PRACE NAUKOWE UNIWERSYTETU EKONOMICZNEGO WE WROCŁAWIU
RESEARCH PAPERS OF WROCŁAW UNIVERSITY OF ECONOMICS nr 526 • 2018

ISSN 1899-3192
e-ISSN 2392-0041

Maciej Pondel
Uniwersytet Ekonomiczny we Wrocławiu, Unity S.A., Wrocław
e-mail: maciej.pondel@ue.wroc.pl

Jerzy Korczak
Międzynarodowa Wyższa Szkoła Logistyki i Transportu, Wrocław
e-mail: jerzy.korczak@ue.wroc.pl

WYBRANE ALGORYTMY *MACHINE LEARNING* W MARKETINGU

SELECTED MACHINE LEARNING ALGORITHMS IN MARKETING

DOI: 10.15611/pn.2018.526.02
JEL Classification: C55

Streszczenie: W artykule zaprezentowano wybrane metody i narzędzia *machine learning*, wykorzystujące najnowsze technologie informacyjno-komunikacyjne, które mogą być używane przez podmioty prowadzące działalność handlową w modelu *omnichannel* w celu osiągnięcia przewagi konkurencyjnej na trudnym i turbulentnym rynku internetowym. Opisane zostaną metody pracy użytkownika platformy Real-Time Omnichannel Marketing (menedżera, analityka marketingu), poszukującego nietrywialnej, nowej i użytecznej wiedzy, którą będzie mógł wykorzystać w procesie podejmowania decyzji. Pokazano kilka przykładów zastosowań algorytmów *machine learning*. W szczególności opisano metody poszukiwania wzorców zachowań klienta, które zilustrowano eksperymentami wykonanymi na rzeczywistych danych. Pozyskana wiedza może być użyta w sposób automatyczny w procesach komunikacji z klientem m.in. do zwiększenia szansy zakupu, polepszenia satysfakcji klienta, zmniejszenia ryzyka odejścia klienta czy optymalizacji marży na produkcie.

Słowa kluczowe: eksploracja danych marketingowych, *machine learning*, Hadoop, MLlib, reguły asocjacyjne, Spark.

Summary: The paper presents selected methods and tools of machine learning using the latest information and communication technology, that can be used by companies performing commercial activities in the omnichannel model in order to achieve a competitive advantage in a difficult and turbulent Internet market. Using the Real-Time Omnichannel Marketing (RTOM) analytic platform, we will show several examples of the use of machine learning algorithms. Operations of the RTOM platform user (manager, marketing analyst) who looks for a non-trivial, new and useful knowledge useful in the decision-making process will be presented, in particular customer behaviour patterns will be exemplified by experiments

performed on real data. Acquired knowledge can be used automatically in the processes of communication with the client in order to increase the chance of buying, improve customer satisfaction, reduce the risk of customer leaving or to optimize the margin on the product.

Keywords: marketing data exploration, Machine Learning, Hadoop, MLlib, association rules, Spark.

1. Wstęp

Eksploracja danych marketingowych jest procesem, przy pomocy którego możemy lepiej poznać nie tylko preferencje klientów i ich zachowania, ale także prognozować przyszłe transakcje, indywidualizować oferty handlowe, organizować kampanie marketingowe. Dzięki temu możemy zwiększyć lojalność klientów, co docelowo powinno wpłynąć pozytywnie na rentowność prowadzonych działań handlowych. Najogólniej mówiąc, zadaniem eksploracji jest analiza danych i procesów w celu lepszego ich poznania, zrozumienia i wykorzystania w procesach podejmowania decyzji. Współczesne systemy eksploracji wykorzystują szerokomobilne technologie informacyjno-komunikacyjne, technologie Web, metody wyszukiwania informacji, techniki geolokalizacji, przetwarzania sygnałów i bioinformatyki.

Problematyka analizy i eksploracji dużych baz marketingowych jest przedmiotem wielu badań i projektów aplikacyjnych [Gordon, Linoff, Berry, 2011; Han, Pei, Kamber 2012; Korczak, Pondel 2017; Morzy 2013; Pawełoszek, Korczak 2017; Weichbroth 2009; Pondel 2015]. Celem artykułu jest z jednej strony pokazanie podejścia do eksploracji danych transakcyjnych sklepu internetowego, a z drugiej zaprezentowanie możliwości wykorzystania algorytmów m.in. analizy koszyka zakupów, wyszukiwania wzorców zachowania klientów w czasie rzeczywistym oraz algorytmów klasyfikacji. Problemy te zostaną przedstawione na rzeczywistych danych transakcyjnych, odnoszących się do wielokanałowej sprzedaży (*omnichannel marketing*), zwanej dalej RTOM. Platforma gromadzi głównie w czasie rzeczywistym i przetwarza je w ogromnych ilościach, przy dużej heterogeniczności ich źródeł, formatów, wolumenu i intensywności napływu. Użytkownik platformy (menedżer, analityk marketingu) oczekuje nietrywialnej, nowej i użytecznej wiedzy, którą będzie mógł wykorzystać w procesie podejmowania decyzji. Pozyskana wiedza z zebranych danych powinna być użyta w sposób automatyczny w procesach komunikacji z klientem tak, aby zwiększyć prawdopodobieństwo zakupu, satysfakcję klienta, marżę na produkcie, zmniejszyć ryzyko odejścia klienta i wiele innych. Dzięki modelom predykcyjnym, zbudowanym w oparciu o metody *machine learning*, będziemy w stanie podejmować działania marketingowe w czasie rzeczywistym. Proces analizy i eksploracji danych został przeprowadzony według metodyki opisanej w pracy [Pondel, Korczak 2017].

Struktura artykułu jest następująca. W następnym punkcie zaprezentowano źródła danych poddanych eksploracji oraz zastosowane technologie. Dane dotyczyły

transakcji zakupów klientów sklepu internetowego, w szczególności informacji o koszyku zakupów, sekwencji transakcji klientów i zwrotów zakupionych towarów. W części eksperymentalnej artykułu przedstawiono cztery przykłady zastosowań algorytmów eksploracji. Pierwsze dwa dotyczą poszukiwania tzw. koszyka zakupów i wzorców zachowań klientów z wykorzystaniem algorytmów FPGtrowth i PrefixSpan. W ostatnich częściach opisano dwa następne eksperymenty budowy modeli predykcyjnych korzystających z indukcyjnych drzew decyzyjnych. Rozwiązywane problemy biznesowe dotyczyły prognozy zwrotu zakupionych produktów i strategii wyprzedaży. Wszystkie przedstawione eksperymenty zostały przeprowadzone i ocenione na rzeczywistych danych marketingowych.

2. Źródła danych transakcyjnych

Głównym źródłem informacji są dane pochodzące z systemu eCommerce. Format i struktura danych została poddana niewielkim zmianom denormalizacyjnym w stosunku do struktury bazy danych sklepu internetowego. Jednym z typowych zadań eksploracji jest wykrycie najczęściej kupowanych grup produktów przez klientów sklepu internetowego oraz określenie reguł asocjacyjnych opisujących relacje między często kupowanymi razem produktami. Oczekuje się, że znalezione wzorce zakupów zostaną wykorzystane do opracowania strategii sprzedaży, akcji promocyjnych, organizacji stron internetowych czy doskonalenia katalogu oferowanych produktów. W drugiej grupie eksperymentów skoncentrowano się na tworzeniu modeli predykcyjnych związanych ze zwrotami zakupionych towarów i strategią wyprzedaży.

W sklepie internetowym dane transakcyjne są przechowywane w relacyjnej bazie danych PostgreSQL. W przedstawionych w artykule eksperymentach ograniczono zainteresowania do danych z systemu ERP o fakturach korygujących, wynikających z faktu dokonania zwrotu przez klienta. Należy zaznaczyć, że platforma RTOM kopiuje dane transakcyjne do swojej własnej struktury opartej na technologii Apache Hadoop. Dane składowane są w systemie plików HDFS, a dostęp do nich zapewniają mechanizmy hurtowni danych Hive oraz Impala. Wybrane mechanizmy składowania danych są wiodącymi technologiami w zastosowaniach Big Data opartych na przetwarzaniu ogromnych danych pochodzących z heterogenicznych źródeł.

3. Analiza koszyka zakupów i wzorce zachowań klienta

Platforma RTOM zawiera wiele metod i algorytmów eksploracji danych marketingowych. Jednym z bardziej atrakcyjnych biznesowo jest moduł wyszukiwania reguł asocjacyjnych. Umożliwia on przeprowadzenie dwóch podstawowych badań, mianowicie analizy koszyka sklepowego oraz wyszukiwania wzorców sekwencji zakupów. Oba te zadania sprowadzają się do zbudowania modelu, który pozwoli lepiej zrozumieć zachowania klienta oraz zaproponować efektywne rekomendacje zakupowe (por. [Pondel, Pondel 2011; Chorianopoulos 2016; Schutt, O'Neil 2013]).

Istnieje wiele algorytmów wyszukiwania reguł asocjacyjnych [Kulurkar, Badole 2016; Han i in. 2004; Győrödi i in. 2004]. Algorytmy wyszukiwania reguł asocjacyjnych, takie jak Apriori, Charm, FP-Growth i inne, różnią się złożonością obliczeniową, a zatem zapotrzebowaniem na zasoby oraz czasem wykonania. Ich sposób działania jest deterministyczny, a więc uzyskane rezultaty będą takie same. W artykule ograniczono się ze względów implementacyjnych do algorytmów dostępnych w bibliotece MLlib, zatem w celu przeprowadzenia analizy koszyka zakupów w projekcie wykorzystaliśmy popularny algorytm FP-Growth [Morzy 2013; Han, Fu 1999; Setia, Jyoti 2013].

W algorytmie proces wykrywania zbiorów częstych jest realizowany w dwóch krokach:

1) kompresja bazy danych D do *FP-drzewa*, w którym baza danych transakcji D jest kompresowana i przekształcana do postaci *FP-drzewa*,

2) eksploracja *FP-drzewa*, w którym *FP-drzewo* jest analizowane w celu znalezienia zbiorów częstych.

Progowe częstości są określone przez analityka lub menedżera; m.in. podstawowy parametr *minimalne wsparcie* określa minimalną liczbę transakcji zawierających dany produkt, tym samym zakreśla obszar zainteresowań transakcjami klientów. Na ogół w zależności od celów marketingowych wartość tego parametru wyznacza się eksperymentalnie. Po określeniu produktów często kupowanych można wygenerować listę reguł asocjacyjnych, spełniających zadane poziomy minimalnego wsparcia oraz wskaźnika ufności. Tym samym wskazane zostaną określone prawdopodobieństwa produktów kupowane wspólnie. Na tej podstawie można zdefiniować rekomendacje produktowe. Działanie reguł oraz ich implementacja w platformie RTOM przy użyciu biblioteki MLlib oraz algorytmów FPGrowht oraz PrefixSpan zostały opisane w artykule [Pondel, Korczak 2017].

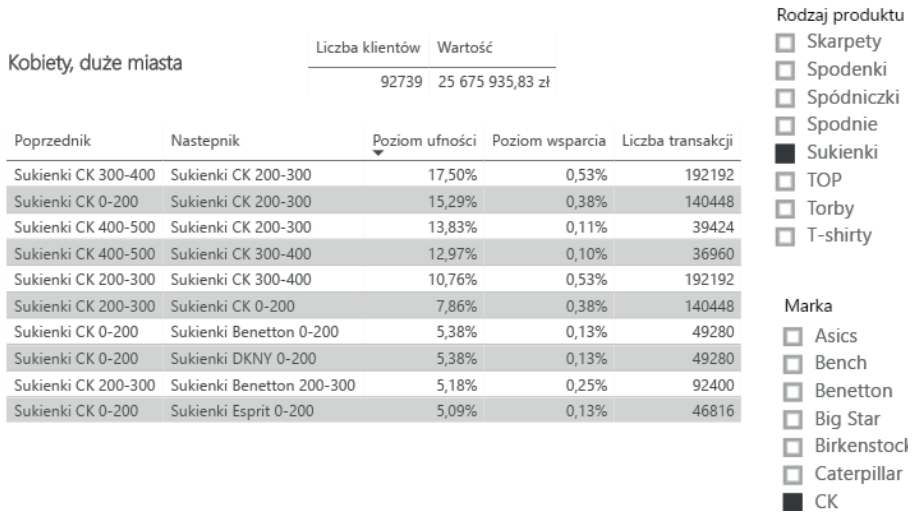
W przykładach posłużymy się danymi z branży odzieżowej. Do wyznaczania asocjacji podzielono produkty na segmenty obejmujące np. kategorię (np. buty, koszulki, spodnie), osobę, dla której produkt jest przeznaczony (damskie, dziecięce, męskie), poziom cenowy (np. do 100 zł, 100-300 zł, powyżej 300 zł), markę, kolor, poziom rabatu i wiele innych. Na rys. 1 pokazano podstawowe informacje do utworzenia reguł asocjacyjnych opisujących prawdopodobieństwo dokonania zakupu produktu. Po wyborze przez analityka produktów (z prawej strony ekranu) platforma generuje reguły w postaci:

Jeśli Poprzednik, to Następnik z danym poziomem ufności[1] *i wskaźnikiem wsparcia*[2], gdzie *Następnikiem* jest zakupiony produkt w przypadku zdecydowania się przez klienta na produkt opisany jako *Poprzednik*.

[1] Poziom ufności (*confidence*) określa, w jakim stopniu wykryta reguła asocjacyjna jest „pewna"; obliczany jest jako stosunek liczby transakcji wspierających regułę do wszystkich transakcji zakupu poprzednika.

[2] Wskaźnik wsparcia (*support rate*) określa liczbę transakcji klientów, którzy kupują zgodnie z daną regułą; obliczany jako stosunek liczby transakcji wspierających regułę do wszystkich transakcji.

W podanym raporcie zaprezentowano reguły dotyczące kobiet dokonujących zakupów w dużych miastach. Przykładowo, jeśli kobieta dokonała zakupu Sukienki CK kosztującej pomiędzy 300 a 400 zł, to w 17,5% dokonała zakupu drugiej sukienki tej samej marki, jednak z segmentu cenowego 200-300 zł. Reguła ta ma zastosowanie w 0,53% wszystkich transakcji, co przekłada się na ponad 192 tys. transakcji. Oprócz podanych reguł, RTOM informuje analityka o wielkości populacji klientów oraz liczbie dokonanych transakcji i ich wartości.



Rys. 1. Prezentacja reguł asocjacyjnych w platformie RTOM

Źródło: opracowanie własne.

Dla pozostałych segmentów klientów obliczono odrębny zestaw reguł. Możliwość interaktywnego wskazania produktów na platformie RTOM ułatwia analitykowi przeprowadzenie *à la carte* analizy transakcji klientów pod kątem koszyka zakupów.

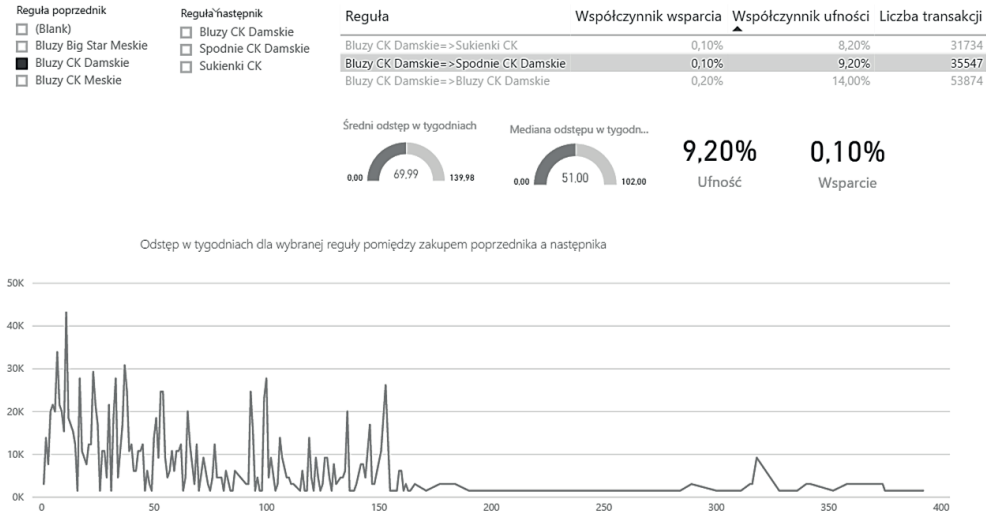
Dzięki wygenerowaniu tego typu reguł:

- możemy zaproponować klientowi, który dokonał zakupu, produkt, którym może również być zainteresowany; taką propozycję klient może otrzymać w świecie wirtualnym jako inteligentną rekomendację bądź w tradycyjnym sklepie od sprzedawcy,
- możemy przeprowadzić kampanie marketingowe celowane do poszczególnych segmentów klientów (a nawet klientów indywidualnych).

Algorytm poszukiwania wzorców zachowań jest nieco inny i korzysta z dodatkowych danych. Skorzystaliśmy tu z algorytmu PrefixSpan dostępnego w bibliotece org.apache.spark.mllib.fpm.PrefixSpan [Han, Pei, Kamber 2012; Morzy 2013; Pei i in. 2004]. Generuje on często pojawiające się sekwencje wraz z ich częstością.

Niestety, w przypadku tego algorytmu system nie zwraca reguł oraz nie oblicza wskaźników wsparcia oraz ufności (dlatego musiano dodatkowo dokonać implementacji tych funkcji). Model tworzony przy pomocy algorytmu PrefixSpan zasilony jest tablicą trójwymiarową, gdzie pierwszy wymiar stanowią poszczególni klienci, w ramach drugiego wymiaru podawane są wszystkie transakcje klientów. Trzeci wymiar to produkty znajdujące się na poszczególnych transakcjach. Inaczej mówiąc, do obliczeń konieczna jest nie tylko informacja o zakupionych produktach, ale też informacja o kliencie (np. jego identyfikator) i o dacie dokonania transakcji. W tym algorytmie poszukujemy zatem takich reguł sekwencyjnych, które wskażą kolejność, inaczej sekwencje, dokonanych zakupów przez klientów z podaniem odstępu pomiędzy zakupami.

Na rys. 2 pokazano prawdopodobieństwo zakupu następnika po zakupie poprzednika oraz rozkład czasu, po jakim klient dokonuje kolejnego zakupu. Dzięki temu można wyznaczyć, w którym momencie zaproponować klientowi zakup, aby był on najbardziej prawdopodobny. Przykładowo w przypadku wskazanej reguły wiadomo, że jeśli klient dokonał zakupu produktu z kategorii Bluza CK Damska, to w kolejnej transakcji w 9,2% dokona zakupu Spodni CK Damskich. Możemy także odczytać, jaki jest średni odstęp w tygodniach pomiędzy tymi zamówieniami. Jednak ciekawsza jest analiza wykresu wskazującego maksima lokalne odległości pomiędzy zakupami, które przypadają w tym przykładzie na tygodnie 11, 25, 53, po dokonaniu pierwotnego zakupu. Jak widać, wykres nie ma regularnego charakteru (zwłaszcza na początku), co oznacza, że odległości pomiędzy zakupem poprzednika a następnika (wyrażone w tygodniach) w dużym zbiorze transakcji są różnorodne.



Rys. 2. Wizualizacja reguł sekwencyjnych w platformie RTOM

Źródło: opracowanie własne.

Dzięki automatycznemu wygenerowaniu wzorców sekwencyjnych menedżer marketingu, planując kampanię, nie musi analizować każdego wzorca zachowania klienta oraz definiować dla niego odpowiedniej oferty. Dla przykładu, nie musimy się zastanawiać, czy osobom kupującym na wiosnę buciki dziecięce zaproponować wraz z nadejściem zimy buciki zimowe. Informacja taka jest podana algorytmicznie. Platforma automatycznie generuje setki lub nawet tysiące reguł (hipotez) i poddaje je automatycznej ocenie, a następnie prezentuje, a nawet sama wprowadzi je w życie poprzez automatyczne skierowanie do odpowiednich grup właściwych komunikatów marketingowych.

W dalszej części artykułu skupiono się na następnych dwóch eksperymentach przeprowadzonych z wykorzystaniem klasyfikatorów binarnych oraz dokonano próby oceny ekonomicznej uzyskanych wyników. W cele zilustrowania technologii Big Data wybrano dwa następne problemy aplikacyjne, mianowicie: predykcję zwrotów zakupionych produktów i predykcję skorzystania z wyprzedaży. Wybór tych przykładów był podyktowany, z jednej strony, rangą zagadnienia w kontekście rentowności sklepu internetowego, z drugiej złożonością problemu.

4. Predykcja zwrotów zakupionych produktów

Umożliwienie klientom dokonywania zwrotów zakupionych produktów jest istotnym elementem, budującym przewagę konkurencyjną podmiotu handlowego. Może być traktowane jako benefit dla klienta, stanowiący argument za dokonaniem zakupów, ponieważ klienci cenią sobie możliwość dokonania zwrotu. Jednakże z logistycznej i finansowej perspektywy obsługa zwrotu jest bardzo kosztowna. Mało restrykcyjna polityka zwrotów generuje zatem koszty operacyjne, obniża rentowność, a także niejako zachęca część klientów do dokonywania nadużyć [Wood 2001]. W USA w branży odzieżowej wskaźnik zwrotów zakupów z *e-commerce* waha się pomiędzy 20% a 40% [Ratcliff 2014; Stacey 2016]. Podmioty działające w modelach *e-commerce* oraz *omnichannel commerce* dążą zatem do redukcji liczby zwrotów [Walsh i in. 2014; Hjort 2013; Kim, Larose 2003], posługując się różnymi środkami, takimi jak: narzędzia dodatkowe redukujące przyczyny zwrotów, ograniczenia w regulaminie czyniące politykę zwrotów bardziej restrykcyjną lub dostarczenie klientowi wartości dodatkowej w przypadku dokonania zamówienia. Osiągnięcie tych celów wymaga posiadania rozwiązań, które:

1) umożliwią w czasie rzeczywistym (podczas dokonywania zakupu) oszacowanie prawdopodobieństwa, że zamówienie zakończy się zwrotem, celem podjęcia decyzji o zaoferowaniu klientowi wartości dodatkowej w przypadku braku zwrotu;

2) pozwolą wyjaśnić przyczyny oraz wzorce zachowań prowadzące do dokonania zwrotu przez klienta; innymi słowy, pokaże osobie decydującej o polityce sklepu typowe sytuacje, w których występują zwroty, celem wykluczenia np. z regulaminu sytuacji, w których może dochodzić do nadużyć.

W zbiorze danych, który analizowano, pochodzącym z polskiej sieci sprzedaży działającej w branży odzieżowej, zwroty stanowią również istotną część biznesu. Ze względu na tajemnicę przedsiębiorstwa w opisie rezultatów eksperymentu nie podano konkretnych danych, lecz jedynie skupiono się na sposobie rozwiązania problemu.

Celem eksperymentu jest opracowanie modelu wskazującego dla każdej nowej transakcji prawdopodobieństwo dokonania zwrotu przez klienta. Drugim niezmiernie ważnym zadaniem jest zbudowanie zwizualizowanego modelu wskazującego osobom podejmującym decyzje dotyczące polityki firmy przypadków, w których produkty są najczęściej zwracane.

Aby model został uznany za spełniający pierwszy warunek, powinno się oszacować koszty błędnie podjętych decyzji oraz korzyści wynikające z decyzji prawidłowych. Jeśli do kosztów dodamy koszty związane z wdrożeniem modelu, to będziemy mogli uznać, że model spełni swoje zadanie, jeśli rezultaty ekonomiczne będą pozytywne. W dalszej części artykułu podano krótkie podsumowanie finansowe działania modelu i porównanie z sytuacją, gdy nie korzystamy z modelu i ponosimy faktyczne koszty obsługi zwrotów. W przypadku drugiego warunku, aby model spełnił swoje zadanie, musi być zrozumiały dla menedżera podejmującego decyzje o polityce zwrotów. W niniejszym artykule skoncentrowano się na rozwiązaniu pierwszego problemu.

Zbudowanie klasyfikatora wymaga wcześniejszego przygotowania zbioru o dotychczasowych zamówieniach klientów. Zbiór historyczny zamówień uzupełniono zmienną celu przyjmującą wartość 0 – oznaczającą, że pozycja zamówienia (czyli produkt) nie została zwrócona, lub 1 – mówiącą, że została zwrócona. Algorytm klasyfikacyjny wymaga, aby wszystkie zmienne w zbiorze były liczbami.

Każda pozycja zamówienia została opisana 29 zmiennymi objaśniającymi w następujących zakresach:

1. Szczegóły danej pozycji zamówienia – sprowadzające się do podsumowania liczby produktów w podziale na:
   i) kategorię (np. buty, spodnie itp.),
   ii) odbiorcę (damskie, męskie, dziecięce),
   iii) charakterystyki pozycji zamówienia (średni rabat, średnią obniżkę w stosunku do ceny katalogowej, liczbę produktów kupionych w promocji, wartość zamówienia); zmienna celu dotyczy właśnie tej pozycji zamówienia.
2. Profil demograficzny klienta – opisujący płeć oraz miejsce zamieszkania klienta. W przypadku miejsca zamieszkania oparto się na wielkości miasta, jego obszarze oraz gęstości zaludnienia. Dodatkowo dwie zmienne opisują położenie województwa na mapie Polski jako odległość od zachodniej granicy oraz odległość od południowej granicy.
3. Historię poprzednich zamówień klienta, opisującą zamówienia dokonane przez klienta, które poprzedzały zamówienie będące bazowym. Podobnie jak w przypadku bazowego zamówienia podano tu liczbę wcześniejszych zamówień klienta, liczbę

produktów zakupionych uprzednio w podziale na kategorie oraz przeznaczenie, staż klienta, częstotliwość dokonywania zakupów, średnią wartość koszyka.

Zbiór historyczny zamówień klientów wymagał przeprowadzenia pewnych prac przygotowawczych i transformacji, m.in. standaryzacji. Celem standaryzacji zmiennych było przetłumaczenie bezwzględnej wartości zmiennej na wartość mówiącą o tym, w jakim stopniu dana obserwacja jest odstająca od średniej obserwacji. Standaryzacji zmiennych dokonano przy użyciu biblioteki Scikit-learn. Po tym kroku zbiór wejściowy został podzielony na dwie części (w sposób losowy):

- zbiór, na którym klasyfikator został nauczony – stanowiący 75% zbioru wejściowego, oraz
- zbiór testowy, na podstawie którego klasyfikator będzie oceniany – stanowiący pozostałe 25% zbioru wejściowego.

W projekcie zbudowano wiele modeli klasyfikacyjnych. Z uwagi na ograniczone ramy artykułu w dalszej części opisano proces budowy klasyfikatora opartego na algorytmie indukcyjnych drzew decyzyjnych. Wybór drzewa decyzyjnego wynikał z jego stosunkowo wysokiej skuteczności w dotychczasowych badaniach oraz łatwej do zrozumienia graficznej reprezentacji, wyjaśniającej, jakie parametry i w jakim stopniu wpływają na fakt dokonania zwrotu przez konsumenta. W implementacji klasyfikatora posłużono się algorytmem z biblioteki sklearn.tree.DecisionTreeClassifier.

Przy budowaniu drzewa konieczne jest skonfigurowanie jego parametrów. Niektóre z parametrów przyjmują jedną z dwóch możliwości, a inne wiele wartości liczbowych. W przypadku gdy liczba kombinacji możliwych wartości parametrów klasyfikatora jest duża, wówczas należy dobrać możliwie najskuteczniejszą kombinację wartości parametrów. W celu rozwiązania tego złożonego problemu wykorzystano algorytm GridSearchCV, który umożliwia równoległe wykonywanie zadań przy jednoczesnym użyciu wielu serwerów i ich procesorów. W tym celu użyto silnika Spark, który obecnie jest wiodącym mechanizmem obliczeniowym wykorzystywanym w zadaniach Big Data. Algorytm GridSearchCV[3] dla każdej możliwej kombinacji parametrów przygotowuje odrębny model, następnie dokonuje jego oceny w oparciu o wybraną miarę. Model o najwyższej wartości danej miary uznaje za najlepszy i sugeruje użycie go w dalszych badaniach. Sposób działania tego algorytmu ilustruje rys. 3. W ocenie posłużono się trzema miarami skuteczności modeli klasyfikacji, mianowicie:

- wskaźnikiem trafności klasyfikacji (*accuracy rate*), wyrażającym stosunek poprawnych klasyfikacji/predykcji do wszystkich przypadków);
- wskaźnikiem dokładności (*precision*) liczony jako TP[4]/(TP +FP[5]), oznaczający biznesowo procent prawidłowo przewidzianych zwrotów w stosunku do wszyst-

[3] Opisany na stronie: https://databricks.com/blog/2016/02/08/auto-scaling-scikit-learn-with-apache-spark.html (3.11.2017).

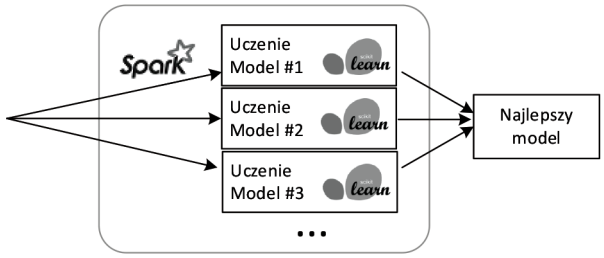
[4] TP – liczba przypadków prawdziwie pozytywnych (*true positive*), czyli prawidłowo sklasyfikowanych przez model jako pozytywne.

[5] FP – liczba przypadków fałszywie pozytywnych (*false positive*), czyli nieprawidłowo sklasyfikowanych przez model jako pozytywne (naprawdę były negatywne).

kich, dla których klasyfikator przewidział zwrot; błąd w tym przypadku oznacza, że klasyfikator wskazał, że dana osoba zwróci, a faktycznie to nie nastąpiło;

- wskaźnikiem wrażliwości (*recall* lub *true positive rate*), liczonym jako TP/(TP+ FN[6]), który w tym eksperymencie oznacza procent prawidłowo przewidzianych zwrotów w stosunku do wszystkich zwróconych zamówień; w tym przypadku błąd to pominięcie w przewidywaniu klienta, który dokonał zwrotu.

s_avg_markdown_percent	s_promotions_count	s_order_value	s_female_count
-0.4320237733759281	0.6096557522720328	0.013181399313738 3007	0.036125108820623 92
-1.4651409827842 24	-0.5078106606904077	0.955137442904 3412	0.738530831880946 3
-1.4585159562777088	1.7271221652344735	-0.3102654536113343	-0.198010132199483 52
-0.8889509781543572	1.7271221652344735	-0.1232252958248 3638	-0.198010132199483 52
-0.8889353069384451	1.7271221652344735	-0.2099901138835 2745	-0.198010132199483 52
-0.9411376603713936	1.7271221652344735	0.7680972851178 63	0.738530831880946 3
-0.4133533048468137	-0.5078106606904077	-0.2843599579338 169	0.738530831880946 3
-0.7136191953244554	-0.5078106606904077	0.211439002401 4464	-0.198010132199483 52
0.6559764264000064	-0.5078106606904077	-0.4083096980176326	-0.198010132199483 52
1.9330376088846477	1.7271221652344735	-0.5322504581014485	0.738530831880946 3
-0.0013752458974194107	1.7271221652344735	1.5129112732815122	0.738530831880946 3
0.4067935760286006	-0.5078106606904077	-0.7180600984870883	0.738530831880946 3
0.5069441606186595	1.7271221652344735	-0.4577656443110751 4	-0.198010132199483 52
0.9227075824966846	1.7271221652344735	1.1193708485153 97	0.738530831880946 3



Rys. 3. Sposób działania algorytmu GridSearchCV

Źródło: https://databricks.com/blog/2016/02/08/auto-scaling-scikit-learn-with-apache-spark.html (3.11.2017).

W finalnej fazie badania liczba parametrów drzewa została ograniczona z 13 do 5 parametrów, mianowicie: kryterium podziału, maksymalna głębokość drzewa, maksymalna liczba zmiennych branych przy podziale drzewa, maksymalna liczba liści drzewa, minimalna liczba wierszy dla utworzenia liścia. Przy wyborze parametrów kierowano się rekomendacjami autorów algorytmu oraz własnymi obserwacjami wpływu modyfikacji danego parametru na ocenę drzewa. Przy ustawieniu parametru decydującego o wyborze drzewa jako wskaźniku trafności na zbiorze testowym osiągnięto wyniki, które zaprezentowano w tabeli 1. Ostateczny dobór parametrów będzie obszarem kolejnych badań.

Warto podkreślić fakt, że najgorsze w tym eksperymencie drzewo osiągnęło trafność 83,7%, bardzo zbliżoną do najlepszego. Należy zaznaczyć, że w większości zastosowań biznesowych, zwłaszcza przy dużym niezrównoważeniu liczebności klas, posługiwanie się ogólnym wskaźnikiem trafności klasyfikacji (*accuracy rate*), wyrażającym stosunek poprawnych klasyfikacji/predykcji do wszystkich przypadków, nie jest dobrą metryką. Zakładając, że zwroty stanowią 10% wszystkich zamówień, to klasyfikator, który zawsze będzie przywidywał brak zwrotu, osiągnie trafność na poziomie 0,9, a więc będzie lepszy od tych obliczonych przez nas. Stąd większą wagę należałoby przyłożyć do innych miar.

[6] FN – liczba przypadków fałszywie negatywnych (*false negative*), czyli nieprawidłowo sklasyfikowanych przez model jako negatywne (naprawdę były pozytywne).

Tabela 1. Lista 10 najlepszych drzew przy wybraniu wskaźnika trafności jako parametru oceny

Wartość wskaźnika na zbiorze testowym	Wartość wskaźnika na zbiorze testowym	Kryterium podziału	Maksymalna głębokość	Maksymalna liczba zmiennych przy podziale	Maksymalna liczba liści	Minimalna liczba wierszy na liść	Pozycja modelu w rankingu
0,838299	0,838510	gini	11	log2	30	220	1
0,838137	0,838215	gini	10	log2	30	220	2
0,838120	0,838169	entropy	11	log2	30	220	3
0,838075	0,838165	entropy	10	log2	30	250	4
0,838058	0,838242	entropy	10	log2	30	250	5
0,838032	0,838113	entropy	10	log2	30	220	6
0,838013	0,838158	gini	11	log2	30	220	7
0,837990	0,838014	gini	10	log2	30	250	8
0,837964	0,838083	gini	11	log2	30	250	9
0,837896	0,837930	entropy	10	log2	30	220	10

Źródło: opracowanie własne przy użyciu GridSearchCV.

W tabeli 2 podano ocenę drzew decyzyjnych przy użyciu wskaźników dokładności oraz wrażliwości.

Tabela 2. Wartości trzech wskaźników w przypadku najlepszego i najgorszego klasyfikatora

Ocena	*Accuracy*	*Precision*	*Recall*
Najlepszy	83,8%	60,4%	61,0%
Najgorszy	83,7%	33,0%	38,0%

Źródło: opracowanie własne.

Dla klasyfikatora, który osiągnął najwyższą wartość wskaźnika wrażliwości, zaprezentowano także krzywą ROC (rys. 4). Krzywa ROC prezentuje wartości *recall* (*true positive rate*) oraz wskaźnika *false positive rate* dla różnych poziomów progów akceptacji. Lewy górny róg macierzy oznacza sytuację idealną, w której klasyfikator nie popełnia żadnych błędów. Przyjęta inna miara AUC (*area under ROC curve*, czyli pole powierzchni pod krzywą) wskazuje nam odległość naszego klasyfikatora od sytuacji idealnej.

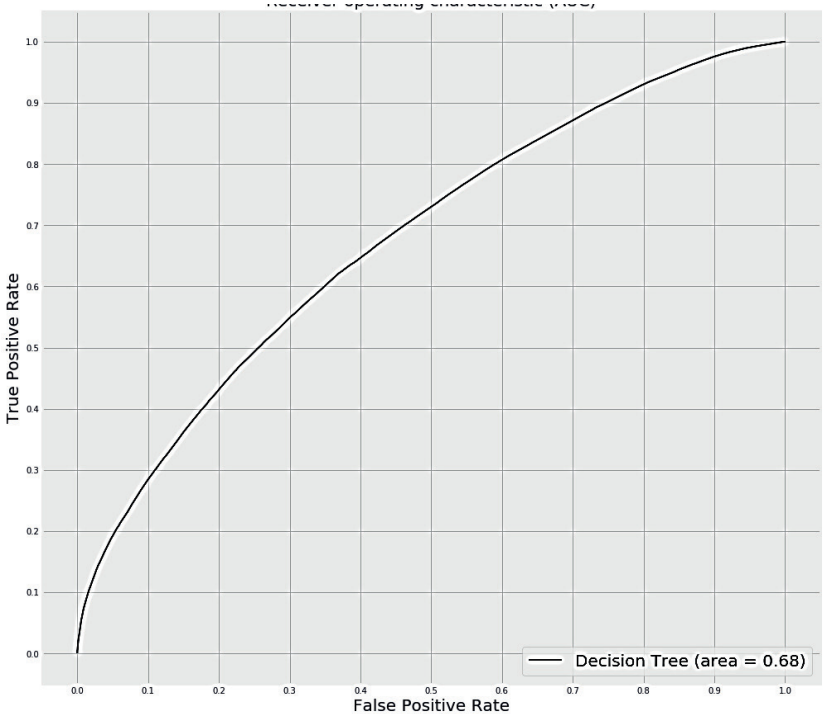
Na podstawie zaprezentowanych miar można oszacować jakość klasyfikatora, niemniej jednak trudno wnioskować o ekonomicznym sukcesie wdrożenia rozwiązania. Poleganie na prostej wartości wybranego wskaźnika może skończyć się implementacją rozwiązania o negatywnym skutku ekonomicznym.

Nawiązując do wcześniej zdefiniowanego celu eksperymentu, powinno się zdefiniować koszty podjęcia danej decyzji i obliczyć jej skutek ekonomiczny.

W celu przeprowadzania obliczeń przyjęto, że:

1) obsługa pojedynczego zwrotu (dalej zwana KOPZ) kosztuje nas 50 zł (wysyłka produktu, koszty pracy przy obsłudze, utracona marża, koszty ponownej sprzedaży produktu i inne);

2) przy każdej transakcji, dla której klasyfikator zdecyduje, że zostanie zwrócona, zaproponuje się klientowi wartość dodatkową (rabat, dodatkowy produkt za darmo, dodatkową usługę), co kosztuje nas średnio 10 zł; wartość ta będzie dalej zwana KJWD (koszt jednostkowy wartości dodatkowej).



Rys. 4. Krzywa ROC

Źródło: opracowanie własne przy użyciu biblioteki Scikit-learn.

Przy założeniu, że uruchomiono model dla 1 miliona transakcji zakupu (ten parametr nazywamy LT) i że obecnie 10% (wskaźnik zwrotów będziemy nazywali WZ) transakcji kończy się zwrotem, można oszacować, że obsługa zwrotów będzie kosztowała 5 mln zł. Problem obsługi zwrotów można też próbować rozwiązać w ten sposób, że każdemu klientowi dokonującemu transakcji oferowana będzie wartość dodatkowa w zamian za rezygnację ze zwrotu. Niestety, takie działanie kosztowałoby 10 mln zł, czyli przekroczyłoby koszt obsługi zwrotów. Dlatego do rozwiązania problemu zaproponowano użycie wspomnianego klasyfikatora, który dla danej transakcji dokona predykcji pojawiania się jej zwrotu i zdecyduje o zaproponowaniu klientowi wartości dodatkowej w zamian za rezygnację z możliwości dokonania zwrotu.

Należy zaznaczyć, że wraz z zastosowaniem modelu pojawiają się następujące koszty:

1. Koszt wartości dodatkowej (KWD) zaproponowanej klientowi, którego transakcję model sklasyfikował jaką taką, która zostanie zwrócona. Koszt występuje zarówno dla prawidłowej, jak i błędnej decyzji klasyfikatora. Koszt ten zatem jest niezależny bezpośrednio od wartości wskaźnika *precision*. Można go obliczyć na podstawie miary *positive rate* (stosunek transakcji wskazanych przez klasyfikator jako tych, które zostaną zwrócone do wszystkich przebadanych transakcji).

$$KWD = LT \times \frac{TP + FN}{TP + FP + TN + FN} \times KJWD ,$$

gdzie: *KWD* – koszt całkowity wartości dodatkowej; *LT* – liczba transakcji; *JWD* – koszt jednostkowy wartości dodatkowej.

2. Koszt obsługi zwrotu dla transakcji, których klasyfikator błędnie nie wskazał jako zwróconych. Koszt całkowity obsługi zwrotów (KOZ) jest ściśle związany z wartością wskaźnika *recall* (dla *recall* = 100%, koszt byłby zerowy).

$$KOZ = LT \times WZ \times \left(1 - \frac{TP}{TP + FN}\right) \times KJOZ ,$$

gdzie: *KOZ* – koszt całkowity obsługi zwrotów; *LT* – liczba transakcji; *WZ* – współczynnik zwrotów; *KJOZ* – koszt jednostkowy obsługi zwrotu.

W tabeli 3 zaprezentowano podsumowanie kosztów występujących przy zastosowaniu modelu oraz obliczono oszczędności w stosunku do obsługi zwrotów bez stosowania modelu. W wyliczeniu uwzględniono również szacunek ogólnych kosztów wdrożenia modelu, związanych z jego przygotowaniem oraz uruchomieniem aplikacji wykorzystującej model.

Tabela 3. Podsumowanie kosztów stosowania modelu

Rodzaj kosztu	Koszt [mln PLN]
Koszt obsługi zwrotów bez modelu	5
Koszt zwrotów, których model nie wykrył (KOZ)	2
Koszt zaproponowanej wartości dodatkowej (KWD)	1,1
Koszt implementacji modelu	0,05
Oszczędność przy zastosowaniu modelu	1,85

Źródło: opracowanie własne.

Dla przyjętych założeń oszczędność (wg których obsługa zwrotu jest kilkukrotnie droższa niż wartość dodatkowa) w procesie obsługi zwrotów wyniosła 37%. Przygotowanie kilku symulacji bazujących na różnych wartościach przyjętych parametrów wykazało, że najważniejszym parametrem oceny klasyfikatora jest *recall*.

Oczywiście autorzy artykułu zdają sobie sprawę z faktu, że przyjęta metoda obliczania wyniku ekonomicznego działania klasyfikatora bazuje na uproszczonych założeniach (np. nie wszyscy klienci zdecydują się na zamianę wartości dodatkowej

na możliwość dokonania zwrotu), niemniej jednak pokazuje sposób rozumowania zmierzającego do obiektywnej oceny klasyfikatora w kontekście konkretnego problemu ekonomicznego.

W kolejnych badaniach autorzy skoncentrują się na optymalizacji klasyfikatora, uwzględniającej maksymalizację spodziewanej oszczędność/korzyści ekonomicznej zamiast wybranego parametru ogólnej oceny klasyfikatora (takiego jak *precision*, *recall* czy *accuracy*).

5. Predykcja skorzystania z wyprzedaży

W kolejnym eksperymencie założenie biznesowe było inne. Celem eksperymentu było zoptymalizowanie kampanii marketingowej związanej z wyprzedażami organizowanymi w okresie tzw. *Black Friday*. Zwyczaj ten trafił do Polski z USA, gdzie w pierwszy piątek po Święcie Dziękczynienia organizowane są znaczące wyprzedaże, na które klienci czekają od dłuższego czasu. Pierwotnie wyprzedaże te były organizowane w piątek, następnie wprowadzono tzw. *Cyber Monday*, będący w założeniu przeznaczony na wyprzedaże w sklepach internetowych. Obecnie wiele podmiotów handlowych urządza niemal tygodniową wyprzedaż (por. [Boyd, Peters 2011; Swilley, Goldsmith 2013]).

Celem eksperymentu było dokonanie analizy wszystkich zarejestrowanych w sklepie klientów celem oszacowania prawdopodobieństwa dokonania przez nich zakupu w okresie *Black Friday*.

Dane, które posłużyły do zbudowania klasyfikatora, opisywały historię od początku działania sklepu do *Black Friday* w 2016 roku. Zmienną celu w tym przypadku był fakt dokonania zakupu w okresie *Black Friday*. Na podstawie danych historycznych zostały zbudowane dwa klasyfikatory. Pierwszy przy użyciu drzew decyzyjnych (wykorzystano tę samą bibliotekę co w poprzednim eksperymencie), zaś drugi przy użyciu algorytmu regresji liniowej (również dostępnego w bibliotece Scikit-learn). Problem, który pojawił się w analizie na samym początku badania, to niezrównoważenie zbioru. W zbiorze uczącym jedynie 2% klientów zostało sklasyfikowanych jako zainteresowanych zakupami w *Black Friday*. Po zbudowaniu klasyfikatora dokonano jego walidacji polegającej na uruchomieniu na wszystkich danych zebranych do tej pory. Oba klasyfikatory dla każdego klienta określiły prawdopodobieństwo dokonania zakupu w *Black Friday* 2017. Przy takim niezrównoważeniu zbioru podjęliśmy decyzję o przydzieleniu klienta do klasy „kupi" w przypadku, gdy co najmniej jeden z klasyfikatorów uzna, że jest przynajmniej 10% szansy na dokonanie zakupu przez klienta. Do wszystkich klientów sklasyfikowanych do grupy „kupi" skierowaliśmy kampanię poprzez Facebooka, a następnie zweryfikowaliśmy, którzy klienci faktycznie dokonali zakupu.

Wartości wskaźników dla tej kampanii są zawarte w tabeli 4.

W takich warunkach trudno było spodziewać się wysokich wskaźników *precision* oraz *recall* dla walidacji. Jednak biorąc pod uwagę fakt, że ze wszystkich klientów 2% dokonało zakupu w rzeczonym okresie, to wskaźnik *precision* na poziomie

Tabela 4. Wartości podstawowych wskaźników

Wskaźnik	Wartość
Accuracy	98,11%
Recall	6,18%
Precision	11,02%

Źródło: opracowanie własne.

11% oznacza zdecydowanie większą konwersję niż w przypadku kampanii definiowanej bez oparcia na wyniku klasyfikatora.

W kolejnych etapach prac podjęty będzie problem pomiaru efektywności ekonomicznej podejścia, przy założeniu, że błąd FP kosztuje utraconą korzyść z transakcji, do której nie dojdzie, a błąd FN to koszt nieskutecznie wyświetlonej klientowi reklamy.

6. Zakończenie

W artykule został przedstawiony problem eksploracji danych pochodzących z bazy transakcji sklepu internetowego w oparciu o zaproponowaną metodykę eksploracji danych wykorzystaną w projekcie RTOM. Problem biznesowy oraz aspekty implementacyjne zostały zaprezentowane w oparciu o algorytmy reguł asocjacyjnych FPGrowth oraz reguł sekwencyjnych PrefixSpan dostępne w bibliotece MLlib silnika Spark. Opisano również dwa dodatkowe eksperymenty polegające na przygotowaniu modeli klasyfikacyjnych wraz z ich oceną. Zaprezentowane przykłady pochodzą z rzeczywistej platformy handlu elektronicznego, jednak na potrzeby artykułu dane zostały zanonimizowane, tak aby nie zdradzać szczegółów biznesowych przedsięwzięcia internetowego. Sam proces eksploracji przyniósł wartościowe dla menedżerów marketingu informacje o wzorcach zachowań klientów, które do tej pory nie były odkryte. Dzięki rozbudowaniu procesu eksploracji i poddaniu tym samym działaniom różnych segmentów klientów wskazano specyfikę zachowań klientów należących do różnych segmentów, co przyczyniło się do skuteczniejszego przygotowania tak zwanych kampanii celowanych (*targeted campaigns*).

Literatura

Boyd T.J., Peters C., 2011, *An exploratory investigation of Black Friday consumption rituals*, International Journal of Retail & Distribution Management, 39(7), s. 522-537.

Chorianopoulos A., 2016, *Effective CRM Using Predictive Analytics*, John Wiley & Sons, Hoboken.

Gordon S., Linoff M., Berry J.A., 2011, Data *Mining Techniques: For Marketing, Sales, and Customer Relationship*, Wiley, Hoboken.

Gray R., Owen D., Sopher M.J., 1998, *Setting up a control system for your organization*, Nonprofit World, vol. 16, no. 3, s. 65-76.

Győrödi C., Győrödi R., Holban S., 2004, *A comparative study of association rules mining algorithms.* In Hungarian Joint Symposium on Applied Computational Intelligence, Oradea.

Han J., Fu Y., 1999, *Mining Multi level Association Rules in Large Databases*, IEEE Knowledge and Data Engineering, vol. 11, s. 798-805.

Han J., Pei J., Kamber M., 2012, *Data Mining: Concepts and Techniques*, Elsevier, Burlington.
Han J., Pei J., Yin Y., Mao R., 2004, *Mining frequent patterns without candidate generation: A frequent-pattern tree approach*, Data Mining and Knowledge Discovery, 8(1), s. 53-87.
Hjort K., 2013, *On Aligning Returns Management with the E-commerce Strategy to Increase Effectiveness*, Chalmers University of Technology, University of Borås.
Kim J., Larose R., 2003, *What makes e-commerce websites sticky? Interactivity and impulsivity in online browsing behavior*, The 2003 Annual Meeting of the International Communication Association.
Kołodko G., 2010, *Neoliberalizm i światowy kryzys gospodarczy*, Ekonomista, nr 1, s. 23-30.
Korczak J., Pondel M., 2017, *Metodyczne podejście do analizy i eksploracji danych marketingowych*, Proc. Kongres Informatyki Ekonomicznej, Poznań. Studia Ekonomiczne, nr 342, s. 52-71.
Kowalski J. (red.), 2013, *Rola polityki logistycznej*, Wydawnictwo Uniwersytetu Ekonomicznego we Wrocławiu, Wrocław.
Kowalski T., Nowak T., Pisarek W. (red.), 2003, *Aspekty zarządzania*, PWN, Warszawa.
Kulurkar P., Badole K., 2016, *Improving Association Rule Mining with Apriori Algorithm and Charm*, International Journal for Scientific Research and Development, vol. 3, issue 11.
Li H., Wang Y., Zhang D., Zhang M., Chang E.Y., 2008, *Pfp: parallel fp-growth for query recommendation*, eProceedings of the 2008 ACM Conferece on Recommender Systems, s. 107-114.
Morzy T., 2013, *Eksploracja danych. Metody i algorytmy*, Wydawnictwo Naukowe PWN, Warszawa.
OECD, 2010, Sprawozdanie dotyczące przygotowania Strategii Zielonego Wzrostu, http://www.oecd-ilibrary.org (12.02.2013).
Pawełoszek I., Korczak J., 2017, *From Data Exploration to Semantic Model of Customer*, Proc. IntelliSys, London, In Intelligent Systems Conference (IntelliSys), s. 382-388.
Pei J., Han J., Mortazavi-Asl B., Wang J., Pinto H., Chen Q., Dayal U., Hsu M., 2004, *Mining sequential patterns by pattern-growth: The PrefixSpan approach*, IEEE Transactions on Knowledge and Data Engineering, 16 (10), s. 1-17.
Pondel M., 2015, *A concept of enterprise Big Data and BI workflow driven platform*, Federated Conference on Computer Science and Information Systems (FedCSIS).
Pondel M., Korczak J., 2017, *Eksploracja danych transakcyjnych sklepu internetowego*, Zeszyty Naukowe Politechniki Częstochowskiej. Zarządzanie, nr 26, s. 132-145.
Pondel J., Pondel M., 2011, *Eksploracja danych w systemach e-commerce*, Prace Naukowe / Uniwersytet Ekonomiczny w Katowicach, Systemy wspomagania organizacji SWO 2011, s. 212-223.
Ratcliff C., 2014, *How fashion ecommerce retailers can reduce online returns. Blog text, Econsultancy. Saatavissa*, https://econsultancy.com/blog/65026-how-fashion-ecommerce-retailers-can-reduce-onlinereturns.
Schutt R., O'Neil C., 2013, *Doing Data Science: Straight Talk from the Frontline*, O'Reilly Media, Inc., Sebastopol.
Setia S., Jyoti D., 2013, *Multi-Level Association Rule Mining*, A Review, International Journal of Computer Trends and Technology (IJCTT), vol. 6, no. 3, s. 166-170.
Stacey R., 2016, *E-commerce Product Return Statistics and Trends*, INFOGRAPHICS.
Swilley E., Goldsmith R.E., 2013, *Black Friday and Cyber Monday: Understanding consumer intentions on two major shopping days*, Journal of Retailing and Consumer Services, 20(1), s. 43-50.
The World Bank, 2012, *Inclusive Green Growth: The Pathway to Sustainable Development*, DC, Washington.
Ustawa z 17 grudnia 2004 r. o odpowiedzialności za naruszenie dyscypliny finansów publicznych, Dz.U. nr 14, poz. 114 ze zm.
Walsh G., Möhring M., Koot C., Schaarschmidt M., 2014, *Preventive product returns management systems – A review and model*, ECIS 2014 Proceedings.
Weichbroth P., 2009, *Odkrywanie reguł asocjacyjnych z transakcyjnych baz danych*, Prace Naukowe Uniwersytetu Ekonomicznego we Wrocławiu, nr 82, s. 301-309.
Wood S.L., 2001, *Remote purchase environments: The influence of return policy leniency on two-stage decision processes*, Journal of Marketing Research, 38(2), s. 157-169.