą efektywność. Warto byłoby również przetestować całą metodykę badawczą na wielojęzycznych modelach generatywnych, co pozwoliłoby na stworzenie rozwiązania o zasięgu globalnym.

Bibliografia

[1] A. Vaswani, N. Shazeer, N. Parmar i in., "Attention is all you need", *Advances in neural information processing systems*, t. 30, 2017.

[2] M. Lewis, Y. Liu, N. Goyal i in., "BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension", *arXiv preprint arXiv:1910.13461*, 2019.

[3] A. Q. Jiang, A. Sablayrolles, A. Mensch i C. Bamford, "Mistral 7B", *arXiv preprint arXiv:2310.06825*, t. 3, 2023.

[4] S. Dadas, *A repository of Polish NLP resources*, Dostęp zdalny (11.05.2025): `https://github.com/sdadas/polish-nlp-resources`, 2019.

[5] K. Ociepa, Ł. Flis, K. Wróbel, A. Gwoździej i R. Kinas, "Bielik 7B v0.1: A Polish Language Model - Development, Insights and Evaluation", *arXiv preprint arXiv:2410.18565*, 2024.

[6] L. P. Kaelbling, M. L. Littman i A. W. Moore, "Reinforcement learning: A survey", *Journal of artificial intelligence research*, t. 4, s. 237–285, 1996.

[7] K. Arulkumaran, M. P. Deisenroth, M. Brundage i A. A. Bharath, "Deep reinforcement learning: A brief survey", *IEEE signal processing magazine*, t. 34, nr. 6, s. 26–38, 2017.

[8] Y. Bai, A. Jones, K. Ndousse i in., "Training a helpful and harmless assistant with reinforcement learning from human feedback", *arXiv preprint arXiv:2204.05862*, 2022.

[9] J. Schulman, F. Wolski, P. Dhariwal, A. Radford i O. Klimov, "Proximal policy optimization algorithms", *arXiv preprint arXiv:1707.06347*, 2017.

[10] R. Rafailov, A. Sharma, E. Mitchell, C. D. Manning, S. Ermon i C. Finn, "Direct preference optimization: Your language model is secretly a reward model", *Advances in neural information processing systems*, t. 36, s. 53 728–53 741, 2023.

[11] J. Hong, N. Lee i J. Thorne, "Orpo: Monolithic preference optimization without reference model", *arXiv preprint arXiv:2403.07691*, 2024.

[12] K. Papineni, S. Roukos, T. Ward i W.-J. Zhu, "Bleu: a method for automatic evaluation of machine translation", w *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, 2002, s. 311–318.

[13] C.-Y. Lin, "Rouge: A package for automatic evaluation of summaries", w *Text summarization branches out*, 2004, s. 74–81.

[14] L. Yujian i L. Bo, "A normalized Levenshtein distance metric", *IEEE transactions on pattern analysis and machine intelligence*, t. 29, nr. 6, s. 1091–1095, 2007.

[15] Y. Liu, D. Iter, Y. Xu, S. Wang, R. Xu i C. Zhu, "G-eval: NLG evaluation using gpt-4 with better human alignment", *arXiv preprint arXiv:2303.16634*, 2023.

[16] J. Fu, S.-K. Ng, Z. Jiang i P. Liu, "Gptscore: Evaluate as you desire", *arXiv preprint arXiv:2302.04166*, 2023.

[17] W. Zhao, M. Peyrard, F. Liu, Y. Gao, C. M. Meyer i S. Eger, "MoverScore: Text generation evaluating with contextualized embeddings and earth mover distance", *arXiv preprint arXiv:1909.02622*, 2019.

[18] *AI Business Name Generator*, Dostęp zdalny (28.08.2025): https://www.shopify.com/tools/business-name-generator.

[19] *GoDaddy Domain Name Generator*, Dostęp zdalny (28.08.2025): https://www.godaddy.com/domains/domain-name-generator.

[20] *Domain Name Generator*, Dostęp zdalny (28.08.2025): https://www.namecheap.com/domains/domain-name-generator/.

[21] *Transformer Architectures*, Dostęp zdalny (11.05.2025): https://huggingface.co/learn/llm-course/en/chapter1/6.

[22] C. Yan, *Understanding Transformer Architectures: Decoder-Only, Encoder-Only, and Encoder-Decoder Models*, Dostęp zdalny (11.05.2025): https://chrisyandata.medium.com/understanding-transformer-architectures-decoder-only-encoder-only-and-encoder-decoder-models-285a17904d84.

[23] D. Naik, I. Naik i N. Naik, "Decoder-only transformers: the brains behind generative AI, large language models and large multimodal models", w *The International Conference on Computing, Communication, Cybersecurity & AI*, Springer, 2024, s. 315–331.

[24] C. R. Wolfe, *Decoder-Only Transformers: The Workhorse of Generative LLMs*, Dostęp zdalny (11.05.2025): https://cameronrwolfe.substack.com/p/decoder-only-transformers-the-workhorse.

[25] *PLLuM*, Dostęp zdalny (03.09.2025): https://huggingface.co/collections/CYFRAGOVPL.

[26] *PLLuM - Polish Large Language Model*, Dostęp zdalny (03.09.2025): https://pllum.org.pl/.

[27] J. Murel i E. Kavlakoglu, *What is reinforcement learning?*, Dostęp zdalny (11.05.2025): https://www.ibm.com/topics/reinforcement-learning.

[28] M. A. Wiering i M. Van Otterlo, "Reinforcement learning", *Adaptation, learning, and optimization*, t. 12, nr. 3, s. 729, 2012.

[29] O. Berger-Tal, J. Nathan, E. Meron i D. Saltz, "The Exploration-Exploitation Dilemma: A Multidisciplinary Framework", *PLOS ONE*, t. 9, nr. 4, s. 1–8, kw. 2014. DOI: 10.1371/journal.pone.0095693.

[30] *Illustrating Reinforcement Learning from Human Feedback (RLHF)*, Dostęp zdalny (11.05.2025): https://huggingface.co/blog/rlhf.

[31] H. Lee, S. Phatale, H. Mansoor i in., "Rlaif: Scaling reinforcement learning from human feedback with ai feedback", 2023.

[32] J. Goldstein, M. Kantrowitz, V. Mittal i J. Carbonell, "Summarizing text documents: Sentence selection and evaluation metrics", w *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, 1999, s. 121–128.

[33] D. K. Po, "Similarity based information retrieval using Levenshtein distance algorithm", *Int. J. Adv. Sci. Res. Eng*, t. 6, nr. 04, s. 06–10, 2020.

[34] M. Wiszenko, "System do przeprowadzania konkursów przetwarzania języka naturalnego", Wydział Elektroniki i Technik Informacyjnych, Politechnika Warszawska, 2023.

[35] Y. Graham, "Re-evaluating automatic summarization with BLEU and 192 shades of ROUGE", w *Proceedings of the 2015 conference on empirical methods in natural language processing*, 2015, s. 128–137.

[36] W. H. Gomaa, A. A. Fahmy i in., "A survey of text similarity approaches", *international journal of Computer Applications*, t. 68, nr. 13, s. 13–18, 2013.

[37] T. Hu i X.-H. Zhou, "Unveiling llm evaluation focused on metrics: Challenges and solutions", *arXiv preprint arXiv:2404.09135*, 2024.

[38] P. Arabas, P. Jaskóła, M. Kamola i M. Karpowicz, "Analysis and modeling of domain registration process", *Journal of Telecommunications and Information Technology*, nr. 2, s. 63–73, 2012.

[39] *Label Studio*, Dostęp zdalny (13.09.2025): `https://labelstud.io/`.

[40] S. Dadas, Dostęp zdalny (03.09.2025): `https://huggingface.co/sdadas/polish-bart-base`.

[41] J. Kaplan, S. McCandlish, T. Henighan i in., "Scaling laws for neural language models", *arXiv preprint arXiv:2001.08361*, 2020.

Spis rysunków

2.1 Architektura Transformer [1] 14
2.2 Architektura Transformer typu decoder-only [24] 15
2.3 Proces treningu RLHF - krok SFT oraz budowa modelu nagrody [30] 17
3.1 Uproszczony schemat procedury badawczej dla metody RLHF 23
3.2 Przykładowa anotacja przy użyciu programu Label Studio [39] 26
4.1 Wyniki ilościowe dla metody RLHF 30
4.2 Wyniki ilościowe dla metody DPO 30
4.3 Wyniki ilościowe dla metody ORPO 31
4.4 Trening ORPO - wykres funkcji straty 32
4.5 Trening ORPO - wykres funkcji nagrody 32

Spis tabel

3.1 Charakterystyka zbiorów danych 24
3.2 Szczegółowa charakterystyka zbioru danych SFT 24
3.3 Szczegółowa charakterystyka zbioru preferencyjnego ręcznych anotacji 25
3.4 Szczegółowa charakterystyka zbioru preferencyjnego danych sprzedażowych 25
4.1 Porównanie wyników trzech analizowanych podejść 35