Autoreferat

1. Imię i nazwisko

Agnieszka Ławrynowicz

2. Posiadane dyplomy, stopnie naukowe – z podaniem nazwy, miejsca i roku ich uzyskania oraz tytułu rozprawy doktorskiej

1. Stopień doktora nauk technicznych - Politechnika Poznańska, Wydział Informatyki i Zarządzania; dyscyplina – informatyka; specjalizacja – sztuczna inteligencja:
   — data nadania: 05 maja 2009r.,
   — tytuł rozprawy doktorskiej: "Frequent Pattern Discovery from Knowledge Bases in Description Logic with Rules",
   — promotor: prof. dr hab. inż. Joanna Maria Józefowska.
2. Tytuł zawodowy magistra – Politechnika Poznańska, Wydział Elektryczny:
   — kierunek: informatyka,
   — specjalizacja: komputerowo zintegrowane systemy wytwarzania i zarządzania,
   — data nadania: 31 października 2000r.,
   — tytuł pracy magisterskiej: "Algorytmy uczenia się ze wzmocnieniem z reprezentacją opartą na klasyfikacji zakresowej".
3. Tytuł zawodowy inżyniera – Politechnika Poznańska, Wydział Elektryczny:
   — kierunek: informatyka,
   — data nadania: 30 lipca 1998r.

3. Informacje o dotychczasowym zatrudnieniu w jednostkach naukowych

1. Asystent:
   — Instytut Informatyki, Politechnika Poznańska,
   — okres: 1 stycznia 2003r.–30 września 2009r.
2. Adiunkt:

— Instytut Informatyki, Politechnika Poznańska,
— okres: 1 października 2009r.–obecnie

3. Staż naukowy EU Marie Curie Fellowship, Personet programme (Personalising E-Commerce using Web Mining), University of Ulster, Irlandia Pn.; czerwiec 2004r.–sierpień 2004r.
4. Zatrudnienie w ramach projektu DIPIS, University of Bari, Włochy; listopad 2009r.

4. Wskazanie osiągnięcia wynikającego z art. 16 ust. 2 ustawy z dnia 14 marca 2003 r. o stopniach naukowych i tytule naukowym oraz o stopniach i tytule w zakresie sztuki (Dz. U. 2016 r. poz. 882 ze zm. w Dz. U. z 2016 r. poz. 1311.)

4.1. Tytuł osiągnięcia naukowego

Semantyczna eksploracja danych. Metodyka oparta o ontologie.

4.2. Autor, tytuł publikacji, rok wydania, nazwa wydawnictwa, recenzenci wydawniczy

Agnieszka Ławrynowicz, *Semantic Data Mining: An Ontology-based Approach*, volume 29 of *Studies on the Semantic Web*. IOS Press/AKA Verlag, 2017.

4.3. Omówienie celu naukowego ww. pracy i osiągniętych wyników wraz z omówieniem ich ewentualnego wykorzystania

4.3.1. Wprowadzenie

Przedstawiona monografia "*Semantic data mining. An ontology-based approach*" [31] dotyczy zagadnienia *semantycznej eksploracji danych*. Istotą semantycznej eksploracji danych jest użycie w procesie eksploracji nie tylko surowych danych ale także *ontologii* i/lub *grafów wiedzy*, które dostarczają symbolicznej wiedzy dziedzinowej. Tematyka ta mieści się w obszarze *sztucznej inteligencji*, w tym reprezentacji wiedzy i wnioskowania, inżynierii wiedzy oraz statystyki i systemów baz danych. Praca łączy w szczególności obszary eksploracji danych i inżynierii wiedzy.

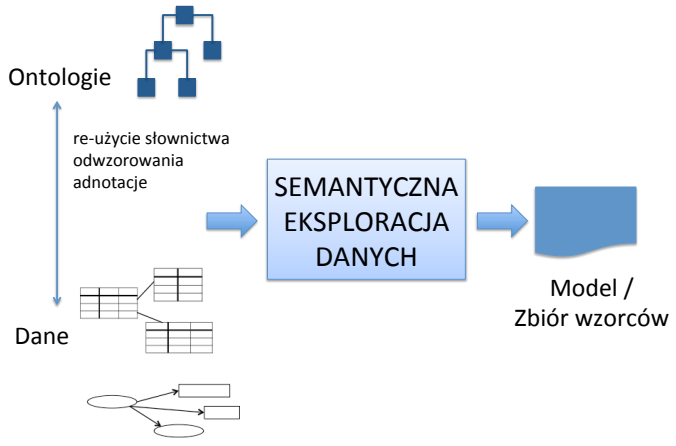
Eksploracja danych jest dziedziną, która dotyczy procesów odkrywania interesujących i użytecznych wzorców i wiedzy z dużych wolumenów danych [10, 15]. Procesy eksploracji danych przyjmują na wejściu *dane* i produkują *modele* lub *zbiory wzorców*. Dane mogą przyjmować różne formy takie jak pojedyncza tabela danych bądź też dokumenty tekstowe. Pojedyncza tabela danych jest klasycznym formatem, za pomocą którego każdy *przykład* reprezentowany jest poprzez ustaloną liczbę *atrybutów* (*cech*). Jednakże, dane nie zawsze mają tego rodzaju płaską strukturę. W praktyce, wiele rzeczywistych problemów eksploracji danych wymaga manipulowania danymi o rdzennie złożonej strukturze, pomiędzy którymi występuje wiele powiązań. W związku z tym, prowadzone były na świecie prace badawcze w obszarze tzw. *relacyjnej eksploracji danych* [8, 58]. Metody relacyjnej eksploracji danych radzą sobie bezpośrednio z danymi w formacie relacyjnej bazy

danych, grafu, zbioru faktów logicznych i innych, bez ich uprzedniego przetwarzania wstępnego do formatu dwuwymiarowej tabeli.

Istotną częścią metodyki przedstawionej w monografii jest wykorzystanie *wiedzy dziedzinowej* w eksploracji danych. Szerokie badania nad wykorzystaniem wiedzy dziedzinowej w ramach eksploracji danych relacyjnych przeprowadzono w dziedzinie *indukcyjnego programowania w logice* [50, 28], którego celem jest opracowanie metod uczenia się indukcyjnego relacyjnych modeli reprezentowanych w podzbiorach logiki pierwszego rzędu. W metodach indukcyjnego programowania w logice przyjętym formalizmem reprezentacji wiedzy były często programy logiczne [47, 19] wyrażone najczęściej za pomocą języków opartych na formalizmie klauzul Horna. W ostatnich latach coraz bardziej popularną formą reprezentacji wiedzy stały się jednak *ontologie* [63].

Ontologie są powszechnie przyjętym formalizmem do reprezentowania wiedzy terminologicznej, koncepcyjnej i jako takie są odpowiednie do wyrażenia formalnej semantyki ("znaczenia") opisywanych danych. Ontologie są coraz częściej wykorzystywane do integracji, opisu i organizacji danych i wiedzy, zwłaszcza w aplikacjach opartych na intensywnym wykorzystaniu danych i wiedzy w badaniach naukowych i przemyśle. W książce uwzględniłam formalizmy reprezentacji wiedzy, które są odpowiednie do wyrażania wiedzy dziedzinowej za pomocą ontologii oraz poświęciłam uwagę zagadnieniu wykorzystania tego rodzaju wiedzy w sposób automatyczny w procesach eksploracji danych.

Semantyczna eksploracja danych [52] jest podejściem, w którym ontologie są nie tylko wykorzystywane jako źródła wiedzy dziedzinowej, ale gdzie nowym wyzwaniem jest eksploracja także wiedzy zakodowanej w ontologiach i grafach wiedzy oprócz tylko czysto empirycznych danych. Rysunek 1 ilustruje to wyzwanie pokazując, że na wejściu do procesu eksploracji mogą znajdować się dane adnotowane terminami z ontologii, bądź też bezpośrednio same ontologie.



Rysunek 1. Semantyczna eksploracja danych.

Monografia zawiera syntetyczny opis dziedziny semantycznej eksploracji danych, podsumowując niektóre z kluczowych problemów i wyników i zarazem będąc pierwszą książką, która przedstawia jednolity opis tego obszaru badań. Rozdziały 1-3 zawierają wstęp oraz teoretyczne podstawy dotyczące zarówno języków reprezentacji ontologii i baz wiedzy jak i podstaw eksploracji danych z perspektywy semantycznej eksploracji danych. Opisane najbardziej szczegółowo metody semantycznej eksploracji danych (rozdziały 4–6) jak i aplikacje (rozdziały 8–10) są tymi, których jestem autorką bądź współautorką.

4.3.2. Podstawy semantycznej eksploracji danych i ontologie

W pierwszej części rozdziału 1 monografii wprowadziłam pojęcie semantycznej eksploracji danych w kontekście metod relacyjnej i opartej o programowanie logiczne eksploracji danych oraz eksploracji danych wykorzystującej różne formy wiedzy dziedzinowej. Dokonałam także krótkiego przeglądu podejść semantycznej eksploracji danych. Mój wkład naukowy do jakiego odnoszę się w tym fragmencie dotyczy współautorskiego rozdziału (wraz z Prof. Volkerem Trespem) [42] w książce zatytułowanej "Perspectives on Ontology Learning", edytowanej przez Prof. Johannę Voelker i Prof. Jensa Lehmanna [44]. W tym współautorskim rozdziale przedstawiłam zagadnienia uczenia maszynowego w kontekście danych opisanych semantycznie za pomocą ontologii i słowników danych.

W drugiej część rozdziału 1 dokonałam wprowadzenia do tematu ontologii w informatyce.

4.3.3. Języki reprezentacji ontologii

W rozdziale 2 monografii omówiłam języki reprezentacji ontologii wraz z odpowiadającymi im językami zapytań. W szczególności przedstawiłam języki *Resource Description Framework (RDF)* [60] i *RDF Schema (RDFS)* [13, 14] (jego podzbiór o nazwie ρdf), język zapytań *SPARQL* [57], formalizm logik deskrypcyjnych [2] (oraz zapytań koniunkcyjnych do baz wiedzy reprezentowanych w logikach deskrypcyjnych) oraz język modelowania ontologii *Web Ontology Language (OWL)* [64].

RDF jest językiem opisu zasobów. RDF udostępnia format danych oparty na grafach, który umożliwia wyrażanie zdań opisujących zasoby w sieci Web i ich własności w formie trójek: podmiot–predykat–obiekt. Rozważmy parami rozłączne, nieskończone zbiory $\mathbf{U}$, $\mathbf{B}$ i $\mathbf{L}$. Oznaczają one odpowiednio: referencje URI, puste węzły i literały. *Trójka RDF* jest krotką, gdzie s jest podmiotem, p jest predykatem, a o obiektem trójki. *Graf RDF* (lub *zbiór danych RDF*) G jest zbiorem trójek RDF.

RDFS rozszerza semantykę języka RDF przez dostarczenie słownictwa do opisu i ustrukturalizowania zasobów RDF. RDFS dostarcza podstawowych elementów do modelowania prostych ontologii. Muńoz i in. [51] zdefiniowali fragment RDFS nazywany ρdf, który obejmuje podstawowe elementy języka RDFS, tj. elementy służące do modelowania: relacji przynależności instancji do klasy (type), relacji klasa–podklasa, relacji własność-podwłasność oraz dziedziny i przeciwdziedziny własności. Muńoz i in. [51] zdefiniowali także reguły wnioskowania dla fragmentu RDFS ρdf, których aplikacja na danym grafie G generuje nowe trójki.

SPARQL jest językiem zapytań do pobierania i manipulowania danymi reprezentowanymi w RDF. Zapytanie Q reprezentowane w języku SPARQL składa się z ciała zapytania *body*(Q) i nagłówka zapytania *head*(Q). Ciało zapytania ma postać wzorca grafowego RDF, na który składają się wzorce trójkowe RDF zawierające zmienne, koniunkcje, dysjunkcje, części opcjonalne, a także ograniczenia wartości zmiennych. Nagłówek zapytania składa się ze słowa kluczowego określającego formę odpowiedzi na zapytanie: odpowiedź typu logicznego (prawda/fałsz), tabela wartości lub graf RDF. Najczęściej spotykaną formą jest zapytanie typu SELECT, którego wynikiem jest tabela wartości. Ewaluacja zapytania SPARQL opiera się zasadniczo na dopasowywaniu grafów.

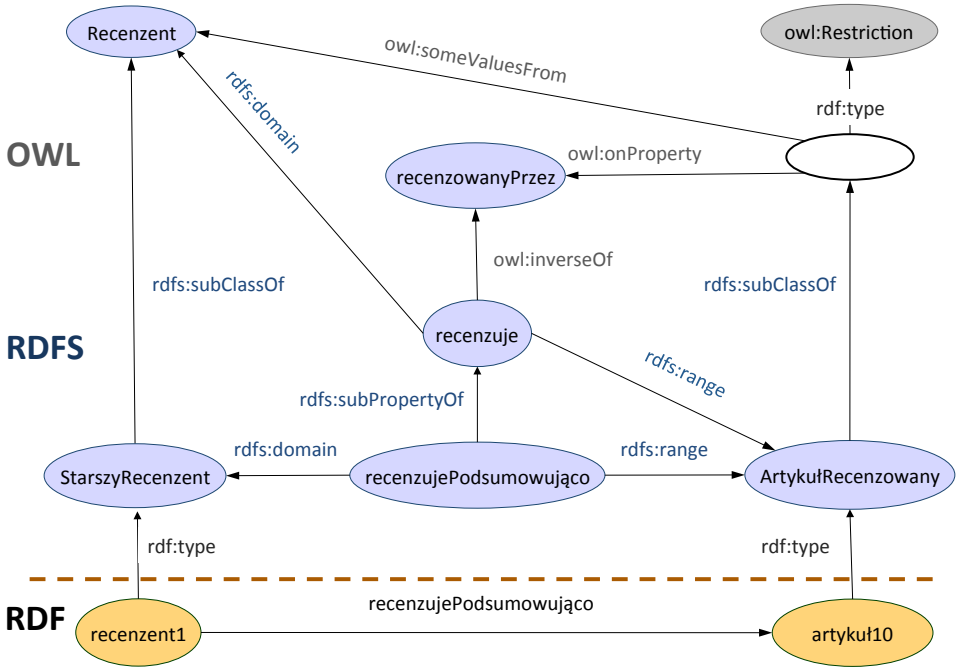
Logiki deskrypcyjne to rodzina języków zaprojektowanych w celu formalnej reprezentacji wiedzy terminologicznej, szeroko stosowanych do modelowania ontologii. Typowe logiki deskrypcyjne to fragmenty logiki pierwszego rzędu ze składnią ograniczoną do formuł mających maksymalnie dwie zmienne i nie zawierających symboli funkcyjnych. Te fragmenty zostały wybrane ze względu na rozstrzygalność problemów wnioskowania związanych z reprezentacją wiedzy i ze względu na odpowiedni stopień ekspresywności języka. W logikach deskrypcyjnych wyróżnia się trzy rodzaje symboli atomowych: *atomowe pojęcia*, oznaczone A, które reprezentują zbiory indywiduuów, *atomowe role*, oznaczone przez R i S, które reprezentują binarne relacje między indywiduami i *indywidua*, oznaczone przez a i b, które reprezentują pojedyncze instancje w danej dziedzinie. Poprzez N_C, N_R, N_I oznacza się, odpowiednio, zbiory *nazw pojęć*, *nazw ról abstrakcyjnych* oraz *nazw indywiduuów*. Ponadto, w niektórych językach logik deskrypcyjnych występują też *role konkretne*, które łączą indywidua z wartościami z różnych, innych dziedzin, takich jak liczby całkowite, łańcuchy znaków, daty itp., a zbiór *nazw ról konkretnych* jest oznaczony poprzez N_P.

Symbole atomowe to podstawowe *deskrypcje*, z których budowane są *złożone deskrypcje* przy użyciu *konstruktorów* logicznych, np.: $\sqcap$ (iloczyn logiczny), $\exists$ (kwantyfikator egzystencjalny), $\neg$ (negacja) itp. *Baza wiedzy* w logice deskrypcyjnej KB jest zbiorem *aksjomatów*, które dzielą się zazwyczaj na część *terminologiczną* ($TBox$) bazy wiedzy oraz na część *asercyjną* ($ABox$) bazy wiedzy. TBox zawiera aksjomaty reprezentujące relacje pomiędzy pojęciami i rolami, np. *subsumcję* ($\sqsubseteq$) i *równoważność* ($\equiv$) pojęć lub ról, wyrażające ogólną wiedzę na temat dziedziny. ABox zawiera asercje indywiduów do pojęć i ról, wyrażając wiedzę o pojedynczych obiektach w dziedzinie.

Poprzez N_V oznaczmy nieskończony, przeliczalny zbiór *zmiennych* v, który jest rozłączny z N_C, N_R, N_P i N_I. Poprzez $\mathbf{x}$ i $\mathbf{y}$ oznaczmy, odpowiednio, zbiory wyróżnionych i niewyróżnionych zmiennych, gdzie $\mathbf{x}, \mathbf{y} \subseteq N_V$. *Zapytanie koniunkcyjne* do bazy wiedzy w logice deskrypcyjnej, oznaczone $Q(\mathbf{x}, \mathbf{y})$ lub po prostu Q, jest skończonym zbiorem atomów, gdzie *atom* B jest wyrażeniem $C(v)$ ($C \in N_C$) lub $R(v, v')$ ($R \in N_R$) lub $P(v, v')$ ($P \in N_P$) i $v, v' \in N_V$.

OWL jest językiem opartym na formalizmie logik deskrypcyjnych, a jednocześnie opierającym się na technologiach sieci Web i Sieci Semantycznej. OWL jest standardem rekomendowanym przez konsorcjum World Wide Web (W3C). OWL posiada kilka serializacji, z których jedna opiera się na składni języka RDF. Zarówno RDFS jak i OWL rozszerzają RDF dostarczając dodatkowe słownictwo, które można umieścić w dokumentach RDF i interpretować jako ontologie. Słownictwo to po części odpowiada elementom formalizmu logik deskrypcyjnych.

Rysunek 2 ilustruje jak dane reprezentowane w języku RDF wraz z ich semantycznym opisem



Rysunek 2. Ilustracja przykładowego grafu wiedzy (sieci semantycznej) pokazująca trójki RDF, które zawierają słownictwo RDFS i OWL. Trójki RDF nie zawierające takiego słownictwa mogą być postrzegane jako fakty w grafie wiedzy, których semantyka jest wyrażona poprzez aksjomaty ontologii serializowane do trójek RDF zawierających słownictwo wyrażające aksjomaty ontologii.

wyrażonym w językach RDFS i OWL (serializowanym jako trójki RDF) mogą utworzyć jeden *graf wiedzy* (*sieć semantyczną*).

Niektóre słownictwo w grafie ma specjalne znaczenie związane z semantyką języków RDFS i OWL. W szczególności, graf może mieć swoją logiczną reprezentację, np. odpowiadającą następującej bazie wiedzy wyrażonej w logice deskrypcyjnej:

TBox $\mathcal{T}$	Atomowe pojęcie: Recenzent Role: recenzuje, recenzujePodsumowująco, recenzowanyPrzez Konstruktory: $\sqcap$, $\exists$ Aksjomaty: ArtykułRecenzowany $\sqsubseteq \exists$recenzowanyPrzez.Recenzent StarszyRecenzent $\sqsubseteq$ Recenzent recenzujePodsumowująco $\sqsubseteq$ recenzuje $\top \sqsubseteq \forall$recenzuje.ArtykułRecenzowany $\top \sqsubseteq \forall$recenzuje$^-$.Recenzent recenzuje $\equiv$ recenzowanyPrzez$^-$
ABox $\mathcal{A}$	Asercja faktu: recenzujePodsumowująco(recenzent1, artykuł10)

4.3.4. Eksploracja danych jako przeszukiwanie

W rozdziale 3 monografii przedstawiłam podstawy teoretyczne zagadnień eksploracji danych. Problemy eksploracji danych są w tym rozdziale dyskutowane jako problemy przeszukiwania przestrzeni hipotez. Celem przeszukiwania jest znalezienie hipotez spełniających określone miary jakości. W rozdziale omówione są relacje uogólnienia pomiędzy hipotezami w przestrzeni. Relacje te strukturalizują przestrzeń przeszukiwania. Przedstawiona jest także definicja operatorów uszczegółowienia/uogólnienia, czyli funkcji służących do przemieszczania się w przestrzeni pomiędzy hipotezami o różnym stopniu ogólności.

Moim oryginalnym wkładem naukowym opisanym w tym rozdziale jest zdefiniowanie nowej relacji uogólnienia, nazwanej *taksonomiczną subsumcją* (lub t-subsumcją) oraz pojęcia *taksonomicznie domkniętego wzorca* (oryginalnie wprowadzone w [39]). Taksonomiczna subsumcja, pod względem swojej siły ekspresji, plasuje się pomiędzy czysto syntaktyczną relacją uogólnienia jaką jest θ-subsumcja [53] a w pełni semantyczną relacją uogólnienia, tj. taką która uwzględnia pełną semantykę danego języka. W szczególności, taksonomiczna subsumcja uwzględnia wnioskowanie dotyczące hierarchii klas i własności. Definicję taksonomicznej subsumcji podałam dla baz wiedzy wyrażonych w podzbiorze ρdf języka RDFS. Dotyczy ona wzorców w postaci zapytań SPARQL typu SELECT, zawierających zmienną wyróżnioną $\mathbf{x_i}$, wzorce trójkowe i ograniczenia wartości zmiennych FILTER.

Definicja taksonomicznej subsumcji opiera się na relacji *pokrycia* wzorców wyrażonych jako zapytania SPARQL do bazy wiedzy w RDFS, tzn. relacji, która określa czy wzorzec Q *pokrywa* dany przykład e. Mając dane zapytanie Q zawierające wzorzec grafowy SPARQL GP i graf RDF G, mówi się, że Q *pokrywa* przykład $(\mathbf{x_i}, y_i)$, jeżeli istnieje odwzorowanie $\mu \in [\![GP]\!]_{cl(G)}$, które odwzorowuje zmienne z head(Q) do referencji URI identyfikującej $\mathbf{x_i}$ (gdzie cl oznacza domknięcie grafu RDF uzyskane poprzez sukcesywną aplikację reguł wnioskowania, dopóki mogą być wygenerowane nowe trójki).

Pojęcie *taksonomicznie domkniętego* wzorca opiera się na podzbiorze reguł wnioskowania, mianowicie reguł, które dotyczą dziedziczenia względem hierarchii klas i własności. Biorąc pod uwagę język reprezentacji wzorców i bazy wiedzy (język SPARQL i ρdf, odpowiednio), taksono-

micznie domknięte zapytanie jest zapytaniem do którego nie można już dodać więcej wzorców trójkowych o postaci ($?x$ type C), jeśli wzorzec trójkowy o takiej postaci i ze zmienną $?x$ już istnieje w zapytaniu, lub o postaci ($?x$ P $?y$) gdy wzorzec trójkowy o takiej postaci i ze zmiennymi $?x$ i $?y$ już istnieje w zapytaniu, bez wpływu na semantykę wzorca.

Definicja 1 (Taksonomiczna subsumcja)**.** Mając dane dwa wzorce Q_1 i Q_2 na zbiorze danych ρdf G i ich semantyczne domknięcia (tzn. taksonomicznie domknięte wzorce Q_1^t i Q_2^t, odpowiednio), Q_1 *taksonomicznie zawiera* Q_2 wtedy i tylko wtedy jeśli istnieje odwzorowanie σ takie, że zbiór wzorców trójkowych i wyrażeń typu FILTER z $(body(Q_1^t))\sigma$ jest podbiorem zbioru wzorców trójkowych i wyrażeń typu FILTER z $(body(Q_2^t))\sigma$.

Główny mój samodzielny wkład naukowy opisany w tym rozdziale to:
— zdefiniowanie nowej relacji uogólnienia, nazwanej *taksonomiczną subsumcją*, dla podzbioru ρdf języka RDFS,
— zdefiniowanie taksonomicznie domkniętych wzorców.

4.3.5. Odkrywanie częstych złożonych klas w ontologicznych bazach wiedzy

W rozdziale 4 monografii przedstawiłam problem odkrywania częstych wzorców w ujęciach od prostych zbiorów wzorców, poprzez wzorce wielo-relacyjne w relacyjnych bazach danych, klauzule logiczne w programach w logice aż do problemu odkrywania częstych złożonych klas w ontologiach. Ostatnie wymienione ujęcie jest moim oryginalnym wkładem naukowym i zostało po raz pierwszy przedstawione w artykule konferencyjnym autorstwa Ławrynowicz i Potońca [38]. W rozdziale omówiono algorytm odkrywania częstych pojęć logiki deskrypcyjnej o nazwie Fr-ONT (zaproponowany w [38]), który rozwiązuje wariant tego zadania, tzn. odkrywanie częstych wzorców wyrażonych jako pojęcia w języku logik deskrypcyjnych $\mathcal{EL}^{++}$.

W zadaniu odkrywania częstych wzorców w postaci złożonych pojęć wyrażonych w logikach deskrypcyjnych, problem odkrywania wzorców jest sformułowany jako problem przeszukiwania przestrzeni pojęć logik deskrypcyjnych. Przy formułowaniu algorytmu Fr-ONT zastosowano podejście oparte na operatorach uszczegóławiania, definiując operator uszczegóławiania, który konstruuje specjalizacje hipotez, które w tym przypadku przyjmują postać pojęć w logikach deskrypcyjnych.

Poprzez $(\mathcal{L}_h, \succeq)$ oznaczmy quasi-uporządkowaną przestrzeń pojęć w logice deskrypcyjnej. Operator uszczegółowienia ρ dla logik deskrypcyjnych produkuje odwzorowanie z $\mathcal{L}_h$ do $2^{\mathcal{L}_h}$, takie, że dla każdego $C \in \mathcal{L}_h$, $C' \in \rho(C)$ implikuje $C \succeq C'$ (C jest bardziej ogólne od C'). W takim sformułowaniu, naturalne uporządkowanie przestrzeni przeszukiwania jest wyznaczone przez subsumcję pomiędzy pojęciami reprezentowanymi w logice deskrypcyjnej. Co więcej, jeśli C uogólnia D ($D \sqsubseteq C$), to C pokrywa wszystkie instancje, które są pokryte poprzez D. Dlatego subsumcja pomiędzy pojęciami może służyć jako relacja uogólnienia.

Miarą oceny wzorców powszechnie wykorzystywaną w celu generowania częstych wzorców jest miara *wsparcia*. W [38] zaproponowaliśmy obliczanie wsparcia wzorca, który jest arbitralnym pojęciem C, w odniesieniu do liczby instancji tak zwanego *pojęcia referencyjnego* $\hat{C}$. Pojęcie $\hat{C}$

jest zdefiniowane przez użytkownika i jest punktem w przestrzeni hipotez, od którego zaczyna się przeszukiwanie od najbardziej ogólnych do najbardziej szczegółowych pojęć.

Definicja 2 (Wsparcie)**.** Niech C oznacza pojęcie w logice deskrypcyjnej, przez $KB = (TBox, ABox)$ oznaczona niech będzie baza wiedzy w logice deskrypcyjnej, przez $memberset(C, KB)$ funkcja zwracająca zbiór wszystkich indywiduów a takich, że $ABox \models C(a)$, i niech $\hat{C}$ oznacza pojęcie referencyjne, takie że $\hat{C}$ jest pojęciem pierwotnym, oraz $C \sqsubseteq \hat{C}$.
Wsparcie (*support*) wzorca C w odniesieniu do bazy wiedzy KB jest zdefiniowane jako stosunek liczby wystąpień instancji pojęcia C do liczby wystąpień instancji pojęcia referencyjnego $\hat{C}$ w KB: $support(C, KB) = \frac{|memberset(C,KB)|}{|memberset(\hat{C},KB)|}$.

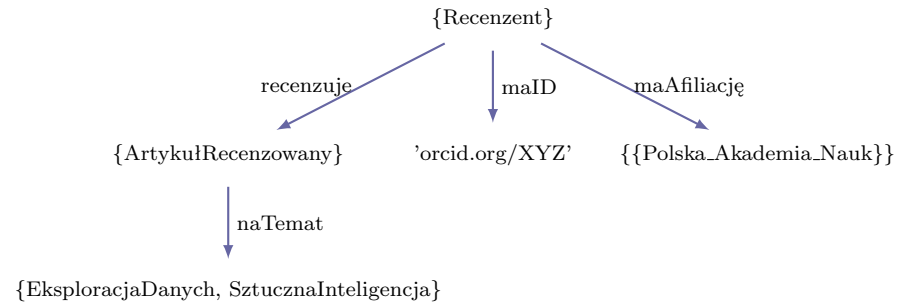
Korzystając z definicji wsparcia, można przedstawić definicję *odkrywania częstych pojęć logiki deskrypcyjnej* (por. [38]).

Definicja 3 (Odkrywanie częstych pojęć logiki deskrypcyjnej)**.** Mając daną bazę wiedzy w logice deskrypcyjnej KB, język $\mathcal{L}_h$ wzorców w postaci pojęć logiki deskrypcyjnej C, gdzie każde pojęcie C jest specjalizacją pojęcia referencyjnego $\hat{C}$ ($C \sqsubseteq \hat{C}$), minimalny próg wsparcia *minsup* określony przez użytkownika i zakładając, że wzorce ze wsparciem s są częste w KB jeżeli $s \geqslant minsup$, zadaniem *odkrywania częstych pojęć logiki deskrypcyjnej* jest znalezienie zbioru częstych wzorców.

Operator uszczegółowienia algorytmu Fr-ONT wykorzystuje reprezentację pojęć w języku $\mathcal{EL}^{++}$ w formie *drzewa pojęcia* [3, 43]. *Drzewo pojęcia* w języku $\mathcal{EL}^{++}$ $T = (V, E)$ jest skierowanym, etykietowanym drzewem, gdzie V oznacza skończony zbiór *wierzchołków*, podczas gdy $E \subseteq V \times V$ oznacza zbiór *krawędzi*. Krawędzie drzewa są etykietowane elementami z N_R lub N_P. Korzeń drzewa pojęcia jest etykietowany przez $\top$ lub $\bot$ lub wszystkie elementy z $N_C \in \mathsf{prim}(C)$. Jeżeli ograniczenie egzystencjalne $\exists R_k.C'$ występuje na najwyższym poziomie danego pojęcia C, wtedy do drzewa dodawana jest krawędź etykietowana poprzez R_k i prowadząca do poddrzewa reprezentującego C'. W przypadku, gdy C' jest pojęciem nominalnym ($\{a\}$), poddrzewo nie jest rozszerzane dalej i staje się liściem. Jeśli istnieje ograniczenie egzystencjalne $\exists P_l.f$ występujące na najwyższym poziomie C, to dodaje się krawędź (oznaczoną P_l) do wierzchołka będącego liściem, reprezentującego konkretną wartość f. Pojęcie uniwersalne ($\top$) jest reprezentowane jako pusta etykieta wierzchołka. Rysunek 3 przedstawia przykładowe drzewo pojęcia.

Operacje ρ w zakresie manipulowania pojęciami to: (a) dodanie pierwotnego pojęcia jako nowego czynnika iloczynu, (b) zastąpienie pierwotnego pojęcia jego podpojęciem, (c) dodanie egzystencjalnego ograniczenia z rolą abstrakcyjną i z pojęciem uniwersalnym $\top$ w przeciwdziedzinie, (d) dodanie egzystencjalnego ograniczenia z rolą abstrakcyjną i z pojęciem nominalnym jako ograniczeniem wartości jako nowego czynnika iloczynu, (e) zastąpienie jednego czynnika iloczynu jego uszczegółowieniem wynikającym z zastąpienia abstrakcyjnej roli jej podrolą, (f) dodanie egzystencjalnego ograniczenia z rolą konkretną jako nowego czynnika iloczynu, g) zastąpienie jednego czynnika iloczynu jego uszczegółowieniem wynikającym z zastąpienia konkretnej roli przez jej podrolę, (h) rekursywne uszczegółowienie przeciwdziedziny abstrakcyjnej roli wynikające z zastosowania operatora ρ.

Główny mój samodzielny wkład naukowy opisany w rozdziale 4 monografii to:
— sformułowanie problemu odkrywania częstych pojęć w logice deskrypcyjnej,



Rysunek 3. Drzewo pojęcia odpowiadające następującemu pojęciu: $\mathsf{Recenzent} \sqcap \exists \mathsf{recenzuje}.(\mathsf{ArtykułRecenzowany} \sqcap \exists \mathsf{naTemat}.(\mathsf{EksploracjaDanych} \sqcap \mathsf{SztucznaInteligencja})) \sqcap \exists \mathsf{maID}_{'orcid.orgXYZ'} \sqcap \exists \mathsf{maAfiliację}.\{\mathsf{Polska\_Akademia\_Nauk}\}$

— zdefiniowanie kanonicznej reprezentacji pojęć w logice deskrypcyjnej $\mathcal{EL}^{++}$,
— operator uszczegółowienia dla pojęć reprezentowanych w logikach deskrypcyjnych $\mathcal{EL}^{++}$,
— opracowanie algorytmu Fr-ONT do odkrywania częstych pojęć logiki deskrypcyjnej $\mathcal{EL}^{++}$.

4.3.6. Klasyfikacja

Rozdział 5 monografii poświęcony jest zagadnieniu klasyfikacji, ze szczególnym uwzględnieniem zagadnienia klasyfikacji opartej o odkrywanie wzorców (ang. *pattern based classification*). W rozdziale omówiono algorytm Fr-ONT-Qu, który realizuje wariant takiego zadania, uwzględniając semantykę języka ρdf.

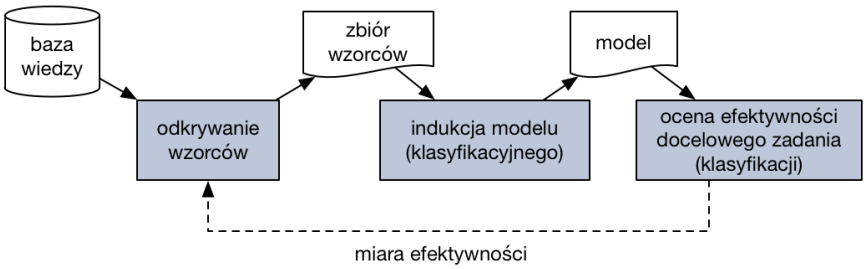
Algorytm Fr-ONT-Qu, zaproponowany oryginalnie w artykule w czasopiśmie z listy JCR przez Ławrynowicz i Potońca [39] jest metodą opartą o odkrywanie wzorców, o następującej charakterystyce:

— iteracyjne i poziom po poziomie (względem wzrastającej szczegółowości wzorców) wykonywanie podrzędnego algorytmu odkrywania wzorców,
— selekcja wzorców odbywa się względem miary efektywności, która uwzględnia optymalizację zadania klasyfikacji.

Kluczowe komponenty algorytmu to operator uszczegóławiania wzorców oraz strategia wyboru najlepszych wzorców do uszczegółowienia w każdej iteracji algorytmu. Ideę algorytmu ilustruje Rysunek 4. Zasadniczo, algorytm Fr-ONT-Qu realizuje naprzemiennie dwa etapy obliczeń:

1. *uszczegółowienie* każdego z wzorców z poprzedniej iteracji za pomocą operatora uszczegóławiania,
2. *ocena* wzorców i *wybór* najlepszych wzorców.

Do generowania kolejnych wzorców kandydujących wykorzystywane jest k najlepszych wzorców, posortowanych według malejącej jakości. Liczba iteracji jest ograniczona wartością parametru MAXLEVEL, podaną przez użytkownika. Algorytm odkrywania dyskryminacyjnych wzorców wykonuje więc przeszukiwanie wiązkowe. Przeszukiwanie rozpoczyna się od zapytania bazowego Q_{base}, następnie w powtarzalny sposób generowane są wzorce kandydujące $Q \in \mathbb{Q}$ za pomocą operatora uszczegóławiania ρ i ewaluuowana jest ich jakość. Algorytm stosuje metodykę odkry-



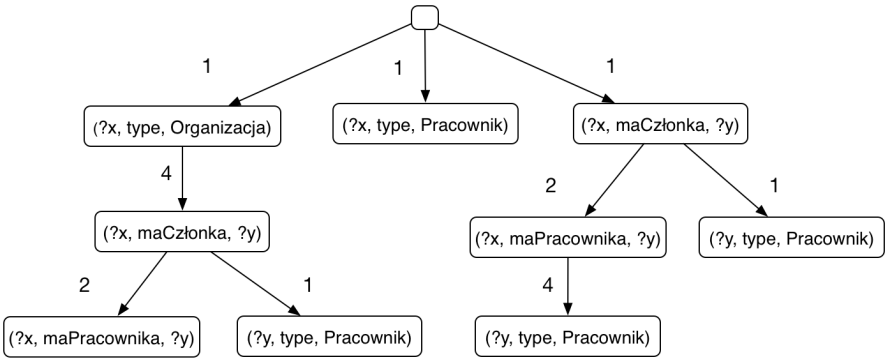
Rysunek 4. Ogólna ilustracja algorytmu Fr-ONT-Qu jako zależnego od modelu iteracyjnego podejścia do klasyfikacji opartej na odkrywaniu wzorców. Zbiory wzorców indukowane są iteracyjnie, od najbardziej ogólnych wzorców do najbardziej specyficznych, a podczas każdej iteracji zbiór odkrytych wzorców jest używany jako zestaw cech przez algorytm klasyfikacji. Jakość wzorców jest oceniana w odniesieniu do zadania klasyfikacji, a co za tym idzie, w odniesieniu do efektywności wyindukowanego modelu klasyfikacyjnego ocenianej odpowiednią miarą, a informacja o jakości jest używana w fazie odkrywania wzorców w kolejnej iteracji.

wania *optymalnych reguł* [46], a zatem stosuje dwie miary jakości, ω_1 i ω_2. Pierwsza z nich jest używana do bezpośredniego odcinania słabych jakościowo wzorców, a druga jest używana do dodatkowego odcięcia przestrzeni przeszukiwania przez zaprzestanie uszczegóławiania wybranych wzorców. Dlatego też algorytm używa dwóch progów jakościowych jakie muszą spełnić ostatecznie wybrane wzorce. Aby zastosować metodykę odkrywania optymalnych reguł, musimy wybrać opti-monotoniczną miarę jakości [6] jako ω_2. W konfiguracji algorytmu Fr-ONT-Qu opisanej w [39], jako pierwsza z miar ω_1 została użyta miara *lift*, a jako druga ω_2 miara pokrycia (lokalnego wsparcia).

Operator uszczegóławiania algorytmu Fr-ONT-Qu wykorzystuje strukturę drzewa poszukiwań przechowującego w wierzchołkach fragmenty kluczy. Przykład takiego drzewa jest pokazany na Rysunku 5.

Punktem startowym algorytmu jest bazowe zapytanie Q_{base}, gdzie $(body(Q_{base}), \mathbb{V}_{base}) \in basepatterns$. Oznaczmy przez T drzewo poszukiwań, przez Q zapytanie, a przez $last(Q)$ ostatnie wyrażenie w ciele zapytania $body(Q)$. Wzorce trójek lub wyrażenia typu FILTER są dodawane do drzewa T na jeden z poniższych sposobów, które definiuje operator uszczegóławiania, przy czym niektóre z tych grup reguł zawierają po kilka reguł rozszerzania wzorca, które zostały szczegółowo opisane w monografii (zob. także [39]):

1. *wyrażenia zależne od składni* (które posiadają wspólną zmienną $?x$ z $last(Q)$, która była nowa w $last(Q)$ lub, w przypadku gdy T jest puste, $?x \in \mathbb{V}$, gdzie $\mathbb{V}$ jest zbiorem zmiennych z wzorca bazowego),
2. *wyrażenia zależne od semantyki* (uszczegółowienia, które wykorzystują hierarchie klas i własności),
3. *uszczegółowienia wyrażeń typu FILTER*,
4. *prawi sąsiedzi* danego wierzchołka w T (tzn. kopie wyrażeń, które mają takiego samego rodzica



Rysunek 5. Przykład struktury drzewa poszukiwań algorytmu Fr-ONT-Qu. Każdy wierzchołek w drzewie jest etykietowany wzorcem trójkowym lub wyrażeniem typu FILTER. Każda ścieżka od korzenia drzewa do innego wierzchołka reprezentuje zapytanie, które stanowi fragment klucza związany z tym wierzchołkiem. Symbole na krawędziach wskazują zastosowanie odpowiednich typów reguł uszczegóławiania wzorca jakie obejmuje operator uszczegóławiania algorytmu Fr-ONT-Qu.

jak dane wyrażenie i które są umieszczone po prawej stronie listy jego dzieci, nowe zmienne zostają przemianowane w taki sposób aby były nadal nowe w kopii),

5. *kopia last*(Q), dodawana jeśli *last*(Q) jest wzorcem trójkowym, zachowując maksymalną liczbę dozwolonych kopii i odpowiednio przemianowując zmienne.

Dowody własności operatora ρ, (lokalnej) skończoności i zupełności, można znaleźć w [39].

W ramach przeprowadzonych eksperymentów obliczeniowych wykazaliśmy, że algorytm jest lepszy m.in. niż podejścia do klasyfikacji danych semantycznych, które zostały opisane w publikacjach na wiodących konferencjach w zakresie *Semantic Web* nagrodzonych jako najlepsze publikacje (ESWC 2008, ESWC 2012). Szczegółowe wyniki zostały opisane w artykule [39].

Główny mój samodzielny wkład naukowy jakiego dotyczy ten rozdział to:

— sformułowanie problemu opartej na wzorcach klasyfikacji danych z uwzględnieniem semantyki RDFS (ρdf),

— opracowanie algorytmu Fr-ONT-Qu do odkrywania wzorców w celu klasyfikacji z uwzględnieniem semantyki RDFS (ρdf), w tym operatora specjalizacji i wewnętrznej struktury danych algorytmu,

— przeprowadzenie dowodów własności algorytmu Fr-ONT-Qu: (lokalnej) skończoności i zupełności.

4.3.7. Analiza skupień i metody oparte na podobieństwie

W rozdziale 6 monografii opisałam podstawowe zagadnienia dotyczące nienadzorowanych metod eksploracji danych, których głównym celem jest opis danych. Formalnie zdefiniowałam zadanie analizy skupień. Poświęciłam szczególną uwagę na zebranie i usystematyzowanie zagadnień oraz wyników prac naukowych dostępnych w literaturze przedmiotu związanych z mierzeniem podobieństwa w sposób uwzględniający semantykę zasobów wyrażoną za pomocą ontologii.

Omówiłam w szczególności kryteria jakie były zaproponowane w literaturze przedmiotu odnośnie semantycznych miar podobieństwa.

Część rozdziału została poświęcona wynikom moich prac naukowych dotyczących zagadnienia tzw. *semantycznych funkcji jądrowych*. W monografii opisano wyniki badań prezentowane na konferencjach, jednak dotąd nie opublikowane w archiwalnych wydawnictwach [23]. Moim wkładem naukowym w te prace jest koncepcja dwóch semantycznych funkcji jądrowych, które mogą służyć do pomiaru podobieństwa pojęć przedstawionych w logice deskrypcyjnej $\mathcal{EL}^{++}$. Jedna z tych funkcji opiera się na idei *splotowych funkcji jądrowych* (ang. *convolution kernels*), a druga na *grafowych funkcjach jądrowych*. Propozycje tych dwóch funkcji zostały następnie dopracowane we współautorskich pracach [23], głównie przed dr Józefowskiego, który podał także dowody własności zaproponowanych funkcji jądrowych.

Splotowa funkcja jądrowa opiera się na postaci normalnej pojęcia C, która jest złożoną strukturą. Ogólną metodykę konstruowania funkcji jądrowych dla danych strukturalnych, *splotowe funkcje jądrowe*, wprowadził Haussler [16]. W tej metodyce przykłady mają postać złożonych struktur i istnieje relacja *dekompozycji* $\mathcal{R}$ określająca sposób dekompozycji przykładu na jego części, która jest funkcją odwracalną. Relacja ta zależy od natury języka przykładów. Możliwe jest wiele sposobów dekompozycji każdego przykładu. Splotowe funkcje jądrowe są oparte na idei definiowania funkcji jądrowych na częściach zdekomponowanych, strukturalnych obiektów, a następnie obliczania agregacji. Można zauważyć, że pojęcia w logice $\mathcal{EL}^{++}$ w normalnej postaci mają strukturę, która składa się z trzech podstawowych części: części, na którą składają się pojęcia pierwotne i nominalne, części, na którą składają się wyrażenia z rolami abstrakcyjnymi oraz części, na którą składają się wyrażenia z konkretnymi rolami. Możemy również założyć, bez utraty ogólności, że pojęcie uniwersalne $\top$ i pojęcie "najniższe" $\bot$ należą do części, na którą składają się pojęcia pierwotne i nominalne. Zatem każde pojęcie w języku $\mathcal{EL}^{++}$ w normalnej postaci składa się z co najwyżej trzech różnych podstawowych części. Wprowadzamy specjalne słowo $EMPTY$, nie należące do języka $\mathcal{EL}^{++}$. Użyjemy słowa $EMPTY$ aby reprezentować dowolną z trzech części, jeśli jest pusta. Taka trójka jest złożoną strukturą i można dla niej zdefiniować relację dekompozycji $\mathcal{R}$. Następnie dla każej części definiujemy odpowiednie funkcje jądrowe (np. oparte o przecięcia zbiorów dla części dotyczącej pojęć pierwotnych dwóch pojęć C_1 i C_2 lub o kolejną funkcję splotową dla części, na którą składają się wyrażenia z rolami abstrakcyjnymi) i stosujemy to podejście rekurencyjnie.

Grafowa funkcja jądrowa opiera się na podobnej strukturze, jaka była wykorzystywana w algorytmie Fr-ONT w ramach jego operatora uogólnienia, tzn. drzewie pojęcia. Funkcja jądrowa uwzględnia zarówno strukturę drzewa jak i semantykę jego wierzchołków i krawędzi. Funkcja jądrowa w dużej mierze opiera się na funkcji jądrowej w postaci losowego spaceru zaproponowanej przez Gaertnera i innych [12] i jej modyfikacji zaproponowanej przez Borgwardta i innych [5]. Mając dane dwa etykietowane grafy G_1 i G_2, funkcja jądrowa w postaci losowego spaceru zlicza liczbę pasujących do siebie etykietowanych losowych spacerów. Dopasowanie dwóch wierzchołków lub dwóch krawędzi w ramach losowego spaceru jest określane poprzez porównanie ich atrybutów. Wartość funkcji jądrowej opartej o spacer dla dwóch spacerów $walk_{G_1}$ i $walk_{G_2}$, w dwóch grafach G_1 i G_2 jest obliczana jako iloczyn wartości funcji jądrowych odpowiadających wierzchołkom i

krawędziom odwiedzonym podczas spaceru. Wartość funkcji jądrowej opartej o spacer losowy dla dwóch grafów G_1 i G_2 jest sumą wartości funkcji wszystkich par spacerów w obrębie tych grafów. Jednak taka funkcja nie jest jeszcze wystarczająca aby porównać dwa pojęcia $\mathcal{EL}^{++}$ ponieważ atrybuty dwóch wierzchołków v_1 w grafie G_1 i v_2 w grafie G_2 są uważane za podobne, jeśli są zupełnie identyczne, co jest rzadkością w przypadku drzew pojęć dwóch pojęć $\mathcal{EL}^{++}$. Dlatego też zaproponowaliśmy podobną redefinicję funkcji jądrowej opartej na losowym spacerze do tej, która została zaprezentowana w [5], gdzie funkcja dla każdego kroku losowego spaceru jest iloczynem wartości funkcji dla początkowego wierzchołka, dla docelowego wierzchołka i krawędzi między nimi. Zmodyfikowana funkcja jądrowa oparta na spacerze losowym dla dwóch drzew T_1 i T_2 jest sumą wszystkich spacerów losowych dla par spacerów w obrębie drzew. Dla zdefiniowania funkcji jądrowej związanej z pojedynczym krokiem spaceru użyliśmy trzech ogólnych typów funkcji: funkcji jądrowej typu, funkcji jądrowej etykiet wierzchołków oraz funkcji jądrowej etykiet krawędzi. Funkcja jądrowa typu jest używana w celu zapewnienia porównywania tylko wierzchołków i krawędzi tego samego typu. Rozróżniamy cztery podstawowe typy. Pierwsze dwa to typy dotyczące wierzchołków drzewa: wierzchołków oznaczonych zbiorem nazw pojęć pierwotnych występujących w C i wierzchołków oznaczonych etykietą będącą wartością konkretnego typu danych f. Kolejne dwa typy to typy dotyczące krawędzi drzewa: oznaczonych poprzez nazwę abstrakcyjnej roli R i oznaczonych przez nazwę konkretnej roli P. Funkcje jądrowe w ramach tych ogólnych typów mogą być formułowane na różne sposoby, np. wykorzystując funkcję jądrową bazującą na przecięciu zbiorów (dla zbiorów nazw pojęć pierwotnych) albo wykorzystując funkcję jądrową dla drzew (dla hierarchii ról). Szczegóły takich sformułowań zostały podane w monografii.

Główny mój wkład naukowy jakiego dotyczy ten rozdział to:

— koncepcja dwóch funkcji jądrowych dla pojęć reprezentowanych w logice deskrypcyjnej $\mathcal{EL}^{++}$: splotowej oraz grafowej funkcji jądrowej.

4.3.8. Eksploracja danych przy założeniu "otwartego świata"

W rozdziale 7 monografii omówiłam specyficzne problemy wynikające z przyjęcia założenia *otwartego świata* jakie zazwyczaj jest czynione podczas wnioskowania ontologicznego i ich wpływ na eksplorację danych z wykorzystaniem ontologii. Jednym z głównych problemów jest generowanie przykładów negatywnych, ponieważ negacja faktów musi zostać jawnie wyrażona, a brak informacji nie jest traktowany domyślnie jako informacja negatywna. Zebrałam i omówiłam różne rozwiązania proponowane w literaturze w celu radzenia sobie z przedstawionymi problemami podczas eksploracji danych:

— alternatywne sformułowanie problemu ('zamykanie' bazy wiedzy przez umożliwienie wnioskowania o przynależności indywiduum do klasy przy założeniu *zamkniętego świata*),

— operator epistemiczny **K** (parafrazowany jako "znany"), co pozwala na wykonywanie zapytań o 'znane' przez bazę wiedzy własności 'znanych' przez bazę wiedzy indywiduów, oraz

— nowe miary oceny (uwzględniające niekompletność baz wiedzy).

Mój wkład w tę tematykę to współautorska praca (wraz z Prof. Rossem Kingiem) prezentowana na konferencji [34].

4.3.9. Rozszerzanie grafów wiedzy

Rozdział 8 monografii dotyczy problemu aktualizowania grafów wiedzy. Rozdział ten opiera się po części na artykule konferencyjnym autorstwa Morzego, Ławrynowicz i Zozulińskiego [49]. Główną uwagę poświęcono rozszerzaniu grafów o nową wiedzę. Szczegółowo omówione zostało zadanie odkrywania synonimów relacji w grafie wiedzy. Do realizacji tego zadania wykorzystano metodę odkrywania tzw. *zbiorów substytutywnych* wprowadzoną w wymienionej wyżej publikacji konferencyjnej [49], przy czym sama koncepcja zbiorów substytutywnych wraz z formalizacją jest autorstwa Mikołaja Morzego.

Termin "graf wiedzy" został spopularyzowany przez Google w 2012 r. wraz z rozwojem ich Grafu Wiedzy, zaprojektowanego do celów wyszukiwania semantycznego. Termin ten jest obecnie używany w odniesieniu do innych baz wiedzy zarówno komercyjnych, jak i zakorzenionych w środowiskach akademickich i projektach wspólnotowych, a przede wszystkim w otwartej domenie. Istnieje kilka godnych uwagi, publicznie dostępnych grafów wiedzy, w tym: DBpedia [1], OpenCyc [45], Wikidata [65], YAGO [18, 59] i NELL [48]. Istnieje również kilka znaczących komercyjnych grafów wiedzy, w tym Graf Wiedzy Google, Satori Microsoftu, Graf Encji Facebooka oraz Graf Wiedzy Yahoo!.

Podstawowymi składowymi grafu wiedzy są: *encje*, wyrażone za pomocą wierzchołków grafu, ich *własności* (atrybuty) i *relacje* łączące wierzchołki, wyrażone przez krawędzie w grafie. Można zauważyć, że atrybuty są odpowiednikiem konkretnych ról w logikach deskrypcyjnych a relacje są odpowiednikiem ról abstrakcyjnych. Encje mogą mieć (semantyczne) typy, co reprezentuje ogólna relacja *is-a* między encją a jej typem. Możliwe jest również, że niektóre typy, własności i relacje zawarte w grafie wiedzy są ustrukturalizowane w ontologii lub schemacie danych (TBoxie). Ta ontologia (lub schemat) posiada najczęściej niewielką ekspresywność, a stopień jej aksjomatyzacji jest niski. Z kolei grafy wiedzy koncentrują się na faktach (ABoxie), a liczba instancji w typowym grafie wiedzy jest ogromna. Grafy wiedzy są często reprezentowane w języku RDF.

Zarówno liczba jak i rozmiar grafów wiedzy wzrosły w ciągu ostatnich kilku lat. Ponadto liczba linków i odwzorowań między grafami wiedzy wzrosła w znacznym stopniu, szczególnie w odniesieniu do tych opublikowanych w obrębie Otwartych Powiązanych Danych (`http://lod-cloud.net`) [4]. Ze względu na ich liczbę i rozmiary, a w znacznym stopniu automatyczny sposób ich budowy, problem zapewnienia jakości grafów wiedzy stał się poważnym problemem. Jedna z głównych kwestii jakościowych dotyczy nadmiarowości w grafach wiedzy: identyczne encje świata rzeczywistego są opisywane przy użyciu różnego słownictwa z różnych przestrzeni nazw i do oznaczania równoważnych klas i własności używanych jest wiele synonimów, nawet w obrębie pojedynczego grafu wiedzy. Dlatego też znalezienie powiązań między odpowiadającymi sobie lub podobnymi encjami jest kluczowym zadaniem w zakresie utrzymania i rozwoju grafu wiedzy.

W rozdziale 8 monografii, opisuję zatem swoją metodę znajdowania synonimów relacji w grafie wiedzy, opartą o koncepcję zbiorów substytutywnych. Zadanie odkrywania zbiorów substytutywnych polega na znajdowaniu równoważnych elementów, tj. par elementów, które mogą być używane zamiennie w różnych kontekstach. Odkrywanie substytutywnych zbiorów wpisuje się w szersze zagadnienie odkrywania asocjacji i odbywa się w dwóch krokach: odkrywanie częstych zbiorów w transakcyjnej bazie danych oraz generowanie zbiorów substytutywnych na bazie wygenerowanych

częstych zbiorów. Zbiory substytutywne generowane są w oparciu o tzw. *zbiory pokrywające*. Dla każdego częstego elementu, jego zbiór pokrywający jest kolekcją wszystkich częstych zbiorów odkrytych w transakcyjnej bazie danych, z którymi dany element tworzy częsty zbiór. Zbiory substytutywne są zbiorami spełniającymi dwa kryteria. Po pierwsze, zbiory pokrywające danych elementów x i y muszą mieć zdefiniowany przez użytkownika minimalny procent wspólnych elementów, co gwarantuje, że pojawiają się one w podobnych kontekstach niezależnie od siebie. Drugim kryterium jest ograniczenie współwystępowania x i y poprzez próg maksymalnego współwystępowania, zdefiniowany przez użytkownika, który służy do odcinania par elementów, które są funkcjonalnie zależne.

Aby wykorzystać ideę zbiorów substytutywnych do odkrywania synonimów relacji w grafie wiedzy, baza danych transakcji jest tworzona w taki sposób, że każda transakcja składa się z trzech elementów i ma postać $\{C_1, P, C_2\}$, gdzie C_1 i C_2 są klasami, odpowiednio, podmiotu s i dopełnienia o, instancji trójki RDF z grafu DBpedii, a P jest relacją łączącą s i o. W celu rozróżnienia pomiędzy podmiotami i dopełnieniami w trójce w wygenerowanych transakcjach, URI klas zostały opatrzone odpowiednim prefiksem.

W rozdziale opisano także nieopublikowane wcześniej przeze mnie wyniki ewaluacji odkrywania własności substytutywnych w grafach wiedzy wraz z ich dyskusją. Eksperymenty zostały przeprowadzone na grafie wiedzy DBpedia. Do ewaluacji wykorzystałam platformę *crowdsourcingową* CrowdFlower (`https://www.crowdflower.com`).

Główny mój samodzielny wkład opisany w rozdziale 8 monografii to:
— opracowanie metody rozszerzania grafu wiedzy poprzez znajdowanie synonimów relacji,
— ewaluacja metody na platformie CrowdFlower.

4.3.10. Semantyczna kategoryzacja wyników zapytań do baz wiedzy

W rozdziale 9 przedstawiona została koncepcja *semantycznej kategoryzacji* wyników zapytań do baz wiedzy oraz algorytmy realizujące tę koncepcję wraz z ich ewaluacją.

Semantyczna kategoryzacja wyników to takie grupowanie wyników zapytania, które bierze pod uwagę ich semantykę wyrażoną w ontologii dziedzinowej. Rozważono dwa rodzaje semantycznej kategoryzacji: dedukcyjną i indukcyjną. W pierwszym przypadku, wyniki są pogrupowane za pomocą wnioskowania dedukcyjnego z uwzględnieniem hierarchii subsumcji pojęć wywodzącej się z bazy wiedzy. W drugim przypadku, wyniki są pogrupowane indukcyjnie, za pomocą analizy skupień i w oparciu o podobieństwo poszczególnych zasobów.

Za oryginalny dorobek naukowy prowadzonych przeze mnie badań opisanych w tym rozdziale uważam po pierwsze samą koncepcję semantycznej kategoryzacji, która została przeze mnie zaprezentowana po raz pierwszy w artykule konferencyjnym [29]. W artykule zaproponowałam także algorytm semantycznej kategoryzacji, wykorzystujący grupowanie konceptualne (konceptualną analizę skupień) oraz przedstawiłam wstępną ewaluację zaproponowanego algorytmu. Następnie, temat ten był przeze mnie kontynuowany w artykule [30], w którym przedstawiłam propozycję rozszerzenia języka SPARQL o klauzulę CLUSTER BY umożliwiającą deklaratywne wywoływanie dynamicznego grupowania wyników zapytań. Zaproponowałam składnię i semantykę takiego rozszerzenia.

Prace te były następnie kontynuowane przeze mnie we współpracy z dr Claudią d'Amato i dr Nicolą Fanizzi, po części w ramach mojej wizyty naukowej na Uniwersytecie w Bari (Włochy) w 2009 roku jak i po wizycie. W ramach tej współpracy opracowaliśmy metodę dedukcyjnego grupowania wyników, która mając dane zapytanie koniunkcyjne i ontologię dziedzinową reprezentowaną w OWL, zwraca dynamiczną kategoryzację wyników zapytania [7]. Podczas generowania kategoryzacji, metoda wykorzystuje semantykę bazy wiedzy (wyrażoną w ontologii) przez zastosowanie wnioskowania dedukcyjnego. W szczególności, biorąc pod uwagę kryterium grupowania wyrażone jako (złożone) pojęcie z bazy wiedzy, wyniki są pogrupowane według (części) wywnioskowanej hierarchii subsumcji pojęć. W efekcie umożliwiamy nawigację po wynikach i pokazywanie ich w podobny sposób jak podczas wyszukiwania fasetowego. Zaproponowana metoda działa w czterech zasadniczych krokach:

1. wywnioskowanie typu związanego z każdą zmienną, na podstawie zapytania oraz ontologii,
2. konstruowanie hierarchii pojęć dla pojedynczej zmiennej,
3. obliczanie iloczynu drzew pojedynczych hierarchii pojęć dla wielu zmiennych,
4. populowanie hierarchii wynikami zapytania.

W ramach kontynuacji powyższych badań, zaproponowałam algorytm opierający się na operatorze uszczegóławiania i statystycznej analizie wyników zapytania. Algorytm ten rozszerza zaproponowaną metodę grupowania dedukcyjnego w przypadkach gdy jako kryterium grupowania użyte są pojęcia będące liśćmi w hierarchii subsumcji oraz gdy kryterium grupowania jest pojęciem uniwersalnym (najbardziej ogólnym). Algorytm ten został oryginalnie opisany w publikacji konferencyjnej autorstwa Ławrynowicz, d'Amato i Fanizzi [32].

Na bazie rezultatów tych badań został utworzony prototyp systemu o nazwie ASPARAGUS, który został opisany w publikacji konferencyjnej [40].

W rozdziale 9 przedstawiłam także krótką (wcześniej nie opublikowaną) ewaluację metody indukcyjnego semantycznego grupowania.

Główny mój samodzielny wkład opisany w tym rozdziale to:

— opracowanie koncepcji semantycznej kategoryzacji wyników zapytań do baz wiedzy,
— rozszerzenie języka SPARQL o klauzulę CLUSTER BY,
— algorytm indukcyjnej semantycznej kategoryzacji wyników zapytań opierający się na konceptualnej analizie skupień,
— rozszerzenie algorytmu dedukcyjnej semantycznej kategoryzacji wyników zapytań o operator uszczegóławiania i statystyczną analizę wyników zapytania.

4.3.11. Semantyczna meta-eksploracja danych

Rozdział 10 monografii dotyczy zagadnienia *semantycznej meta-eksploracji danych* [17]. Rozdział ten jest oparty na dwóch artykułach opublikowanych w czasopismach z listy JCR [25, 39], gdzie występuję w roli autora korespondującego (razem z dr Marią Keet w pierwszym przypadku i Jędrzejem Potońcem w drugim przypadku). Przedstawione wyniki prac są w dużej mierze efektem pracy naukowej wykonanej w ramach projektu europejskiego z 7Pr o nazwie "e-LICO: An e-Laboratory for Interdisciplinary Collaborative Research in Data Mining and Data-Intensive

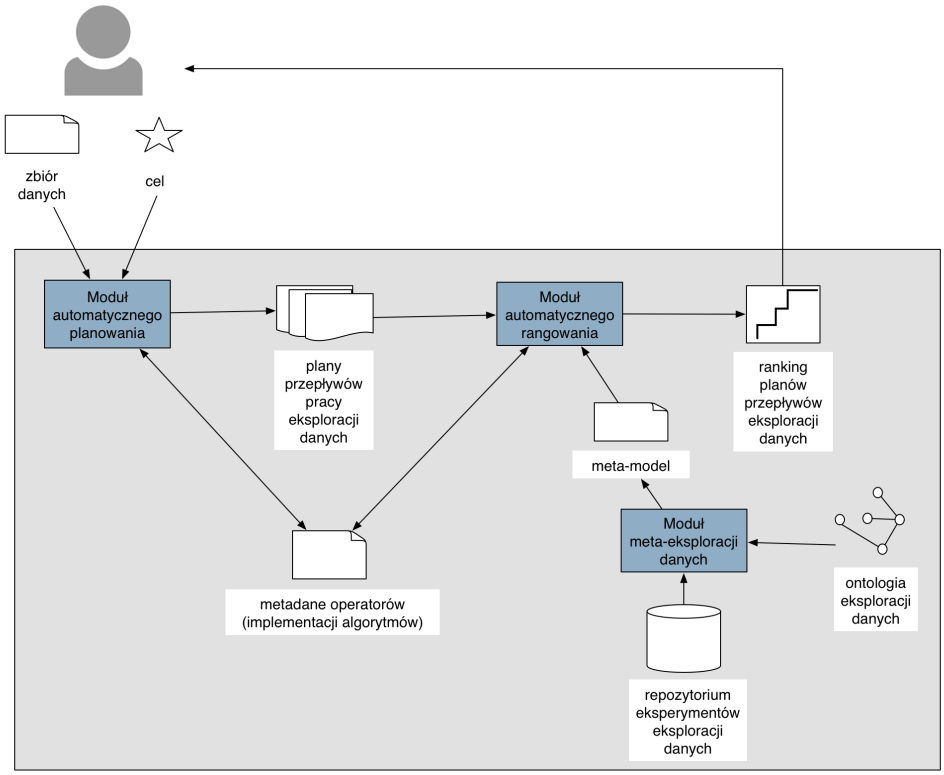
Science" (2009-2012, ICT-2007.4.4 231519), gdzie pełniłam rolę kierownika merytorycznego zespołu z Wydziału Informatyki Politechniki Poznańskiej.

W rozdziale przedstawiłam zagadnienie semantycznej meta-eksploracji danych, jako specjalnego rodzaju *meta-uczenia*. Meta-uczenie [21] w informatyce jest definiowane jako aplikowanie technik uczenia maszynowego na meta-danych przeprowadzonych już eksperymentów uczenia maszynowego w celu modyfikacji wybranych aspektów procesu uczenia aby poprawić rezultaty procesu (tj. efektywność wyprodukowanego modelu uczącego). Semantyczna meta-eksploracja danych wykracza poza konwencjonalne meta-uczenie się na trzy następujące sposoby. Po pierwsze rozszerza podejście meta-uczenia do uczenia się z pełnego procesu eksploracji danych, tj. do *meta-eksploracji*, która uwzględnia współzależności i interakcje między poszczególnymi operacjami. Po drugie, w przeciwieństwie do tradycyjnego meta-uczenia opartego na danych, jest ona mocno sterowana wiedzą na temat procesu eksploracji danych i jego składowych, wyrażoną w ontologii eksploracji danych i bazie wiedzy. Po trzecie, różne cechy wewnętrzne algorytmów eksploracji danych (takie jak funkcje kosztu, strategie optymalizacji, struktura modelu i inne) są jawnie reprezentowane i analizowane w celu skorelowania skuteczności uczonych modeli wytwarzanych przez procesy eksploracji danych zarówno z charakterystyką danych, jak i z charakterystykami algorytmu.

Następnie, w tym rozdziale, opisałam ontologię dziedzinową *Data Mining OPtimization Ontology*, której jestem współtwórczynią. Prace nad ontologią rozpoczęły się wraz z początkiem projektu e-LICO w zespole kierowanym przez dr Melanie Hilario na Uniwersytecie Genewskim (Szwajcaria). Brałam udział w definiowaniu podstawowej struktury, modelowaniu wybranych gałęzi ontologii oraz czuwałam nad aspektami metodologicznymi dotyczącymi procesu inżynierii wiedzy.

Prace nad ontologicznym modelowaniem dziedziny eksploracji danych kontynuowałam także po zakończeniu projektu e-LICO, inicjując działania zmierzające do standaryzacji modelowania metadanych tej dziedziny. W 2015 roku, z mojej inicjatywy została utworzona grupa robocza, której celem jest standaryzacja schematów metadanych eksperymentów uczenia maszynowego *Machine Learning Schema Community Group* pod auspicjami W3C, w której objęłam rolę współprzewodniczącej. W ramach prac grupy opracowaliśmy raport z rekomendacją schematu metadanych dotyczącego typowych eksperymentów uczenia maszynowego [9]. W rozdziale opisałam też nieopublikowane wcześniej odwzorowania ontologii DMOP do innych schematów w tym OPMW, PROV-O i P-plan i *Research Objects* (szczegóły można znaleźć w monografii).

Meta-uczenie ma dostarczyć wiedzy na temat tego, jakie algorytmy najlepiej nadają się do celów danej analizy danych. Aktualnie dostępne platformy eksploracji danych oferują wiele implementacji algorytmów wspomagających różne etapy procesu eksploracji, np. wstępne przetwarzanie danych lub indukcję modelu predykcyjnego. Na przykład środowisko RapidMiner (`https://rapidminer.com/`) oferuje kilkaset implementacji algorytmów (nazywanych operatorami). Użytkownik takiego systemu ma do wyboru wiele operatorów i ich kombinacji w celu skonstruowania najlepszego przepływu pracy (*workflowu*) adresującego jego cel analizy. W związku z tym, zaproponowano różne systemy wspomagające użytkownika w tym procesie, w postaci np. inteligentnych asystentów eksploracji danych [61]. Jedna z proponowanych architektur takiego



Rysunek 6. Architektura systemów typu inteligentny asystent eksploracji danych opartych na automatycznym planowaniu i meta-eksploracji danych.

asystenta to architektura oparta na metodach sztucznej inteligencji realizujących automatyczne planowanie, które służą do skonstruowania zbioru poprawnych planów przepływów pracy (szablonów przepływów pracy), spełniających wymagania postawione przez konkretny cel analizy danych użytkownika oraz uwzględniających charakterystykę zbioru danych użytkownika. W planach przepływów operatory połączone są w taki sposób, że wszystkie ich warunki wstępne i warunki dotyczące rezultatu wyjściowego są spełnione. Chociaż wszystkie plany przepływu pracy generowane przez system automatycznego planowania są *poprawne* w kontekście danego zadania, może być ich bardzo wiele. Dlatego zanim wynikowa lista przepływów pracy zostanie przedstawiona użytkownikowi, potrzebny jest dodatkowy krok, polegający na wygenerowaniu rankingu przepływów wedle przyjętej miary (np. trafności klasyfikacji, gdy docelowe zadanie użytkownika polega na klasyfikacji), aby zarekomendować użytkownikowi *optymalny* przepływ. Krok ten może być wsparty semantycznym meta-modelem predykcyjnym, wyindukowanym za pomocą semantycznej meta-eksploracji repozytorium wykonanych już procesów uczenia maszynowego. Hilario i inni [17] zaproponowali rozszerzenie opisanej powyżej architektury opartej na metodach sztucznej inteligencji realizujących automatyczne planowanie o moduł semantycznej meta-eksploracji danych. Rozszerzoną architekturę przedstawiono na Rysunku 6.

W rozdziale opisałam opracowaną przeze mnie metodę wykorzystania algorytmu Fr-ONT-Qu do celów semantycznej meta-eksploracji danych, która wpisuje się we wspomnianą powyżej architekturę. Metoda ta została następnie zaimplementowana wspólnie z Jędrzejem Potońcem i została przeprowadzona jej ewaluacja. Dane do ewaluacji metody zostały przygotowane w następujący sposób. W pierwszym kroku, za pomocą automatycznego planowania, zostały wygenerowane plany eksperymentów uczenia maszynowego w postaci przepływów pracy RapidMinera i wykorzystując popularne zbiory danych. Następnie zostały wykonane eksperymenty i zarejestrowana efektywność wytworzonych modeli predykcyjnych. Eksperymenty zostały formalnie opisane z wykorzystaniem ontologii DMOP i zostało utworzone repozytorium eksperymentów w postaci grafu wiedzy (po transformacji z formatu przepływów pracy RapidMinera do postaci trójek RDF). Na tak utworzonym grafie wiedzy można było zastosować algorytm Fr-ONT-Qu, który wykrywał wzorce we wzbogaconych semantycznie metadanych przepływów pracy. Wzorce miały postać zapytań SPARQL, zawierających encje z ontologii DMOP. Wzorce zostały wykorzystane jako złożone cechy w zadaniu klasyfikacji, którego celem było zaklasyfikowanie danego planu przeływu pracy jako produkującego model predykcyjny o dobrej bądź też złej jakości. Na podstawie takich klasyfikacji można było wygenerować ranking przepływów pracy. Przeprowadzone eksperymenty potwierdziły efektywność zaproponowanego podejścia.

Zarówno sama metoda, jej implementacja jak i wyniki eksperymentów, które zostały przytoczone w monografii w rozdziale 10, opisane zostały oryginalnie w artykule opublikowanym w czasopiśmie z listy JCR [39].

Główny mój samodzielny wkład badawczy opisany w tym rozdziale to:

— opracowanie konceptualizacji w ontologii DMOP obszaru dotyczącego odkrywania wzorców,
— opracowanie odwzorowań ontologii DMOP na ontologie i schematy OPMW, PROV-O, oraz ML-Schema,
— opracowanie metody semantycznej meta-eksploracji danych z wykorzystaniem algorytmu Fr-ONT-Qu.

5. Omówienie pozostałych osiągnięć naukowo-badawczych

Całość mojego dorobku wykracza poza wyniki opisane w monografii i plasuje się w obszarze szeroko rozumianej inżynierii wiedzy. Pozostałe publikacje ukazujące się od roku 2004 (dorobek po doktoracie obejmuje prace opublikowane po roku 2008) obejmują tematykę eksploracji złożonych danych w tym indukcji programów w logice, analizy sieci semantyczych i ontologii, inżynierię ontologii oraz wzorców projektowych dla ontologii oraz metodologię inżynierii ontologii. Poniżej omówione są najważniejsze osiągnięcia.

5.1. Eksploracja złożonych danych i indukcja ontologii

W ramach pracy doktorskiej opracowałam metodę odkrywania częstych wzorców w bazach wiedzy reprezentowanych w logice deskrypcyjnej z regułami. Badania te zostały opisane w publikacjach konferencyjnych oraz w artykule opublikowanym w czasopiśmie z listy JCR [22].

Eksploracji złożonych ontologii dotyczą badania opisane w artykule znajdującym się w druku w czasopiśmie z listy JCR [41]. Badania te zostały wykonane we współpracy z grupą Protégé z Centrum Badawczego Informatyki Biomedycznej Uniwersytetu Stanforda (USA) w ramach projektu "LeoLOD - Learning and Evolving Ontologies from Linked Open Data" (2013-2015) o numerze POMOST/2013-7/8 finansowanego przez Fundację na Rzecz Nauki Polskiej, którego byłam kierownikiem. W związku z prowadzonymi wspólnymi badaniami, odbyłam wizytę naukową w grupie Protégé w lutym 2015r. Praca [41] jest współautorską pracą, której jestem pierwszym autorem i autorem korespondującym, a pozostali autorzy to Potoniec, Robaczyk oraz dr Tudorache (z grupy Protégé). Celem badań było zbadanie wzorców modelowania aksjomatów, które powtarzają się w ontologiach. Takie wzorce mogą wywodzić się z niektórych rozwiązań projektowych i mogą wskazywać na pojawiające się wzorce projektowe ontologii. Wynikiem pracy jest metoda bazująca na eksploracji drzew mająca na celu identyfikację ujawniających się wzorców projektowych. Metoda obejmuje dwa etapy: (1) przekształcenie aksjomatów ontologii do postaci drzew w celu znalezienia wzorców aksjomatów, a następnie (2) użycia analizy asocjacji do odkrycia współwystępujących wzorców aksjomatów, w celu wyodrębnienia pojawiających się wzorców projektowych ontologii. W ramach pracy przeprowadziliśmy eksperymenty na repozytorium ontologii o nazwie BioPortal. Pokazaliśmy, że powtarzające się wzorce aksjomatów pojawiają się we wszystkich indywidualnych ontologiach, a także w całym zbiorze. W indywidualnych ontologiach znajdujemy częste i nietrywialne wzorce z i bez zmiennych. Niektóre z poprzednich wzorców mają ponad 300 000 wystąpień. Udowodniliśmy także, że jesteśmy w stanie automatycznie wykryć wzorce, dla których ręcznie potwierdziliśmy, że są to fragmenty wzorców projektowych ontologii opisanych w literaturze.

W ramach projektu LeoLOD, opracowaliśmy też metody i narzędzia służące do indukcji ontologii z powiązanych danych w grafach RDF. W artykule zgłoszonym do czasopisma JCR, który jest w druku (autorstwa Potońca, Jakubowskiego i Ławrynowicz) opisaliśmy metodę indukowania złożonych klas reprezentowanych w języku OWL 2 EL za pomocą zapytań do końcowek SPARQL [54].

Innymi pracami poświęconymi tej tematyce są m.in. publikacje konferencyjne: autorstwa Potońca i Ławrynowicz [55] na temat łączenia generowania złożonych klas z modelowaniem matematycznym w celu uczenia się ontologii oraz autorstwa Kowalczuk, Potońca i Ławrynowicz dotycząca ekstrakcji wzorców użycia ontologii w sieci Web na przykładzie słownictwa ontologii GoodRelations [27].

5.2. Inżynieria ontologii i wzorców projektowych dla ontologii

Jestem współautorem kilku ontologii oraz wzorców projektowych ontologii, poza wymienionymi wcześniej w autoreferacie ontologiami i schematami w dziedzinie eksploracji danych (DMOP, ML-Schema):

— ontologii *Digital Multimedia Repositories Ontology*, razem z dr Palmą [37],

— wzorca projektowego ontologii dotyczącego modelowania sytuacji zagrożenia [35] oraz ontologii

inżynierii bezpieczeństwa *Occupational Safety and Health Domain Ontology* [36] razem z Ławniczak,

— rozszerzenia zasobu FrameNet o opis dziedziny uczenia maszynowego [20] z Jakubowskim,

— wzorca projektowego do modelowania raportów zdarzeń [26], razem z Kowalczuk (nagroda za najlepszy poster).

Przy moim udziale powstały także narzędzia służące do inżynierii ontologii takie jak np. wtyczki i rozszerzenia edytora ontologii Protégé, służące do indukcji ontologii z grafów RDF [56, 62], oraz narzędzie generujące treść systemów typu semantyczne Wiki z ontologii dziedzinowych [11].

5.3. Metodologia inżynierii ontologii

Główny mój wkład naukowy w metodologię inżynierii ontologii jest związany z pracami naukowymi wykonywanymi w ramach projektu finansowanego przez Narodowe Centrum Nauki o numerze 2014/13/D/ST6/02076 i o tytule "ARISTOTELES: Metodologia i algorytmy automatycznej aktualizacji ontologii w scenariuszach zadaniowych" (2015-2018), którego jestem kierownikiem. Wraz z dr Keet z Uniwersytetu Kapsztadzkiego (RPA), która jest członkiem zespołu projektowego, opracowałyśmy metodykę inżynierii ontologii sterowanej testami (ang. *Test-Driven Development*) [24]. W ramach realizacji prac badawczych dr Keet odbyła wizytę na Wydziale Informatyki Politechniki Poznańskiej w 2015 roku. Praktycznym rezultatem prac jest także narzędzie TDDOnto, którego pierwsza wersja została zaimplementowana przeze mnie, natomiast druga wersja (TDDOnto2) przez Kierena Daviesa, znajdującego się pod opieką naukową dr Keet w ramach drugiego etapu jego studiów. Obie wersje zostały opisane w publikacjach na warsztatach naukowych [33] i na sesji demonstracyjnej konferencji ESWC2017, wraz z opisem przeprowadzonych eksperymentów.

Literatura

[1] Sören Auer, Christian Bizer, Georgi Kobilarov, Jens Lehmann, Richard Cyganiak, Zachary G. Ives. DBpedia: A nucleus for a Web of open data. *The Semantic Web, 6th International Semantic Web Conference, 2nd Asian Semantic Web Conference, ISWC 2007 + ASWC 2007, Busan, Korea, November 11-15, 2007.*, wolumen 4825 of LNCS, strony 722–735. Springer, 2007.

[2] Franz Baader, Diego Calvanese, Deborah L. McGuinness, Daniele Nardi, Peter F. Patel-Schneider, redaktorzy. *The Description Logic Handbook: Theory, Implementation, and Applications*. Cambridge University Press, New York, NY, USA, 2003.

[3] Franz Baader, Ralf Molitor, Stephan Tobies. Tractable and decidable fragments of conceptual graphs. William M. Tepfenhart, Walling R. Cyre, redaktorzy, *ICCS*, wolumen 1640 serii *Lecture Notes in Computer Science*, strony 480–493. Springer, 1999.

[4] Christian Bizer, Tom Heath, Tim Berners-Lee. Linked Data - the story so far. *Int. J. Semantic Web Inf. Syst.*, 5(3):1–22, 2009.

[5] Karsten M. Borgwardt, Cheng Soon Ong, Stefan Schönauer, S. V. N. Vishwanathan, Alexander J. Smola, Hans-Peter Kriegel. Protein function prediction via graph kernels. *ISMB (Supplement of Bioinformatics)*, strony 47–56, 2005.

[6] Yannick Le Bras, Philippe Lenca, Stéphane Lallich. Optimonotone measures for optimal rule discovery. *Computational Intelligence*, 28(4):475–504, 2012.

[7] Claudia d'Amato, Nicola Fanizzi, Agnieszka Ławrynowicz. Categorize by: Deductive aggregation of semantic web query results. *The Semantic Web: Research and Applications, 7th Extended Semantic Web Conference, ESWC 2010, Heraklion, Crete, Greece, May 30 - June 3, 2010, Proceedings, Part I*, strony 91–105, 2010.

[8] Saso Dzeroski, Nada Lavrac, redaktorzy. *Relational Data Mining*. Springer, 2001.

[9] Diego Esteves, Agnieszka Ławrynowicz, Pance Panov, Larisa N. Soldatova, Tommaso Soru, Joaquin Vanschoren. ML Schema Core specification. Raport instytutowy, W3C, Październik 2016. http://www.w3.org/2016/10/mls/.

[10] Usama M. Fayyad, Gregory Piatetsky-Shapiro, Padhraic Smyth. Advances in knowledge discovery and data mining. rozdział From Data Mining to Knowledge Discovery: An Overview, strony 1–34. American Association for Artificial Intelligence, Menlo Park, CA, USA, 1996.

[11] Dominik Filipiak, Agnieszka Ławrynowicz. Generating semantic media wiki content from domain ontologies. *Proceedings of the Third International Workshop on Semantic Web Collaborative Spaces co-located with the 13th International Semantic Web Conference (ISWC 2014), Riva del Garda, Italy, October 19, 2014.*, 2014.

[12] Thomas Gaertner, Peter Flach, Stefan Wrobel. S.: On graph kernels: Hardness results and efficient alternatives. *In: Conference on Learning Theory*, strony 129–143, 2003.

[13] Ramanathan Guha, Dan Brickley. RDF vocabulary description language 1.0: RDF schema. W3C recommendation, W3C, Luty 2004. http://www.w3.org/TR/2004/REC-rdf-schema-20040210/.

[14] Ramanathan Guha, Dan Brickley. RDF schema 1.1. W3C recommendation, W3C, Luty 2014. http://www.w3.org/TR/2014/REC-rdf-schema-20140225/.

[15] Jiawei Han, Micheline Kamber, Jian Pei. *Data Mining: Concepts and Techniques*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, wydanie 3rd, 2011.

[16] David Haussler. Convolution kernels on discrete structures. Raport instytutowy, University of California at Santa Cruz, 1999.

[17] Melanie Hilario, Phong Nguyen, Huyen Do, Adam Woznica, Alexandros Kalousis. Ontology-based meta-mining of knowledge discovery workflows. *Meta-Learning in Computational Intelligence*, wolumen 358 serii *Studies in Computational Intelligence*, strony 273–315. Springer, 2011.

[18] Johannes Hoffart, Fabian M. Suchanek, Klaus Berberich, Gerhard Weikum. YAGO2: A spatially and temporally enhanced knowledge base from Wikipedia. *Artif. Intell.*, 194:28–61, 2013.

[19] Christopher J. Hogger. *Essentials of Logic Programming*. Oxford University Press, 1990.

[20] Piotr Jakubowski, Agnieszka Ławrynowicz. Extending FrameNet to machine learning domain. *Joint Proceedings of the 5th Workshop on Data Mining and Knowledge Discovery meets Linked Open Data and the 1st International Workshop on Completing and Debugging the Semantic Web (Know@LOD-2016, CoDeS-2016) co-located with 13th ESWC 2016, Heraklion, Greece, May 30th, 2016.*, 2016.

[21] Norbert Jankowski, Wlodzislaw Duch, Krzysztof Grabczewski, redaktorzy. *Meta-Learning in Computational Intelligence*, wolumen 358 serii *Studies in Computational Intelligence*. Springer, 2011.

[22] Joanna Józefowska, Agnieszka Ławrynowicz, Tomasz Lukaszewski. The role of semantics in mining frequent patterns from knowledge bases in description logics with rules. *TPLP*, 10(3):251–289, 2010.

[23] Lukasz Jozefowski, Agnieszka Ławrynowicz, Joanna Jozefowska, Jedrzej Potoniec, Tomasz Lukaszewski. Kernels for measuring similarity of $\mathcal{EL}^{++}$ description logic concepts. *CoLISD: Collective Learning and Inference on Structured Data, Workshop at ECML/PKDD 2011*, Athens, Greece, 2011.

[24] C. Maria Keet, Agnieszka Ławrynowicz. Test-Driven Development of ontologies. *The Semantic Web. Latest Advances and New Domains - 13th International Conference, ESWC 2016, Heraklion, Crete, Greece, May 29 - June 2, 2016, Proceedings*, strony 642–657, 2016.

[25] C. Maria Keet, Agnieszka Ławrynowicz, Claudia d'Amato, Alexandros Kalousis, Phong Nguyen, Raúl Palma, Robert Stevens, Melanie Hilario. The Data Mining OPtimization Ontology. *J. Web Sem.*, 32:43–53, 2015.

[26] Ewa Kowalczuk, Agnieszka Ławrynowicz. The reporting event ontology design pattern and its extension to report news events. *Advances in Ontology Design and Patterns: Proceedings of the 7th Workshop on Ontology and Semantic Web Patterns*, 2017.

[27] Ewa Kowalczuk, Jedrzej Potoniec, Agnieszka Ławrynowicz. Extracting usage patterns of ontologies on the web: a case study on GoodRelations vocabulary in RDFa. *Proceedings of the 11th International Workshop on OWL: Experiences and Directions (OWLED 2014) co-located with 13th International Semantic Web Conference on (ISWC 2014), Riva del Garda, Italy, October 17-18, 2014.*, strony 139–144, 2014.

[28] Nada Lavrac, Saso Dzeroski. *Inductive Logic Programming: Techniques and Applications*. Routledge, New York, NY, 10001, 1993.

[29] Agnieszka Ławrynowicz. Grouping results of queries to ontological knowledge bases by conceptual clustering. *Computational Collective Intelligence. Semantic Web, Social Networks and Multiagent Systems, First International Conference, ICCCI 2009, Wroclaw, Poland, October 5-7, 2009. Proceedings*, strony 504–515, 2009.

[30] Agnieszka Ławrynowicz. Query results clustering by extending SPARQL with CLUSTER BY. *On the Move to Meaningful Internet Systems: OTM 2009 Workshops, Confederated International Workshops and Posters, ADI, CAMS, EI2N, ISDE, IWSSA, MONET, OnToContent, ODIS, ORM, OTM Academy, SWWS, SEMELS, Beyond SAWSDL, and COMBEK 2009, Vilamoura, Portugal, November 1-6, 2009. Proceedings*, strony 826–835, 2009.

[31] Agnieszka Ławrynowicz. *Semantic Data Mining: An Ontology-based Approach*, wolumen 29 serii *Studies on the Semantic Web*. IOS Press/AKA Verlag, 2017.

[32] Agnieszka Ławrynowicz, Claudia d'Amato, Nicola Fanizzi. A refinement operator based method for semantic grouping of conjunctive query results. *Knowledge-Based and Intelligent Information and Engineering Systems - 14th International Conference, KES 2010, Cardiff, UK, September 8-10, 2010, Proceedings, Part III*, strony 359–368, 2010.

[33] Agnieszka Ławrynowicz, C. Maria Keet. The TDDonto tool for Test-Driven Development of DL knowledge bases. *Proceedings of the 29th International Workshop on Description Logics, Cape Town, South Africa, April 22-25, 2016.*, 2016.

[34] Agnieszka Ławrynowicz, Ross D. King. The open and closed-world assumptions in representing systems biology knowledge. *Proceedings of the Third International Workshop on Machine Learning in Systems Biology (MLSB'09)*, strony 165–166, 2009.

[35] Agnieszka Ławrynowicz, Ilona Lawniczak. The hazardous situation ontology design pattern. *Proceedings of the 6th Workshop on Ontology and Semantic Web Patterns (WOP 2015) co-located with the 14th International Semantic Web Conference (ISWC 2015), Bethlehem, Pensylvania, USA, October 11, 2015.*, 2015.

[36] Agnieszka Ławrynowicz, Ilona Lawniczak. Towards a core ontology of occupational safety and health. *Ontology Engineering - 12th International Experiences and Directions Workshop on OWL, OWLED 2015, co-located with ISWC 2015, Bethlehem, PA, USA, October 9-10, 2015, Revised Selected Papers*, strony 134–142, 2015.

[37] Agnieszka Ławrynowicz, Raúl Palma. Applications of ontology design patterns in the transformation of multimedia repositories. *Proceedings of the 3rd Workshop on Ontology Patterns, Boston, USA, November 12, 2012*, 2012.

[38] Agnieszka Ławrynowicz, Jedrzej Potoniec. Fr-ONT: An algorithm for frequent concept mining with formal ontologies. Marzena Kryszkiewicz, Henryk Rybinski, Andrzej Skowron, Zbigniew W. Raś, redaktorzy, *Foundations of Intelligent Systems*, wolumen 6804 serii *Lecture Notes in Computer Science*, strony 428–437. Springer Berlin Heidelberg, 2011.

[39] Agnieszka Ławrynowicz, Jedrzej Potoniec. Pattern based feature construction in semantic data mining. *Int. J. Semantic Web Inf. Syst.*, 10(1):27–65, 2014.

[40] Agnieszka Ławrynowicz, Jedrzej Potoniec, Lukasz Konieczny, Michal Madziar, Aleksandra Nowak, Krzysztof T. Pawlak. ASPARAGUS - A system for automatic SPARQL query results aggregation using semantics. *Computational Collective Intelligence. Technologies and Applications - Third International Conference, ICCCI 2011, Gdynia, Poland, September 21-23, 2011, Proceedings, Part I*, strony 304–313, 2011.

[41] Agnieszka Ławrynowicz, Jedrzej Potoniec, Michał Robaczyk, Tania Tudorache. Discovery of emerging design patterns in ontologies using tree mining. *Semantic Web*, Preprint(Preprint):1–28, 2017.

[42] Agnieszka Ławrynowicz, Volker Tresp. Introducing machine learning. Johanna Völker, Jens Lehmann, redaktorzy, *Perspectives of Ontology Learning*, Studies on the Semantic Web. AKA Heidelberg / IOS Press, 2014.

[43] Jens Lehmann, Christoph Haase. Ideal downward refinement in the EL description logic. *Proceedings of the 19th international conference on Inductive logic programming*, ILP'09, strony 73–87, Berlin, Heidelberg, 2010. Springer-Verlag.

[44] Jens Lehmann, Johanna Völker. *Perspectives on Ontology Learning*, wolumen 18 serii *Studies on the Semantic Web*. IOS Press, 2014.

[45] Douglas B. Lenat. Cyc: A large-scale investment in knowledge infrastructure. *Commun. ACM*, 38(11):33–38, Listopad 1995.

[46] Jiuyong Li. On optimal rule discovery. *IEEE Trans. Knowl. Data Eng.*, 18(4):460–471, 2006.

[47] John Lloyd. *Foundations of Logic Programming (2nd ed.)*. Springer-Verlag, 1987.

[48] Tom M. Mitchell, William W. Cohen, Estevam R. Hruschka Jr., Partha Pratim Talukdar, Justin Betteridge, Andrew Carlson, Bhavana Dalvi Mishra, Matthew Gardner, Bryan Kisiel, Jayant Krishnamurthy, Ni Lao, Kathryn Mazaitis, Thahir Mohamed, Ndapandula Nakashole, Emmanouil Antonios Platanios, Alan Ritter, Mehdi Samadi, Burr Settles, Richard C. Wang, Derry Tanti Wijaya, Abhinav Gupta, Xinlei Chen, Abulhair Saparov, Malcolm Greaves, Joel Welling. Never-ending learning. *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, January 25-30, 2015, Austin, Texas, USA.*, strony 2302–2310, 2015.

[49] Mikolaj Morzy, Agnieszka Ławrynowicz, Mateusz Zozulinski. Using substitutive itemset mining framework for finding synonymous properties in linked data. *Rule Technologies: Foundations, Tools, and Applications - 9th International Symposium, RuleML 2015, Berlin, Germany, August 2-5, 2015, Proceedings*, strony 422–430, 2015.

[50] Stephen Muggleton. Inductive logic programming. *New Generation Comput.*, 8(4):295–318, 1991.

[51] Sergio Muńoz, Jorge Pérez, Claudio Gutierrez. Minimal deductive systems for RDF. Enrico Franconi, Michael Kifer, Wolfgang May, redaktorzy, *The Semantic Web: Research and Applications*, wolumen 4519 serii *Lecture Notes in Computer Science*, strony 53–67. Springer Berlin Heidelberg, 2007.

[52] Petra Kralj Novak, Anze Vavpetic, Igor Trajkovski, Nada Lavrac. Towards semantic data mining

with g-SEGS. *Proceedings of the 11th International Multiconference Information Society (IS 2009)*, wolumen 20, 2009.

[53] Gordon D. Plotkin. A note on inductive generalization. *Machine Intelligence*, 5:153–163, 1970.

[54] Jedrzej Potoniec, Piotr Jakubowski, Agnieszka Ławrynowicz. Swift linked data miner: Mining OWL 2 EL class expressions directly from online RDF datasets. *J. Web Sem.*, w druku, 2017.

[55] Jedrzej Potoniec, Agnieszka Ławrynowicz. Combining ontology class expression generation with mathematical modeling for ontology learning. *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, January 25-30, 2015, Austin, Texas, USA.*, strony 4198–4199, 2015.

[56] Jedrzej Potoniec, Agnieszka Ławrynowicz. A protege plugin with swift linked data miner. *Proceedings of the ISWC 2016 Posters & Demonstrations Track co-located with 15th International Semantic Web Conference (ISWC 2016), Kobe, Japan, October 19, 2016.*, 2016.

[57] Eric Prud'hommeaux, Andy Seaborne. SPARQL query language for RDF. W3C recommendation, W3C, Stycze/n 2008. http://www.w3.org/TR/2008/REC-rdf-sparql-query-20080115/.

[58] Luc De Raedt. *Logical and relational learning.* Cognitive Technologies. Springer, 2008.

[59] Thomas Rebele, Fabian M. Suchanek, Johannes Hoffart, Joanna Biega, Erdal Kuzey, Gerhard Weikum. YAGO: A multilingual knowledge base from Wikipedia, Wordnet, and Geonames. *The Semantic Web - ISWC 2016 - 15th International Semantic Web Conference, Kobe, Japan, October 17-21, 2016, Proceedings, Part II*, strony 177–185, 2016.

[60] Guus Schreiber, Yves Raimond. RDF 1.1 primer. W3C note, W3C, Czerwiec 2014. http://www.w3.org/TR/2014/NOTE-rdf11-primer-20140624/.

[61] Floarea Serban, Joaquin Vanschoren, Jörg-Uwe Kietz, Abraham Bernstein. A survey of intelligent assistants for data analysis. *ACM Comput. Surv.*, 45(3):31:1–35, Lipiec 2013.

[62] Tomasz Sosnowski, Jedrzej Potoniec, Agnieszka Ławrynowicz. Swift linked data miner extension for WebProtégé. *Knowledge Engineering and Knowledge Management - EKAW 2016 Satellite Events, EKM and Drift-an-LOD, Bologna, Italy, November 19-23, 2016, Revised Selected Papers*, strony 184–187, 2016.

[63] Steffen Staab, Rudi Studer, redaktorzy. *Handbook on Ontologies.* International Handbooks on Information Systems. Springer, wydanie 2nd edition, 2009.

[64] Frank van Harmelen, Deborah McGuinness. OWL web ontology language overview. W3C recommendation, W3C, Luty 2004. http://www.w3.org/TR/2004/REC-owl-features-20040210/.

[65] Denny Vrandecic, Markus Krötzsch. Wikidata: a free collaborative knowledge base. *Commun. ACM*, 57(10):78–85, 2014.