ENSEMBLE BERT4REC FOR RECSYS 2025 CHALLENGE

RAPORT TECHNICZNY

Adam Stajek
adamstajek@student.agh.edu.pl

Maksym Szemer
szemermaksym@student.agh.edu.pl

Adam Tokarz
adamtokarz@student.agh.edu.pl

13 sierpnia 2025

ABSTRACT

W niniejszym raporcie przedstawiono rozwiązanie opracowane na potrzeby konkursu RecSys Challenge 2025, którego celem było stworzenie uniwersalnych reprezentacji behawioralnych użytkowników (Universal Behavioral Profiles) na podstawie ich aktywności w systemie. Zaproponowana metoda wykorzystuje uczenie zespołowe oparte na modelach BERT4Rec — sekwencyjnych modelach typu transformer z mechanizmem dwukierunkowej self-attention, trenowanych osobno dla każdego typu interakcji (zakupy, dodania/usunięcia z koszyka, wyszukiwania, odwiedziny stron). W celu uzyskania końcowej reprezentacji użytkownika połączono embeddingi z różnych modeli za pomocą konkatenacji. Otrzymane reprezentacje zostały następnie ocenione w kilku zadaniach predykcyjnych za pomocą przygotowanej przez organizatorów architektury ewaluacyjnej. Wyniki eksperymentów wykazały, że nasza metoda przewyższa rozwiązanie bazowe oparte na klasycznej inżynierii cech, osiągając wyższą skuteczność w różnych zadaniach i zapewniając dobrą generalizację. Zajęcie około 25. miejsca wśród blisko 400 zespołów potwierdza konkurencyjność zaproponowanego podejścia.

1 Wstęp

Wraz z nieustannie rosnącą ilością dostępnych informacji w sieci, systemy rekomendacyjne stały się nieodzownym narzędziem wspierającym użytkowników w podejmowaniu decyzji zakupowych oraz przeszukiwaniu rozbudowanych zasobów treści. Ich rola w personalizacji usług oraz w łagodzeniu zjawiska przeciążenia informacyjnego jest obecnie trudna do przecenienia. W szczególności w kontekście platform e-commerce, mediów streamingowych czy serwisów informacyjnych, systemy te wspierają zarówno użytkowników, jak i dostawców usług, zwiększając skuteczność dotarcia do interesujących treści i produktów [1].

Jednym z kluczowych wyzwań w rozwoju systemów rekomendacyjnych pozostaje brak jednoznacznych informacji zwrotnych od użytkowników. W rzeczywistości preferencje użytkowników są często ukryte w interakcjach z systemem – kliknięciach, wyszukiwaniach, zakupach czy przeglądanych stronach – które cechują się szumem i brakiem jednoznaczności [2]. W odpowiedzi na to wyzwanie duże zainteresowanie badawcze zyskało modelowanie zachowań użytkowników (*User Behavior Modeling*, UBM), którego celem jest ekstrakcja reprezentacji preferencji użytkownika na podstawie historii jego aktywności [2].

Współczesne podejścia do modelowania zachowań użytkowników korzystają z coraz bardziej zaawansowanych technik, w tym metod opartych na uczeniu głębokim. Techniki uczenia głębokiego umożliwiają efektywne wychwytywanie nieliniowych i złożonych zależności między użytkownikami a przedmiotami rekomendacji. Wykorzystanie takich architektur jak sieci rekurencyjne, konwolucyjne czy mechanizmy uwagi pozwoliło na znaczącą poprawę jakości rekomendacji w porównaniu do tradycyjnych metod [1]. Co istotne, modele uczenia głębokiego pozwalają również na integrację wielu źródeł danych kontekstowych – takich jak tekst, obraz czy informacje czasowo-przestrzenne – co zwiększa ich elastyczność i zastosowalność w praktycznych systemach rekomendacyjnych.

W szczególności, nowsze modele sekwencyjne, takie jak `BERT4Rec`, próbują przezwyciężyć ograniczenia klasycznych podejść jednokierunkowych, stosując dwukierunkowe mechanizmy samo-uwagi. Pozwalają one na uwzględnienie kontekstu zarówno poprzedzającego, jak i następującego dane zdarzenie w sekwencji zachowań użytkownika, co prowadzi do bardziej trafnych i stabilnych predykcji [3].

W świetle powyższego, badanie i rozwój metod modelowania zachowań użytkowników stanowi obecnie jeden z najważniejszych kierunków w projektowaniu nowoczesnych systemów rekomendacyjnych. W niniejszym raporcie przedstawiono metodę modelowania preferencji użytkowników opartą na uczeniu zespołowym oraz modelu sekwencyjnym BERT4Rec.

2 Opis konkursu

W kontekście rosnącego znaczenia modelowania zachowań użytkowników, konkurs **RecSys Challenge 2025** stanowi próbę standaryzacji i ujednolicenia podejścia do tego typu problemów analitycznych. Współczesne przedsiębiorstwa coraz częściej polegają na uczeniu maszynowym w podejmowaniu decyzji biznesowych. Typowe zadania w tym obszarze to między innymi rekomendacje produktów, predykcje skłonności zakupowych, przewidywanie odpływu klientów (churn), czy estymacja wartości klienta w czasie (lifetime value). Niezależnie od konkretnego celu, fundamentem dla większości z tych zadań pozostają logi interakcji z systemem – takie jak zakupy, przeglądane strony czy wyszukiwania.

Konkurs promuje koncepcje *Universal Behavioral Profiles* – ujednoliconych reprezentacji użytkowników, które mają uchwycić kluczowe wzorce ich zachowań na podstawie przeszłych interakcji. Celem jest stworzenie takiej formy reprezentacji, która będzie uniwersalnie użyteczna w szerokim zakresie zadań predykcyjnych, bez potrzeby ich dostrajania pod konkretne cele. Dzięki temu możliwe jest projektowanie modeli lepiej uogólniających, co w praktyce przekłada się na większą odporność na zmienne warunki zastosowań biznesowych.

Zadanie polega na przygotowaniu takich właśnie reprezentacji użytkowników, bazując na dostarczonym zbiorze danych obejmującym sześć typów zdarzeń (zakupy, interakcje z koszykiem, wizyty na stronach i zapytania wyszukiwawcze). Ostateczne oceny modeli będą przeprowadzane na podstawie skuteczności tych reprezentacji w różnych zadaniach, zarówno ujawnionych jak i ukrytych.

Do zadań owartych, tj. jawnych dla uczestników, należą:

- Predykcja odpływu użytkownika (*churn prediction*): ocena czy aktywny użytkownik (taki który dokonał przynajmniej jednego zakupu) nie dokona żadnej transakcji w ciągu kolejnych 14 dni
- Predykcja produktów (*product propensity*): wskazanie które produkty z określonego zbioru użytkownik z największym prawdopodobieństwem zakupi w ciągu 14 dni
- Predykcja kategorii (*category propensity*): przewidywanie w których kategoriach produktowych użytkownik najprawdopodobniej dokona zakupu w najbliższych 14 dniach

Poza tym, organizatorzy przewidzieli zestaw zadań ukrytych, tj. takich których szczegóły nie są znane uczestnikom w trakcie trwania konkursu. Tego rodzaju zadania mają na celu sprawdzenie czy przygotowane profile użytkowników rzeczywiście dobrze uogólniają wzorce zachowań, a nie są jedynie dopasowane do znanych celów. Dopiero po zakończeniu konkursu zostaną one ujawnione, razem z kodem umożliwiającym replikację wyników.

Ocena zgłoszeń odbywa się głównie na podstawie wartości wskaźnika AUROC. W przypadku zadań związanych z rekomendacją produktów i kategorii, dodatkowo brane są pod uwagę miary nowości (*novelty*) i różnorodności (*diversity*), a końcowy wynik stanowi ważoną sumę:

$$\text{SCORE} = 0.8 \times \text{AUROC} + 0.1 \times \text{Novelty} + 0.1 \times \text{Diversity} \tag{1}$$

Wyniki cząstkowe z każdego zadania są przedstawiane w postaci osobnych rankingów. Końcowy wynik konkursowy wyliczany jest metodą Borda, w której uczestnicy zdobywają punkty na podstawie pozycji w rankingach poszczególnych zadań [4]. Takie podejście premiuje rozwiązania spójne dobre w różnych kontekstach, zamiast tych które są wybitne tylko w jednym, ale słabe w pozostałych.

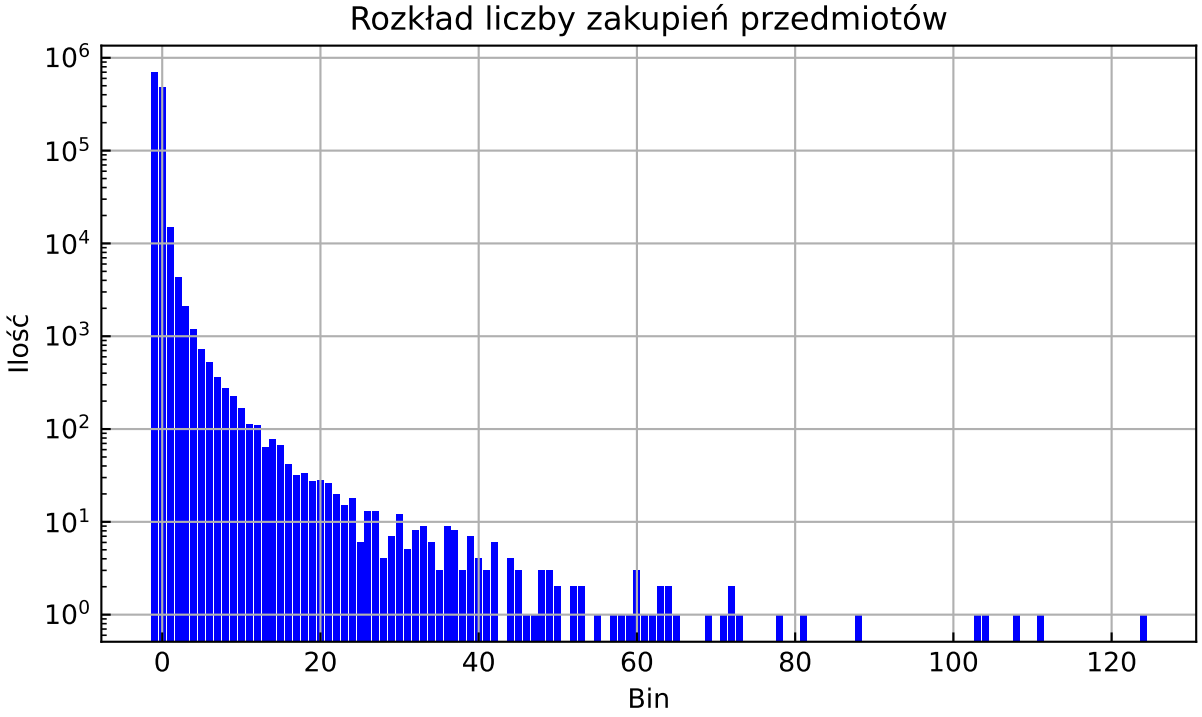
3 Zbiór danych

Na potrzeby konkursu organizatorzy udostępnili zanonimizowany zbiór danych, zawierający rzeczywiste logi interakcji użytkowników z systemem. Wszystkie zarejestrowane zdarzenia mogą być wykorzystane do tworzenia uniwersalnych profili behawioralnych (UBP) użytkowników, jednak uczestnicy zostali zobowiązani do przygotowania takich profili jedynie dla wybranego podzbioru użytkowników, który posłuży do trenowania i oceny modeli.

Zbiór danych obejmuje sześć typów zdarzeń, odzwierciedlających różne aktywności użytkowników, oraz zestaw atrybutów produktów. Pierwsze trzy typy zdarzeń dotyczą interakcji z koszykiem zakupowym: *product_buy* (zakup produktu), *add_to_cart* (dodanie produktu do koszyka) oraz *remove_from_cart* (usunięcie produktu z koszyka). W każdym z tych przypadków rejestrowane są identyfikator użytkownika (`client_id`), znacznik czasu (`timestamp`) oraz identyfikator produktu (`sku`).

Kolejna kategoria danych to *product_properties*, zawierająca cechy opisujące produkty. Dla każdego produktu zapisany jest jego identyfikator (`sku`), identyfikator kategorii (`category`), przedział cenowy (`price`), a także skompresowane za pomocą kwantyzacji osadzenia tekstowe nazwy produktu (`embedding`).

Pozostałe dwa typy zdarzeń to *page_visit* i *search_query*. W przypadku wizyt na stronach internetowych zapisywany jest identyfikator użytkownika, czas wizyty oraz identyfikator odwiedzonej strony (`url`). Warto podkreślić, że nie



Rysunek 1: Spośród wszystkich dostępnych produktów, 57.6% nigdy nie zostało kupionych, a tylko 19.4% zostało zakupionych więcej niż jeden raz

ma dostępnych informacji o tym, jakie konkretnie treści zostały wyświetlone na danej stronie. Natomiast dane typu *search_query* zawierają informacje o zapytaniach wyszukiwanych przez użytkownika, poza identyfikatorem klienta i znacznikiem czasu, zapisane jest również osadzenie zapytania, również poddane kompresji metodą kwantyzacji.

Zbiór danych obejmuje łącznie około 170 milionów zdarzeń. Najliczniejsze są wizyty na stronach (ok, 150 milionów), następnie zapytania wyszukiwania (ok. 9.6 miliona), dodania do koszyka (ok. 5.2 miliona), a także zakupy i usunięcia z koszyka (po około 1.7 miliona zdarzeń). Dane pochodzą od około 19 milionów unikalnych użytkowników.

Po dokładnej analizie zbioru danych, można zauważyć, że pomimo wielu rekordów z wydarzeniami, duża część użytkowników nie dokonała żadnych interakcji ze stroną internetową lub wykonała ich bardzo niewiele. Świadczy to o tym, że zbiór jest rzadki, tzn. większość wartości to zera lub brak danych, tylko niewielki procent możliwych wartości jest wypełniony. Dostarczony przez organizatorów zestaw danych dość trafnie odwzorowuje rzeczywiste zbiory danych.

4 Metodologia

4.1 Architektura ewaluacji

Organizatorzy przygotowali specjalną architekturę sieci neuronowej, która służy do predykcji konkretnych zachowań użytkowników, opisanych w rozdziale 2, na podstawie wytrenowanych przez nas embeddingów [5]. Model został zaprojektowany do uczenia reprezentacji wejściowych (embeddingów) w zadaniach klasyfikacyjnych lub regresyjnych, z wykorzystaniem frameworka PyTorch Lightning dla uproszczonego treningu i walidacji. Składa się z dwóch głównych komponentów:

1. BottleneckBlock (Inverted Bottleneck) - rozszerza wejście z wymiaru cienkiego do szerszego, następnie stosuje nieliniowość GELU i projektuje z powrotem do wymiaru cienkiego. Ten blok służy do zwiększenia zdolności reprezentacyjnej bez dużego narzutu parametrowego.
2. Net – właściwa sieć neuronowa. Na wejściu znajduje się warstwa liniowa redukująca wymiar embeddingów, po której następuje warstwa normalizacji warstwy (LayerNorm). Dalej sieć składa się z trzech sekwencyjnych bloków: normalizacji, BottleneckBlocka, oraz połączenia rezydualnego. Finalnie stosowana jest kolejna normalizacja i warstwa wyjściowa mapująca na odpowiedni wymiar wyjściowy.

Całość tworzy modularny i rozszerzalny model typu MLP z blokami bottleneckowymi, przeznaczony do eksperymentów w środowiskach, w których kluczowe jest ograniczenie biasu strukturalnego (np. brak konwolucji czy mechanizmów uwagi). Dzięki zastosowaniu residual connections i normalizacji warstw, architektura jest stabilna podczas treningu, nawet przy większej głębokości.

4.2 Baseline Synerise

Rozwiązanie baseline, zapropowane przez organizatorów, opiera się na agregacji cech (feature aggregation) i wykorzystuje praktyki inżynierii cech stosowane w rzeczywistych rozwiązaniach z dziedziny modelowania behawioralnego.

Podejście konstruuje dwa główne typy cech: statystyczne i zapytaniowe (query features). Cechy statystyczne reprezentują kategorie, ceny i produkty, którymi interesowali się użytkownicy, odpowiadając liczbie przeszłych interakcji dla każdego użytkownika w określonych oknach czasowych, na przykład "ile razy użytkownik kupił określony produkt X w ciągu ostatnich 30 dni". System rozważa różne okna czasowe takie jak 1 dzień, 1 tydzień i 1 miesiąc, agregując liczbę wydarzeń pogrupowanych według wartości kolumn. Z uwagi na znaczną liczbę kategorii, do obliczenia cech używany jest jedynie podzbiór wartości - kategorie, przedziały cenowe lub produkty są ograniczone do 10 najpopularniejszych wartości.

Cechy zapytaniowe działają inaczej, ponieważ typ wydarzenia search_query zawiera wektory całkowitoliczbowe otrzymane przez kwantyzację embeddingów tekstowych zapytań wyszukiwania użytkowników, więc dla każdego użytkownika konstruowane są nowe cechy poprzez obliczenie średniej z wektorów całkowitoliczbowych odpowiadających zapytaniom użytkownika.

Cały pipeline działa tak, że cechy użytkowników są wyodrębniane z surowych danych o wydarzeniach, najpierw obliczane osobno dla każdego typu wydarzenia, następnie łączone w bogate informacyjnie reprezentacje użytkowników i aplikowane do architektury ewaluacyjnej.

	Churn Prediction	Product Propensity	Category Propensity	Hidden 1	Hidden 2	Hidden 3
Baseline	0.6947	0.6985	0.6919	0.6579	0.7382	0.7207

Tabela 1: Wartości osiąganych wyników na zbiorze testowym dla rozwiązania baseline

4.3 Nieudane próby rozwiązania

Przez caly czas trwania projektu, wypróbowano wiele architektur sieci neuronowych, w szeregu eksperymentów. Opierały się one głównie na próbach ulepszenia opisanego powyżej baseline'u. Starano się dodać dodatkową gęstą sieć neuronową pomiędzy baseline input, a architekturę ewaluacyjną, tak by wytrenować ją podczas ewaluacji i później użyć do generowania embeddingów. Starano się także stworzyć autoenkoder, procesujący embeddingi stworzone w baseline, do formy bardziej przyjaznej dla architektury ewaluacyjnej. Te próby skutkowały jednak znacznym pogorszeniem wyników uzyskiwanych z samego baseline'u, więc nie przytaczamy tutaj ich wyników.

4.4 BERT4REC

Najlepszą znalezioną architekturą do zadanego problemu okazały się modele oparte na paradygmacie self-supervised learning, w szczególności rodzina BERT (Bidirectional encoder representations from transformers). [6]

BERT4Rec to model rekomendacji sekwencyjnej z rodziny BERT, który wykorzystuje dwukierunkową architekturę Transformer do modelowania sekwencji zachowań użytkowników. W przeciwieństwie do tradycyjnych jednokierunkowych modeli, które analizują historię użytkownika tylko z lewej strony, BERT4Rec używa dwukierunkowej samoatencji, pozwalając każdemu elementowi w historii użytkownika na czerpanie informacji zarówno z lewego, jak i prawego kontekstu. Model jest trenowany przy użyciu zadania Cloze, gdzie losowo maskowane elementy w sekwencji są przewidywane na podstawie ich lewego i prawego kontekstu, co zapobiega wyciekowi informacji i umożliwia efektywne trenowanie dwukierunkowego modelu. BERT4Rec składa się z warstw Transformer z mechanizmem wielogłowicowej samoatencji, co pozwala mu na bezpośrednie przechwytywanie zależności na dowolnych odległościach w sekwencji. Eksperymenty na czterech standardowych zbiorach danych pokazały, że model konsekwentnie przewyższa różne najnowocześniejsze modele sekwencyjne. [6]

4.4.1 Preprocessing danych

Model Bert4REC działa na sekwencjach zdarzeń, generując reprezentacje pozwalające z jak największą dokładnością predyktować kolejne ich elementy. Takie sekwencje musiały zostać wyekstraktowane osobno dla każdego typu zdarzenia dostępnego w datasecie. Żadnego processingu nie wymagały listy kupionych, dodanych do koszyka oraz usuniętych z koszyka przedmiotów, gdyż ich liczba nie przekraczała naszych możliwości sprzętowych. W przypadku page visit zdecydowano o usunięciu z datasetu identyfikatorów tych stron, które były odwiedzane mniej niż 5 razy. Wyszukiwania użytkowników zostaly umieszczone w oryginalnym datasecie w postaci 16-elementowych, skwantyfikowanych do wartości 0-255, embeddingów ich zawartości. Na szczęście, unikalnych wyszukiwań okazało się na tyle mało, że wystarczyło zastosować mapping embedding -> id każdej istniejącej kombinacji i na tym wytrenować model BERT4Rec.

4.4.2 Model oparty jedynie na akcjach kupna

Pierwsze eksperymenty z modelem BERT4Rec polegały na trenowaniu jednego modelu, dla jednego typu akcji, kupna produktów. Empirycznie dobrano najlepsze hiperparametry:

Parametr	Wartość
embedding_dim	256
projection_dim	256
hidden_dim	256
embedding_dropout	0.2
encoder_dropout	0.2
num_layers	6
num_heads	8
max_seq_len	30

Tabela 2: Hiperparametry BERT4Rec dla akcji kupna

Osiągnięte wyniki zostały przedstawione w tabeli 3. Są one nadal nieco gorsze od baseline'u, ale stanowiły w naszej opinii dobry prognostyk na przyszłość, przed dodaniem innym typów akcji do modelu.

	Churn Prediction	Product Propensity	Category Propensity	Hidden 1	Hidden 2	Hidden 3
Bert buys	0.649	0.6938	0.6578	0.6207	0.7167	0.7048

Tabela 3: Wartości osiąganych wyników na zbiorze testowym dla rozwiązania opartego na modelu BERT4Rec i akcjach kupna

4.4.3 Model oparty na wszystkich typach zachowań

Naturalnym ulepszeniem dla modelu wytrenowanego na jednym typie akcji było rozszerzenie go o inne aktywności. Oczywiście, model Bert4REC przyjmuje bazowo tylko jeden typ sekwencji, a my nie znamy powiązania pomiędzy wyszukiwaniami, a identyfikatorami przedmiotów. Ze względu na dostępność mocy obliczeniowej, próby augmentacji modelu o informację o typie akcji i połączeniu wszystkich interakcji w 1 dataset nie udały się, dlatego wytrenowano 5 osobnych modeli dla każdego typu zachowania.

Testowano różne sposoby na łączenie embeddingów różnych typów zachowań, między innymi liczenie średniej oraz konkatenację, która ostatecznie została wybrana, ale dobór tej metody nie miał dużego wpływu na ostateczne wyniki.

Hiperparametry modeli nie różniły się względem tabeli 3, z wyjątkiem maksymalnej długości sekwencji, która była dobierana empirycznie dla każdego typu zachowania:

Maksymalna długość sekwencji	Wartość
kupna produktów	30
dodania do koszyka	64
usunięcia z koszyka	48
wyszukiwania	128
odwiedzenia stron	128

Tabela 4: Maksymalne długości sekwencji BERT4Rec dla każdego typu akcji

Osiągnięte wyniki były bardziej niż zadowalające:

	Churn Prediction	Product Propensity	Category Propensity	Hidden 1	Hidden 2	Hidden 3
Bert all	0.6851	0.7677	0.723	0.7282	0.7284	0.7859

Tabela 5: Wartości osiąganych wyników na zbiorze testowym dla rozwiązania opartego na modelu BERT4Rec i wszystkich typach akcji

5 Wyniki

Udało nam się znaleźć bardzo dobre rozwiązanie do skomplikowanego problemu systemów rekomendacyjnych - tworzenia uniwersalnych reprezentacji użytkowników, bazując na wykonywanych przez nich akcjach w aplikacji internetowej.

Proponujemy uniwersalne rozwiązanie uczenia zespołowego oparte na trenowaniu wielu modeli BERT4Rec do różnych typów akcji możliwych do wykonania przez użytkownika. Okazało się ono lepsze od, stworzonego na podstawie starannie dobranych danych statystycznych, baseline'u organizatorów oraz rozwiązań większości konkurentów. Nasz model pozwolił nam zająć około **25.** miejsce na prawie **400** zarejestrowanych zespołów. Niestety, na ostateczne wyniki musimy jeszcze poczekać, ze względu na wyciek danych ze strony organizatorów.

Porównanie osiągniętych przez nasze najlepsze modele wyników oraz baseline'u znajduje się w tabeli 6.

	Churn Prediction	Product Propensity	Category Propensity	Hidden 1	Hidden 2	Hidden 3
Baseline	0.6947	0.6985	0.6919	0.6579	0.7382	0.7207
Bert buys	0.649	0.6938	0.6578	0.6207	0.7167	0.7048
Bert all	0.6851	0.7677	0.723	0.7282	0.7284	0.7859

Tabela 6: Wartości osiąganych wyników na zbiorze testowym dla rozwiązania opartego na modelu BERT4Rec i wszystkich typach akcji

6 Wnioski

Udział w konkursie i stworzenie rozwiązania na postawiony przez organizatorów konkursu problem pozwoliły nam lepiej zrozumieć, jak działają systemy rekomendacyjne i w jaki sposób można wykorzystać do tego metody sztucznej inteligencji.

Warto również podkreślić fakt, że modele BERT4Rec są skuteczne w zadaniach predykcyjnych i pozwalają na dobre uogólnienie rozwiązania. Tego typu podejście może znacznie zwiększyć skuteczność predykcji w systemach rekomendacyjnych, tworząc uniwersalne profile użytkowników. Dalszy rozwój i ulepszanie systemów rekomendacyjnych z pomocą sztucznej inteligencji może być skutecznym narzędziem do optymalizacji i analizy działań współczesnych przedsiębiorstw.

Literatura

[1] S. Zhang, L. Yao, A. Sun, and Y. Tay, "Deep learning based recommender system: A survey and new perspectives," *ACM Computing Surveys*, vol. 52, p. 1–38, Feb. 2019.

[2] Z. He, W. Liu, W. Guo, J. Qin, Y. Zhang, Y. Hu, and R. Tang, "A survey on user behavior modeling in recommender systems," *arXiv preprint arXiv:2302.11087*, 2023.

[3] F. Sun, J. Liu, J. Wu, C. Pei, X. Lin, W. Ou, and P. Jiang, "Bert4rec: Sequential recommendation with bidirectional encoder representations from transformer," 2019.

[4] P. Emerson, "The original borda count and partial voting," *Social Choice and Welfare*, vol. 40, no. 2, pp. 353–358, 2013.

[5] T. H. Gregor Bachmann, Sotiris Anagnostidis, "Scaling mlps: A tale of inductive bias," *arXiv:2306.13575v3*, 2023.

[6] F. Sun, J. Liu, J. Wu, C. Pei, X. Lin, W. Ou, and P. Jiang, "Bert4rec: Sequential recommendation with bidirectional encoder representations from transformer," *arXiv preprint arXiv:1904.06690*, April 2019.