



Marcin Pietranik
Katedra Informatyki Stosowanej
Wydział Informatyki i Telekomunikacji
Politechnika Wroclawska
Wybrzeże Wyspiańskiego 27
50-370 Wrocław

Autoreferat

- 1. Imię i nazwisko:** Marcin Pietranik
- 2. Posiadane dyplomy, stopnie naukowe lub artystyczne – z podaniem podmiotu nadającego stopień, roku ich uzyskania oraz tytułu rozprawy doktorskiej.**
 - 26.09.2008- uzyskanie tytułu zawodowego magistra inżyniera, Politechnika Wroclawska, temat pracy: „*Grupowe środowisko projektowania ontologii webowych*”
 - 15.04.2014- uzyskanie stopnia naukowego doktora, Politechnika Wroclawska, dyscyplina naukowa: informatyka, temat pracy: „*Metody odwzorowywania ontologii wykorzystujące semantykę i wartościowanie atrybutów*”, obrona z wyróżnieniem, promotor prof. dr hab. inż. Ngoc Thanh Nguyen, recenzenci: prof. dr hab. Cezary Orłowski, prof. dr hab. inż. Danuta Zakrzewska
- 3. Informacja o dotychczasowym zatrudnieniu w jednostkach naukowych lub artystycznych.**
 - 15.09.2021 - (nadal trwa) - Politechnika Wroclawska, Wydział Informatyki i Telekomunikacji, Katedra Informatyki Stosowanej, stanowisko: adiunkt badawczo-dydaktyczny
 - 01.01.2020 - 14.09.2021 - Politechnika Wroclawska, Wydział Informatyki i Zarządzania, Katedra Informatyki Stosowanej, stanowisko: adiunkt badawczo-dydaktyczny
 - 01.10.2016 - 31.12.2019 - Politechnika Wroclawska, Wydział Informatyki i Zarządzania, Katedra Systemów Informatycznych, stanowisko: adiunkt badawczo-dydaktyczny
 - 1.12.2014 - 30.09.2016 - Politechnika Wroclawska, Wydział Informatyki i Zarządzania, Katedra Systemów Informatycznych, stanowisko: asystent
 - 17.03.2014 - 31.11.2014 - Politechnika Wroclawska, Wydział Informatyki i Zarządzania, Instytut Informatyki, stanowisko: asystent

4. Omówienie osiągnięć, o których mowa w art. 219 ust. 1 pkt. 2 ustawy z dnia 20 lipca 2018 r. Prawo o szkolnictwie wyższym i nauce (Dz. U. z 2021 r. poz. 478 z późn. zm.) Omówienie to winno dotyczyć merytorycznego ujęcia przedmiotowych osiągnięć, jak i w sposób precyzyjny określać indywidualny wkład w ich powstanie, w przypadku, gdy dane osiągnięcie jest dziełem współautorskim, z uwzględnieniem możliwości wskazywania dorobku z okresu całej kariery zawodowej.

4.1 Osiągnięcie główne

Głównym osiągnięciem naukowym będącym podstawą ubiegania się o nadanie stopnia doktora habilitowanego jest zestaw formalnych metod zarządzania i przetwarzania odwzorowań ontologii, opisanych w monografii naukowej zatytułowanej „**Formal Methods for Managing and Processing Ontology Alignments**”.

4.1.1 Wstęp i cel badań

W informatyce, najczęściej przywoływana definicja ontologii autorstwa Thomasa Grubera, określa je jako „formalną, jednoznaczną specyfikację wspólnej konceptualizacji” ([19]). Definicja ta precyzyjnie oddaje istotę ontologii: „wspólna konceptualizacja” odnosi się do abstrakcyjnego modelu, który opisuje wspólne zrozumienie elementów w danej dziedzinie. Natomiast „formalna, jednoznaczną specyfikacją” oznacza, że konceptualizacja ta jest opisana w sposób precyzyjny za pomocą zbioru pojęć i ich właściwości. Innymi słowy, ontologie to formalne struktury zawierające definicje klas, które wyrażają wiedzę o typach obiektów w określonej dziedzinie, ich relacjach oraz instancjach. Na przykład, ontologia medyczna może zawierać klasy reprezentujące objawy, choroby i terapie, a relacje mogą wskazywać, jaki objaw wiąże się z daną chorobą lub która terapia jest odpowiednia dla konkretnej diagnozy. Instancje mogą natomiast odnosić się do konkretnych pacjentów, ich diagnoz oraz zastosowanych procedur leczniczych.

W literaturze można znaleźć liczne publikacje opisujące różnorodne zastosowania ontologii, a w ostatnich latach rozwój dużych modeli językowych (ang. *Large Language Models*) otworzył nowe możliwości ich wykorzystania ([6]). Jako formalna reprezentacja wiedzy domenowej, ontologie mogą odgrywać kluczową rolę w efektywnym organizowaniu danych - mogą wspierać ugruntowanie wyników generowanych przez duże modele językowe ([3]) lub umożliwiać realizację złożonych zadań wymagających przeprowadzenia procesów wnioskowania, takich jak weryfikacja modeli biznesowych ([56]).

Projektowanie ontologii to złożony i czasochłonny proces, który zazwyczaj wymaga ścisłej współpracy z ekspertami z danej dziedziny. W uproszczeniu można wyróżnić jego trzy główne etapy. Pierwszym jest określenie zakresu modelowanej dziedziny problemowej. Następnie opracowywana jest lista podstawowych pojęć (nazywanych klasami lub konceptami), wraz z opisem ich wewnętrznych struktur oraz relacji jakie występują pomiędzy nimi. Ostatni etap obejmuje zdefiniowanie instancji wyróżnionych klas, co pozwala na szczegółowe odwzorowanie rzeczywistości w ramach projektowanej ontologii. Złożoność tego procesu niesie ryzyko powstawania różnych specyfikacji dla tej samej konceptualizacji, szczególnie w przypadku ontologii obejmujących rozległe i szczegółowe dziedziny, takie jak poświęcona genetyce Gene Ontology ([8]), czy Foundational Model of Anatomy ([59]), zawierająca wiedzę dotyczącą anatomii.

W przypadku w którym konieczne jest zapewnienie komunikacji pomiędzy systemami komputerowymi korzystającymi z różnych ontologii, konieczne staje się zapewnienie możliwości komunikacji między nimi, co wymaga opracowania mapowań pomiędzy zawartościami tych ontologii. Proces wyznaczania takich mapowań jest w literaturze określany mianem odwzorowywania ontologii ([48]), choć w języku polskim spotyka się również terminy takie jak dopasowywanie lub mapowanie ontologii.

Jednym z przykładów praktycznych zastosowań odwzorowywania ontologii jest umożliwienie komunikacji między dwoma systemami medycznymi. Różne urządzenia terapeutyczne, takie jak rezonans magnetyczny lub tomograf komputerowy, mogą opierać się na różnych ontologiach medycznych, takich jak SNOMED-CT lub ICD10 ([58]). Takie ontologie często zawierają nakładające się, pokrewne pojęcia, ale wyrażane w różny sposób. Bez odwzorowania tych ontologii, systemy mogą inaczej interpretować te same informacje kliniczne, co może prowadzić do potencjalnych błędów w opiece nad pacjentem. Poprawne dopasowanie ontologii zapewnia wzajemne zrozumienie systemów oraz niezawodną integrację informacji medycznych na różnych platformach.

Pomimo pozornej prostoty, znalezienie dobrych dopasowań między ontologiami wiąże się z wieloma wyzwaniem. Pierwszym z nich jest opracowanie matematycznej definicji ontologii, dostarczające jednoznacznego opisu zawartości ontologii. W literaturze (szerzej opisanej w kolejnej części niniejszego autoreferatu) większość publikacji traktuje definicje ontologii dość powierzchownie ([65], [29]) lub nawet ich nie podaje ([45], [46]). Najczęściej spotykane podejście ([9], [70]) polega natomiast na przyjęciu formatu OWL [16] (Web Ontology Language) jako jedynej metody definiowania ontologii.

OWL, mimo że jest dojrzałym i szeroko stosowanym narzędziem, posiada pewne ograniczenia, które były podkreślane w wielu publikacjach ([16], [18], [49], [66]). Jednym z głównych problemów jest niedostateczne wsparcie dla atrybutów definiujących strukturę klas - OWL nie pozwala na definiowanie zaawansowanych ograniczeń dla atrybutów i ich wartości, takich jak zastosowanie negacji. Dodatkowo brakuje możliwości obsługi dedykowanych typów danych, co zmusza projektantów do tworzenia nadmiarowych dodatkowych klas i relacji. W konsekwencji, atrybuty definiowane za pomocą elementu `owl:DatatypeProperty`, ograniczają się do par klucz-wartość traktowanych jako proste wartości przypisane do klas. W rzeczywistości jednak znaczenie atrybutów może się różnić w zależności od klasy, w której są używane. Na przykład, atrybut „adres” może być rozumiany inaczej w kontekście klasy „Firma” niż w klasie „Strona internetowa”. Mimo tej oczywistej potrzeby, literatura nie dostarcza rozwiązań dla tego problemu.

OWL oferuje mechanizmy adnotacji, takie jak `rdfs:comment` i `rdfs:seeAlso`, które nie stanowią integralnej części ontologii, a jedynie syntaktyczne konstrukcje w plikach OWL. Poleganie na tych elementach podczas odwzorowywania ontologii, jak ma to miejsce w niektórych podejściach ([64], [7]), jest bardziej próbą obejścia problemu niż jego rzeczywistym rozwiązaniem. W efekcie proces odwzorowywania sprowadza się do analizy podobieństw między plikami OWL, zamiast zajmowania się faktycznymi kwestiami semantycznymi związanymi z odwzorowywaniem ontologii.

W wcześniejszych pracach habilitanta ([53]) przedstawiono alternatywne podejście oparte na formalnej definicji ontologii. Zaproponowano metodę przypisywania semantyki atrybutom, w której każde wystąpienie atrybutu w klasie opisywane jest za pomocą zdania logicznego. Podejście to pozwala na różnorodne interpretacje atrybutów w zależności od klasy, w której są używane. Weryfikacja eksperymentalna wykazała, że metoda ta umożliwia bardziej elastyczne i spójne odwzorowanie ontologii. Główną wadą tej techniki jest jednak konieczność ręcznego definiowania

semantyki dla każdego atrybutu, co znacznie komplikuje i wydłuża proces projektowania ontologii.

Jak łatwo przewidzieć, różne konkurencyjne algorytmy odwzorowujące ontologie mogą generować odmienne dopasowania. Powstaje zatem pytanie, które z tych dopasowań można uznać za „najlepsze”? Jednym z podstawowych podejść do rozwiązania tego problemu jest wykorzystanie odwzorowania referencyjnego i zastosowanie klasycznych metryk, takich jak Precyzja i Kompletność. Takie podejście jest szeroko stosowane w literaturze, głównie dzięki działalności organizacji Ontology Alignment Evaluation Initiative (OAEI)^[1], która od 2004 roku zajmuje się oceną i porównywaniem technik odwzorowywania ontologii. OAEI organizuje coroczne konkursy, w których uczestnicy otrzymują zbiór ontologii testowych, a odwzorowania zwrócone w wyniku działania ich algorytmów są porównywane z odwzorowaniami wzorcowymi, przygotowanymi przez ekspertów. Taka metodyka pozwala na jednoznaczne wyłonienie zwycięzcy. W rzeczywistych zastosowaniach takie wzorcowe odwzorowania niestety nie istnieją. Rodzi to kolejne pytanie: w jaki sposób obiektywnie ocenić i porównać jakość różnych odwzorowań, nie mając dostępu do z góry ustalonego odwzorowania referencyjnego? Kluczowym wyzwaniem staje się zatem opracowanie metod, które pozwalają oceniać odwzorowania ontologii wyłącznie na podstawie informacji zawartych w samych ontologiach.

Warto również podkreślić, że ontologie nie są statyczne – mogą ewoluować, aby uwzględnić nowe potrzeby i wymagania. W takich przypadkach, zmiany w jednej z odwzorowanych ontologii mogą prowadzić do sytuacji, w której istniejące odwzorowanie staje się niekompletne lub nawet błędne. Z tego powodu odwzorowania ontologii, podobnie jak inne elementy współczesnych systemów informatycznych, wymagają bieżącej konserwacji i aktualizacji. Najprostszym, choć kosztownym, rozwiązaniem jest ponowne wyznaczenie odwzorowania po każdorazowej zmianie wprowadzonej do ontologii. Alternatywne rozwiązanie mogłoby składać się z dwóch etapów: wstępnej oceny, czy zmiany wprowadzone w ontologiach w ogóle wpływają na istniejące odwzorowania, a następnie zaktualizowania tylko tych elementów odwzorowania, których dotyczyła wprowadzona modyfikacja.

Podsumowując, wyznaczenie, utrzymanie i ocena jakości odwzorowań ontologii to złożone zagadnienia, a literatura im poświęcona pozostawia wiele obszarów wymagających dalszych prac. Przeprowadzona analiza była inspiracją do sformułowania następujących pytań badawczych:

- *Q1*: W jaki sposób wyznaczyć odwzorowanie ontologii na poziomie klas, relacji i instancji?
- *Q2*: W jaki sposób zarządzać procesem ewolucji ontologii tak, aby możliwe było utrzymanie poprawności ich odwzorowań?
- *Q3*: W jaki sposób ocenić jakość odwzorowań ontologii bazując jedynie na zawartości ontologii?

Odpowiedzi na powyższe pytania znajdują się w monografii habilitanta pod tytułem **Formal Methods for Managing and Processing Ontology Alignments** [50], która zawiera zestaw nowatorskich rozwiązań, wypełniających luki w dotychczasowej literaturze. Wyniki badań obejmują osiągnięcia, które zostaną szczegółowo omówione w dalszych częściach autoreferatu:

1. **Formalny model ontologii oraz odwzorowań** obejmujący kompleksową definicję elementów ontologii, takich jak klasy, relacje i instancje, a także innowacyjne podejście do semantyki atrybutów i relacji oparte na logice zdań. Model ten uwzględnia również szczegółową

¹<https://oaei.ontologymatching.org/>

definicję odwzorowania między ontologiami oraz zestaw kryteriów do wykrywania niespójności w odwzorowaniach.

2. **Zestaw algorytmów do wyznaczania wartości semantyki atrybutów i relacji** które pozwalają na definiowanie znaczenia i kontekstu tych elementów w ontologiach z wykorzystaniem zewnętrznych baz danych leksykalnych, co znacząco upraszcza proces tworzenia ontologii.
3. **Metody odwzorowywania ontologii wykorzystujące logikę rozmytą**, które obejmują algorytmy odwzorowujące klasy, relacje i instancje, w oparciu o dedykowane funkcje podobieństwa oraz reguły wnioskowania rozmytego. Eksperymentalna weryfikacja skuteczności opisanych metod została przeprowadzona zgodnie z metodologią opracowaną przez Ontology Alignment Evaluation Initiative (OAEI). Eksperymenty potwierdziły efektywność opracowanych rozwiązań w różnych scenariuszach testowych, w tym ich przewagę nad innymi podejściami znalezionymi w literaturze.
4. **Formalny model ewolucji ontologii**, obejmujący definicje repozytorium ontologii i dziennika zmian, co pozwala na monitorowanie oraz zarządzanie modyfikacjami wprowadzanymi do ontologii. Model ten zawiera również funkcje oceny istotności wprowadzanych zmian oraz metodę pomiaru stopnia interoperacyjności między ontologiami.
5. **Zestaw algorytmów do aktualizacji odwzorowań ontologii**, pozwalające na zapewnienie poprawności odwzorowań ontologii w odpowiedzi na zmiany wprowadzane w obrębie klas, relacji i instancji.
6. **Metody oceny jakości odwzorowań ontologii na poziomie klas i instancji**, wykorzystujące analizę głębokości klas oraz ciągłości odwzorowań. W odróżnieniu od tradycyjnych metryk, takich jak Precyzja, metody te nie wymagają istnienia odwzorowania referencyjnego opracowanego przez eksperta dziedzinowego.

Monografia habilitanta skupia się na trzech kluczowych obszarach badawczych: wyznaczaniu dopasowań między ontologiami na poziomie klas, relacji i instancji; zarządzaniu procesem ewolucji ontologii; oraz ocenie jakości dopasowań. W dalszej części niniejszego autoreferatu zostanie przedstawiony przegląd literatury, ukazujący istotne luki w badanej dziedzinie, a następnie szczegółowo zaprezentowane zostaną najważniejsze osiągnięcia autora.

4.1.2 Przegląd literatury

W tej części przedstawione zostaną rozwiązania zaproponowane w literaturze, odnoszące się do pytań badawczych określonych w sekcji 4.1.1.

Metody wyznaczania odwzorowania ontologii na poziomie klas, relacji i instancji

Jedną z pierwszych publikacji dotyczących odwzorowywania ontologii był artykuł [47], w którym po raz pierwszy wprowadzono terminy „odwzorowywanie” (ang. alignment) i „łączenie” (ang. merging) ontologii. Autorzy, inspirowani dziedziną baz danych, a zwłaszcza dopasowywaniem schematów baz danych ([57]), opracowali narzędzie PROMPT. Narzędzie to wykorzystywało algorytm, który generował wskazówki wspomagające użytkownika w ręcznym wyznaczaniu odwzorowań między ontologiami.

Od tego czasu w literaturze zaproponowano wiele podejść do problemu odwzorowywania ontologii. Proces ten polega na porównaniu definicji elementów z różnych ontologii w celu wyznaczenia zbioru par elementów, które odnoszą się do tego samego (lub wystarczająco podobnego) obiektu w modelowanej dziedzinie. Pierwszym etapem jest **wstępne przetwarzanie ontologii**, które obejmuje identyfikację typów elementów możliwych do porównania. Następnie zawartość ontologii podlega porównaniu na czterech poziomach: **leksykalnym, strukturalnym, semantycznym i na poziomie instancji**. Kolejny etap obejmuje **wyznaczenie odwzorowań** na podstawie wyników przeprowadzonych porównań, co prowadzi do wygenerowania pierwszej wersji odwzorowania. Ta wersja podlega następnie **doprecyzowaniu i naprawie**, skoncentrowanemu na usuwaniu potencjalnych niespójności ([44]). Kompleksowe rozwiązania odwzorowujące rzadko opierają się tylko na jednej technice.

Rozwój różnych metod bardzo często był kształtowany wykorzystaniem standardu OWL. Innymi słowy, nawet zaawansowane podejścia były inspirowane nie samą ideą odwzorowywania ontologii jako sposobu reprezentacji wiedzy, lecz koniecznością pracy z ontologiami zapisanymi w formacie OWL. Na przykład, narzędzie AgreementMaker i jego pochodne ([68], [13]) łączą porównania terminologiczne, strukturalne i semantyczne w celu osiągnięcia kompleksowych odwzorowań uzyskując dobre wyniki w kampaniach OAEL. System AROA ([75]) został z kolei wzbogacony w reguły asocjacyjne, natomiast system LSMatch ([63]), przedkłada praktyczną skuteczność nad spójność teoretyczną, traktując elementy ontologii jako ciągi znaków. Inne z rozwiązań, OTMapOnto ([2]), wyróżnia się wykorzystaniem metod formalnych; należy go jednak traktować jako wyjątek. Co więcej, wiele systemów (takich jak A-Lion ([1], TOMATO ([60]) i Wiktionary Matcher ([54])) koncentruje się wyłącznie na znajdowaniu odwzorowań klas, zaniedbując odwzorowywanie relacji i instancji. Z drugiej strony, rozwiązania takie jak DLinker ([20]) koncentrują się wyłącznie na mapowaniu instancji, traktując je jako wektory wartości, ignorując fakt, że instancje można przypisać do klas, nadając im dzięki temu semantykę.

Przegląd literatury przedstawiony w monografii wskazuje na rosnący trend w wykorzystaniu technik uczenia maszynowego jako podstawy dla systemów odwzorowywania ontologii. Przykładami takich rozwiązań są ALOD2Vec ([55]), OLaLa ([23]) czy TOM ([34]). Równocześnie platformy, takie jak GraphMatcher ([11]), SORBETMatcher ([17]) czy CIDER-LM ([67]), coraz częściej wykorzystują możliwości oferowane przez duże modele językowe. Skuteczność tych systemów jest jednak bardzo zróżnicowana, a brak uniwersalności pozostaje istotnym wyzwaniem. Systemy osiągające wysokie wyniki w jednej dziedzinie problemowej często prezentują jedynie przeciętne rezultaty w innych. Wynika to z faktu, że duże modele językowe są przygotowywane pod kątem wybranej dziedziny problemowej oraz często traktowane jako „czarne skrzynki”, bez formalnej weryfikacji generowanych wyników. W literaturze istnieją metody wyjaśniania działania dużych modeli językowych ([74]), nie zostały one jak dotąd zastosowane w kontekście odwzorowywania ontologii.

Podsumowując, skuteczność systemów odwzorowywania ontologii jest widoczna głównie w wynikach kampanii OAEL. Mimo że niektóre systemy osiągają znakomite rezultaty, ich działanie jest jednak często zaprojektowane pod kątem specyficznych scenariuszy testowych narzucanych przez OAEL. W rezultacie nie są to uniwersalne rozwiązania, lecz raczej narzędzia skonstruowane z myślą o uczestnictwie w konkursach. Konieczne jest zatem opracowanie holistycznych rozwiązań opartych na podstawach matematycznych, które zwiększyłyby teoretyczną spójność. Dodatkowo, formalne podejście do odwzorowywania ontologii mogłoby zapewnić niezależność od formatu reprezentacji ontologii, eliminując konieczność korzystania wyłącznie ze standardu OWL.

Zarządzanie procesem ewolucji ontologii

Dynamiczny charakter współczesnych wymagań biznesowych często prowadzi do konieczności modyfikowania zawartości ontologii, co może skutkować dezaktualizacją istniejących mapowań. Wyznaczanie odwzorowań od podstaw jest w oczywisty sposób rozwiązaniem mało efektywnym, a dużo lepszą strategią jest analiza wprowadzonych zmian w ontologiach i odpowiednie dostosowanie tylko tych mapowań, na które zmiany mogły wpłynąć.

W artykule [15] przedstawiono ogólną kategoryzację zmian potencjalnie pojawiających się w ontologiach. Stanowi ono podstawę dla kolejnych badań koncentrujących się na dynamice ewolucji ontologii i implikacjach, jakie te zmiany mają na ich powiązania z innymi ontologiami.

Framework τ – OWL, wprowadzony w [71] i rozwinięty w [72], przedstawia kompleksowe rozwiązanie do zarządzania aspektami czasowymi w ontologiach zapisanych w języku OWL. Podejście to zachowuje niezależność danych i upraszcza proces walidacji kolejnych wersji definicji klas i instancji.

Metodę zarządzania ewolucją ontologii, która odbiega od konwencjonalnych technik wersjonowania, można znaleźć w [4]. Zamiast tworzyć odrębne wersje dla każdej aktualizacji ontologii, opisane podejście obejmuje tworzenie historycznego grafu wiedzy, który łączy wszystkie wersje ontologii w jedną, ujednoliczoną strukturę. Ułatwia to dostęp do całej wiedzy zamodelowanej we wszystkich wersjach danej ontologii, jednak pociąga za sobą wiele problemów związanych z przechowywaniem i zarządzaniem stale rozwijającą się ontologią.

Metoda omówiona w [62] przyjmuje podobny punkt wyjścia. Zamiast skupiać się na chronologicznym postępie wersji ontologii, wykorzystuje ona modyfikacje wprowadzone w kolejnych wersjach do skonstruowania tzw. grafu relewancji. Graf ten porządkuje wersje ontologii w oparciu o istotność zmian, a nie porządek chronologiczny. Sama istotność jest określana poprzez ocenę czterech kluczowych kryteriów: poziomu konceptualizacji, częstotliwości użycia, stopnia abstrakcji i kompletności ontologii.

Analiza wpływu ewolucji ontologii na istniejące odwzorowania zostały przedstawione w [22] i oraz [21], gdzie autorzy przedstawili narzędzie COnto-Diff. Zaproponowane rozwiązanie opiera się na analizie odwzorowania ontologii w jego początkowej wersji i następnie przeprowadzania operacji wstawiania nowych mapowań. Podobne rozwiązanie znaleźć można w [69].

Najbardziej kompleksowe podejście zaprezentowano w [32] oraz [33], gdzie kluczowym elementem jest dziennik historii zmian. Pomimo holistycznego charakteru, podejście to traktuje wartość ontologii jednorodnie, bez rozróżnienia między klasami, relacjami i instancjami, co utrudnia przetwarzanie nawet niewielkich zmian w izolacji od innych poziomów. Ponadto nie uwzględnia ono istotności wprowadzanych modyfikacji, co czyni je trudnym do wdrożenia w praktyce.

Podsumowując, podstawowym ograniczeniem rozwiązań znalezionych w literaturze jest brak formalnych ram przetwarzania ontologii i ich mapowania na różnych poziomach abstrakcji (klas, relacji i instancji). Ponadto ontologie są często ograniczane do formatu OWL, co zawęża możliwości ich efektywnego zarządzania. Algorytmy działające niezależnie na poziomie klas, relacji i instancji mogłyby skuteczniej obsługiwać różne typy zmian, dostosowując się do specyfiki każdej warstwy ontologii.

Dodatkowo, w literaturze brakuje ustalonych kryteriów oceny istotności zmian w ontologiach. Nie każda modyfikacja wymaga kompleksowej rewalidacji odwzorowań, co w wielu przypadkach jest zbędne. Opracowanie formalnej miary istotności zmian pozwoliłoby na efektywne śledzenie ich zakresu i charakteru, umożliwiając podjęcie decyzji, czy odwzorowanie między ontologiami

wymaga aktualizacji, na podstawie porównania z ustaloną wartością progową.

Metody oceny jakości odwzorowań ontologii

W literaturze, najczęściej stosowaną metodą oceny jakości odwzorowań ontologii jest podejście opracowane przez organizację OAEI (*Ontology Alignment Evaluation Initiative*). W ramach corocznych konkursów uczestnicy otrzymują zestaw ontologii, a wyniki ich algorytmów są porównywane z referencyjnymi odwzorowaniami przygotowanymi przez ekspertów. Podejście to umożliwia wykorzystanie klasycznych metryk, zaczerpniętych z dziedziny wyszukiwania informacji ([42]).

Jedną z podstawowych metryk jest Precyzja (ang. *Precision*), która mierzy, ile spośród zidentyfikowanych odwzorowań jest poprawnych w odniesieniu do wszystkich znalezionych. Wysoka Precyzja wskazuje na małą liczbę fałszywych pozytywów, co oznacza, że algorytm rzadziej błędnie łączy elementy. Drugą istotną miarą jest Kompletność (ang. *Recall*), która określa zdolność algorytmu do odnajdywania wszystkich poprawnych odwzorowań. Jest to stosunek prawdziwie pozytywnych wyników do całkowitej liczby odwzorowań zawartych w referencyjnym zestawie.

Aby zrównoważyć te dwie miary, stosuje się miarę F (ang. *F-measure*), która jest średnią harmoniczną Precyzji i Kompletności. Miara F pozwala na ocenę balansu pomiędzy dokładnością odwzorowań a ich pełnością, co czyni ją szczególnie przydatną w kontekście porównywania algorytmów.

Mimo swojej powszechności podejście stosowane przez OAEI ma istotne ograniczenia. Największym problemem jest jego zależność od wcześniej przygotowanego odwzorowania referencyjnego, które stanowi punkt odniesienia dla wszystkich ocen. Opisywana metoda ogranicza się do binarnego porównania odwzorowań, polegając jedynie na stwierdzeniu, czy dane dopasowanie jest obecne zarówno w ocenianym odwzorowaniu, jak i w odwzorowaniu referencyjnym. Nie uwzględnia się przy tym stopnia podobieństwa między definicjami elementów w ontologiach.

Alternatywne podejście zaproponowano w pracach [12] oraz [28], w których przedstawiono semantyczne wersje Precyzji i Kompletności. Miały one na celu wykorzystanie wiedzy zawartej w odwzorowywanych ontologiach. W pracy [14] wykazano, że takie podejście lepiej odzwierciedla subiektywną naturę poprawności mapowań. Jednak późniejsze analizy ([10]) wskazały na błędy w tych miarach, które mogą prowadzić do przypisywania różnych wyników równoważnym rozwiązaniom. Ostatnia praca na temat semantycznych wersji Kompletności i Precyzji pochodzi z 2008 roku ([73]), jednak przedstawione w niej modyfikacje nie zostały eksperymentalnie zweryfikowane, co podważa ich praktyczną wartość. W efekcie OAEI nadal opiera swoją metodologię na standardowych wersjach Precyzji i Kompletności, a badania nad ich semantycznymi odpowiednikami zostały zarzucone.

Organizacja OAEI odegrała kluczową rolę w rozwoju całej dziedziny. Zaproponowała spójną, powszechnie akceptowaną metodykę oceny systemów odwzorowujących ontologie oraz udostępniła bardzo obszerny zbiór danych testowych. Niemniej w praktycznych zastosowaniach nie można zakładać istnienia przygotowanego z góry wzorcowego odwzorowania. Powstaje więc pytanie: jak ocenić jakość i porównać dwa konkurencyjne odwzorowania w sytuacji, gdy brak jest takiego odniesienia?

W literaturze nadal brakuje alternatywnych metod oceny jakości odwzorowań ontologii. Należy również podkreślić fakt, że zestaw testowy OAEI nie uległ zmianie na przestrzeni kolejnych edycji. Prowadzi to do sytuacji, w której narzędzia do odwzorowywania rozwijane są przede wszystkim pod kątem konkursu OAEI, a nie z myślą o rzeczywistych zastosowaniach. Wszystko to wyraźnie

wskazuje na potrzebę opracowania nowych metod oceny odwzorowań, które nie będą zależne od danych referencyjnych i jednocześnie zapewnią kompleksową, ekspresywną ocenę ich jakości.

4.1.3 Omówienie osiągniętych wyników

1. Formalny model ontologii oraz odwzorowań

Pierwszym z najważniejszych osiągnięć opisanych w Rozdziale 2 monografii habilitanta [50] jest definicja ontologii wraz ze wszystkimi elementami składowymi oraz odwzorowaniami między nimi. Jest ona wynikiem wieloletnich prac habilitanta, które rozpoczęły się artykułem [53] i stanowi fundament dla pozostałych części monografii, będąc uniwersalnym i niezależnym od metody reprezentacji podejściem do opisu ontologii, które uwzględnią zarówno ich strukturalne jak i semantyczne aspekty. W bieżącej sekcji autoreferatu omówionych zostanie tylko kilka podstawowych definicji.

Definicja 1. Zakładając istnienie zbioru A zawierającego atrybuty mogące służyć opisaniu obiektów z wybranej dziedziny problemowej, przez V_a oznaczamy dziedzinę atrybutu a ze zbioru A . V jest natomiast zbiorem wartości wszystkich atrybutów z A ($V = \bigcup_{a \in A} V_a$). Ontologia oparta o parę (A, V) definiowana jest jako krotka $O = (C, R^C, I, R^I, H)$ gdzie:

- C jest skończonym zbiorem klas,
- R^C jest zbiorem relacji klas $R^C = \{r_1^C, r_2^C, \dots, r_n^C\}$, ($n \in \mathbb{N}$), takim, że każdy $r_i^C \in R^C$ jest podzbiorem iloczynu kartezjańskiego $C \times C$,
- I reprezentuje zbiór unikalnych identyfikatorów instancji,
- $R^I = \{r_1^I, r_2^I, \dots, r_n^I\}$ jest zbiorem relacji instancji, który, w przeciwieństwie do R^C definiującego potencjalne relacje na poziomie klas, opisuje szczegółowo określone interakcje między instancjami klas. Na przykład, w ontologii medycznej, zbiór R^C może zawierać relacje takie jak $diagnoses^C$, wskazujące na możliwe interakcje takie jak $(Lekarz, Pacjent)$, R^I zawiera natomiast konkretne relacje między instancjami takie jak $diagnoses^I$, obejmujące rzeczywiste pary identyfikatorów instancji takie jak $(John\ Smith, Peter\ Jones)$, $(Alice\ Brown, Mary\ Clark)$.
- H reprezentuje hierarchię klas, formalnie będącą podzbiorem iloczynu kartezjańskiego $H \subseteq C \times C$. Elementy tego zbioru służą wyrażeniu wyróżnionej relacji typu *jest podklasą* między klasami ontologii, reprezentującej związek dziedziczenia. W monografii habilitanta zdefiniowany został szereg ograniczeń nakładanych na elementy zbioru H .

Definicja 2. Każda klasa c ze zbioru C ma strukturę zdefiniowaną jako $c = (id^c, A^c, V^c, I^c)$, gdzie id^c jest identyfikatorem klasy, A^c jest zbiorem atrybutów klasy, V^c jest zbiorem dziedzin atrybutów ($V^c = \bigcup_{a \in A^c} V_a$), natomiast I^c jest zbiorem instancji przypisanych do klasy.

Definicja 3. Język atrybutów jest językiem formalnym oznaczonym jako EXP_A . Jest on zdefiniowany jako zbiór wyrażeń zbudowanych z alfabetu $D_A = \{\alpha_1, \alpha_2, \dots, \alpha_p\}$ i trzech operatorów logicznych: koniunkcji \wedge , alternatywy \vee i negacji \neg . Przy wykorzystaniu języka EXP_A możliwe jest zdefiniowanie semantyki atrybutów w ontologii $O = (C, R^C, I, R^I, H)$ jako funkcji o sygnaturze $S_A : A \times C \rightarrow EXP_A$. Funkcja ta przypisuje wybrane wyrażenie EXP_A do każdego

uwzględnienia atrybutów ze zbioru A do klas z danej ontologii. Takie podejście umożliwia jaśniejsze zrozumienie roli atrybutów, a w konsekwencji umożliwia dokładniejsze i skuteczniejsze wykorzystanie atrybutu w różnych scenariuszach. Rozważmy istnienie atrybutu *birthday* zawartego w pojęciu *Person*. Można przyjąć następującą wartość funkcji $S_A: S_A(\text{birthday}, \text{Person}) = \text{day_of_birth} \wedge \text{month_of_birth} \wedge \text{year_of_birth} \wedge \text{age} \wedge \text{zodiac_sign}$, co pozwala opisać, że według autorów danej ontologii, atrybut *birthday* wyraża zarówno datę urodzin danej osoby, jej wiek oraz znak zodiaku.

Definicja 4. Język relacji jest językiem formalnym oznaczonym jako EXP_R . Jest on zdefiniowany jako zbiór wyrażeń zbudowanych z alfabetu $D_R = \{\beta_1, \beta, \dots, \beta_q\}$ i trzech operatorów logicznych: koniunkcji \wedge , alternatywy \vee i negacji \neg . Wykorzystując język relacji można zdefiniować semantykę relacji jako funkcję, która (w analogiczny sposób co funkcja S_A) przypisuje każdej relacji ze zbioru R^C wyrażenie języka EXP_R . Ma ona sygnaturę $S_R: R^C \rightarrow EXP_R$

Definicja 5. Instancja i ze zbioru I^c z pewnego pojęcia $c \in C$ jest zdefiniowana jako para $i = (id^i, v_c^i)$ gdzie id^i jest identyfikatorem instancji, a v_c^i jest funkcją o sygnaturze $v_c^i: A^c \rightarrow V^c$ pozwalającą na przypisanie konkretnych wartości atrybutom pobranych z definicji klasy c . Wartościowanie funkcji v_c^i jest ściśle powiązane z wystąpieniem instancji w ramach klasy i można je traktować jako wektory określonych wartości atrybutów ze zbioru A^c . Dla uproszczenia notacji, możemy zapisać $i \in c$, co wyraża, że instancja i należy do klasy c .

Na podstawie opisanych definicji poszczególnych elementów składowych ontologii możliwe było opracowanie formalnych zasad ich odwzorowywania. Tematyka ta została szczegółowo omówiona w Rozdziale 2.3 prezentowanej monografii habilitacyjnej.

Definicja 6. Oznaczmy zbiór wszystkich potencjalnych dopasowań wszystkich ontologii opartych na (A, V) jako \widetilde{AL} . Biorąc pod uwagę dwie ontologie oparte na (A, V) $O_1 = (C_1, H_1, R^{C_1}, I_1, R^{I_1})$ i $O_2 = (C_2, H_2, R^{C_2}, I_2, R^{I_2})$, odwzorowanie $Align(O_1, O_2) \in \widetilde{AL}$ między nimi jest zdefiniowane następująco:

$$Align(O_1, O_2) = \{Align_C(O_1, O_2), Align_I(O_1, O_2), Align_R(O_1, O_2)\} \quad (1)$$

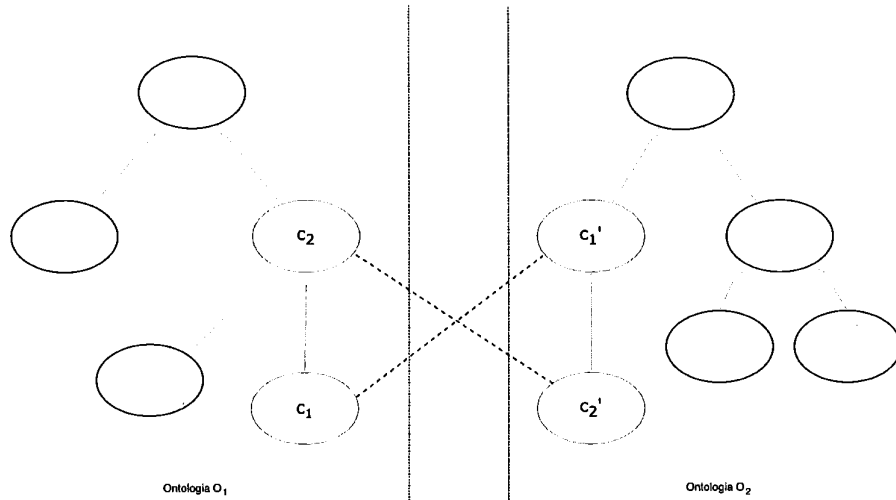
Definicja ze Wzoru [1] umożliwia identyfikację odrębnych zestawów odwzorowań dla poziomu klas (reprezentowanych przez $Align_C(O_1, O_2)$), instancji (reprezentowanych przez $Align_I(O_1, O_2)$) i relacji (reprezentowanych przez $Align_R(O_1, O_2)$). Zbiory wszystkich potencjalnych dopasowań na poziomach klas, relacji i instancji będą oznaczane odpowiednio jako \widetilde{AL}_C , \widetilde{AL}_R , \widetilde{AL}_I . Zbiory $Align_C(O_1, O_2)$ i $Align_R(O_1, O_2)$ są natomiast zdefiniowane jako podzbiory iloczynów kartezyjańskich $C_1 \times C_2$, $R_1 \times R_2$. Odwzorowanie $Align_I(O_1, O_2)$ ma natomiast bardziej złożoną strukturę:

$$Align_I(O_1, O_2) = \{AL_{O_1, O_2}(c_1, c_2) \mid (c_1, c_2) \in Align_C(O_1, O_2)\} \quad (2)$$

gdzie każdy z elementów $AL_{O_1, O_2}(c_1, c_2)$ jest podzbiorem iloczynu kartezyjańskiego $I^{c_1} \times I^{c_2}$, co ogranicza odwzorowania instancji do par instancji należących do klas odwzorowanych na siebie.

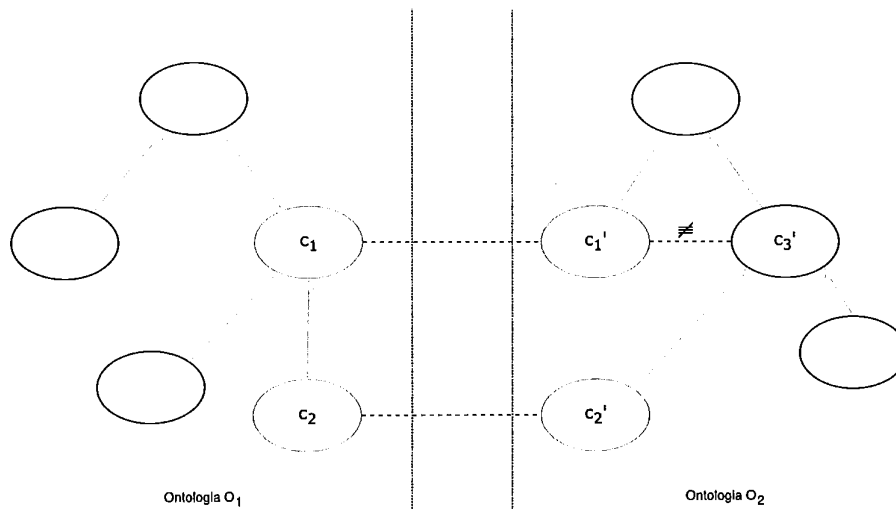
W oparciu o Definicję [6] w Rozdziale 2.3.2 monografii autor opracował formalne kryteria niespójności jakie mogą w nich wystąpić w odwzorowaniach: **dziedziczenie cykliczne** i **naruszenie ograniczeń klas rozłącznych**. Pierwszy typ niespójności, **dziedziczenie cykliczne**, występuje, gdy w hierarchii dziedziczenia między dwoma lub więcej klasami w odwzorowywanych ontologiach

tworzy się pętla. Przykład tego rodzaju niespójności został przedstawiony na Rysunku 1, gdzie klasy c_1 oraz c_2 z dwóch ontologii tworzą cykl dziedziczenia, co jest sprzeczne z podstawowymi zasadami hierarchii dziedziczenia, według których klasa nie może być jednocześnie potomkiem i przodkiem innej klasy. Tego typu niespójność może prowadzić do problemów podczas procesów wnioskowania wykorzystujących ontologie jako bazę wiedzy.



Rysunek 1: Przykład niespójności w postaci dziedziczenia cyklicznego

Drugim typem niespójności jest **naruszenie ograniczeń klas rozłącznych**. Przykład takiej niespójności przedstawiono na Rysunku 2, gdzie klasy c_1 oraz c_2 są odwzorowane na klasy c_1' oraz c_2' , pomimo faktu, że klasa c_2' jest potomkiem klasy c_3' , która jest rozłączna z klasą c_1' , co prowadzi do sprzeczności.



Rysunek 2: Przykład niespójności w postaci naruszenia ograniczeń klas rozłącznych

W monografii wskazano ponadto rzeczywiste przykłady omawianych niespójności występujące znalezione w odwzorowaniach ontologii Foundational Model of Anatomy (FMA), SNOMED-CT, Adult Mouse Anatomy i NCI Thesaurus. Wszystkie te ontologie są szeroko wykorzystywane w praktyce.

Zestawienie przedstawione w bieżącej sekcji obejmuje jedynie podstawowe elementy wyselekcjonowane z Rozdziału 2 monografii stanowiącej główne osiągnięcie habilitanta. Jego głównym celem jest wprowadzenie czytelnika w podstawowe założenia teoretyczne, które stanowią niezbędny fundament dla dalszych części niniejszego autoreferatu.

2. Algorytmy do wyznaczania wartości semantyki atrybutów i relacji

Drugim z osiągnięć zaprezentowanych w monografii habilitacyjnej [50] jest opracowanie algorytmów do wyznaczania semantyki atrybutów i relacji w ontologiach. Opisane w Rozdziale 2.2 metody mają na celu wyznaczenie wartości funkcji S_A oraz S_R z Definicji [3] i [4], które definiują semantykę atrybutów oraz relacji jako wyrażenia logiki zerowego rzędu. W początkowych etapach prac zakładano, że semantykę określa się ręcznie podczas procesu projektowania ontologii. Takie podejście sprawdza się dla małych ontologii, jednak w przypadku złożonych dziedzin modelowanych za pomocą ontologii ten sposób staje się nieefektywny i narażony na błędy. Na przykład, w niewielkiej ontologii złożonej z pięciu klas, każda posiadająca pięć atrybutów, konieczne byłoby utworzenie 25 wyrażeń logicznych. Już dla takiego przypadku manualne przypisywanie semantyki atrybutów staje się trudne, natomiast dla dużych ontologii, takich jak SNOMED-CT (obejmującej ponad 360000 klas), takie podejście jest praktycznie niemożliwe w realizacji.

Aby rozwiązać ten problem, Rozdział 2.2 książki opisuje półautomatyczne podejście, które integruje zewnętrzne bazy danych leksykalnych (tezaursy), umożliwiając automatyzację części procesu. Ze względu na fakt, że wybrany tezaurus musi dostarczać informacje o znaczeniach wyrazów, takich jak ich części mowy, synonimy, antonimy oraz relacje hiperonimii i hiponimii, w monografii habilitanta przedstawiono formalną definicję tezausa. Zakłada ona, że tezaurus jest zbiorem wpisów, z których każdy składa się z nazwy, części mowy, synonimów, antonimów oraz relacji hiperonimii i hiponimii. Ponadto możliwe jest określenie relacji między dwoma wpisami w tezaursie, co pozwala na wyznaczenie najkrótszej ścieżki między dwoma wpisami. Formalne podejście zapewnia niezależność od konkretnej tezausa i pozwala na wykorzystanie dowolnego, który spełnia kryteria wynikające z przyjętej definicji. W monografii zaproponowano dwa algorytmy: do wyznaczania wartości semantyki atrybutów (a więc wartości funkcji S_A) oraz do wyznaczania wartości semantki relacji (a więc wartości funkcji S_R). W przypadku pierwszego wykorzystano bazę WordNet^[2], a w przypadku drugiego użyta została baza WordsAPI^[3].

Algorytm dedykowany atrybutom opiera się na analizie ścieżek w grafie taksonomicznym skonstruowanym na bazie relacji hiperonimii w WordNet. Proces rozpoczyna się od przetworzenia nazw atrybutów i powiązanych z nimi klas na zbiory słów kluczowych. Następnie przeprowadzane jest porównanie tych słów z wpisami w tezaursie, aby zidentyfikować najkrótszą ścieżkę między odpowiednimi węzłami w grafie. Kolejne elementy tej ścieżki, czyli nazwy wpisów reprezentujących pośrednie pojęcia, są dodawane do wyrażenia logicznego, które opisuje relację atrybutu z klasą, tym samym wyznaczając szukaną wartość funkcji S_A . Na przykład, dla klasy „Pies” i atrybutu

²<https://wordnet.princeton.edu/>

³<https://www.wordsapi.com/>

„dieta” algorytm mógłby zidentyfikować ścieżkę prowadzącą przez wpisy takie jak „zwierzę domowe”, „zwierzę” i „pokarm”. Na podstawie tych pojęć generowane jest wyrażenie logiczne odzwierciedlające semantyczny związek atrybutu „dieta” z klasą „Pies”. Takie podejście zapewnia kontekstualizację atrybutów w odniesieniu do danej klasy, co pozwala na precyzyjne modelowanie znaczenia.

W przypadku przypisywania semantyki relacjom zastosowano inne podejście. Proces rozpoczyna się od wstępnej obróbki językowej nazwy relacji. Na tym etapie algorytm usuwa słowa stopu, takie jak spójniki czy przyimki, które nie niosą istotnej informacji semantycznej, oraz sprowadza rzeczowniki i czasowniki do ich podstawowych form (np. lematyzacja). W ten sposób uzyskiwane są kluczowe elementy znaczeniowe, które reprezentują istotę relacji. Następnie algorytm odwołuje się do zewnętrznego tezaury, umożliwiając użytkownikowi wybór wpisów najlepiej odpowiadających znaczeniu nazwy relacji. Ten krok zapewnia kontrolę nad procesem i umożliwia uwzględnienie kontekstu specyficznego dla danej ontologii, co automatyczne procedury mogą pominąć. Wybrane wpisy tezaury stają się podstawą do wygenerowania semantyki relacji. Algorytm tworzy następnie zestaw wyrażeń logicznych, bazując na synonimach i antonimach wybranego wpisu. Synonimy pozwalają na bardziej wszechstronne zdefiniowanie relacji, uwzględniając różne terminy, które mogą być używane zamiennie w modelowanej dziedzinie. Antonimy, z kolei, wprowadzane z symbolem negacji (\neg) pomagają wyeliminować potencjalne niejednoznaczności i lepiej określić granice znaczenia danej relacji. Dzięki takiemu podejściu generowane wyrażenia logiczne nie tylko opisują znaczenie relacji w sposób szczegółowy, ale również pozwalają na łatwiejsze zrozumienie i interpretację jej roli w kontekście ontologii. Proces ten, choć wymagający wkładu użytkownika, znacznie zwiększa precyzję i ekspresyjność semantyki relacji, umożliwiając bardziej zaawansowane modelowanie wybranej dziedziny w ontologiach.

Rozdział 2.2 monografii podkreśla istotną rolę półautomatycznych metod w przypisywaniu semantyki atrybutom i relacjom w ontologiach. Metody te skutecznie eliminują problemy związane z czasochłonną i podatną na błędy pracą manualną, co jest szczególnie istotne przy modelowaniu złożonych dziedzin wiedzy. Zaprezentowane rozwiązania, wykorzystujące integrację z zewnętrznymi tezaurusami, takimi jak WordNet i WordsAPI, nie tylko upraszczają proces tworzenia ontologii, ale także znacząco zwiększają ich spójność. Dzięki temu możliwe jest skuteczniejsze wykorzystanie ontologii w praktyce.

3. Metody odwzorowywania ontologii wykorzystujące logikę rozmytą

Trzecim osiągnięciem jest opracowanie zestawu metod odwzorowywania ontologii z wykorzystaniem logiki rozmytej, opisanych w Rozdziale 4 monografii habilitanta [50]. Dzięki zastosowaniu logiki rozmytej, proces dopasowywania elementów ontologii staje się bardziej elastyczny i możliwe jest zintegrowanie wyników z różnych cząstkowych metod odwzorowywania w spójny wynik końcowy, dzięki zastosowaniu rozmytych reguł wnioskowania. Takie rozwiązanie wprowadza do procesu mapowania dodatkową wiedzę ekspercką.

Proces odwzorowywania rozpoczyna się od wyodrębnienia elementów (klas, relacji oraz instancji) z wejściowych ontologii O_1 i O_2 . Zaproponowane w monografii metody zostały podzielone właśnie z uwagi na różnorodność tych elementów. Kolejnym etapem jest porównanie wyodrębnionych elementów za pomocą specjalnie zaprojektowanych funkcji podobieństwa, dostosowanych do różnych typów elementów. Wyniki funkcji podobieństwa są przekształcane na zmienne rozmyte za pomocą funkcji przynależności, co umożliwia lingwistyczne przedstawienie stopnia podobieństwa

między elementami ontologii. Na tej podstawie wykonywane jest wnioskowanie przy użyciu rozmytych reguł. Reguły te mają strukturę "IF-THEN", gdzie część IF opisuje warunki w kategoriach zmiennych rozmytych, a część THEN określa wynik końcowy. Wynikiem jest wartość zmiennej rozmytej, która odzwierciedla stopień odwzorowania między elementami ontologii.

Wszystkie metody zostały zaimplementowane z użyciem biblioteki jFuzzyLogic⁴, co umożliwiło ich weryfikację eksperymentalną. Testy przeprowadzono na zestawie danych opracowanym i dostarczonym przez Ontology Alignment Evaluation Initiative⁵, korzystając ze standardowych miar oceny jakości: Precyzji, Kompletności i miary F1. Dzięki temu potwierdzono skuteczność i praktyczność proponowanego podejścia.

Metoda wyznaczania odwzorowań na poziomie klas umożliwia znalezienie rozwiązania problemu formalnie zdefiniowanego jako: Dla danych dwóch ontologii $O_1 = (C_1, H_1, R^{C_1}, I_1, R^{I_1})$ i $O_2 = (C_2, H_2, R^{C_2}, I_2, R^{I_2})$ należy wyznaczyć odwzorowanie $Align_C(O_1, O_2)$ zdefiniowanego na równaniu [7]. Tabela [8] przedstawia trzy funkcje podobieństwa użyte do obliczania stopnia pokrewieństwa par klas $c_1 \in C_1$ i $c_2 \in C_2$. Zostały one formalnie zdefiniowane w Rozdziale 4.2.1 omawianej monografii habilitanta.

Tabela 1: Funkcje podobieństwa pomiędzy klasami

Funkcja	Opis
$sim_{attr}(c_1, c_2)$	Wartość podobieństwa atrybutów klas, zdefiniowana jako odległość Jaccarda pomiędzy zbiorami A^{c_1} i A^{c_2} , obliczana na podstawie wzoru $sim_{attr}(c_1, c_2) = \frac{ A^{c_1} \cap A^{c_2} }{ A^{c_1} \cup A^{c_2} }$. Jest ona możliwa do obliczenia ze względu na to, że zarówno O_1 , jak i O_2 są oparte na (A, V) , co oznacza, że dzielą wspólną pulę atrybutów.
$sim_{jw}(c_1, c_2)$	Wartość podobieństwa pomiędzy id^{c_1} i id^{c_2} obliczana przy użyciu podobieństwa Jaro-Winklera
$sim_{wn}(c_1, c_2)$	Wartość podobieństwa Wu-Palmera pomiędzy id^{c_1} i id^{c_2}

W podrozdziale 4.2.2 znajdują się natomiast szczegółowe definicje funkcji przynależności użyte do rozmycia obliczonych podobieństw oraz zestaw reguł wnioskowania, które zostały zastosowane podczas eksperymentu. Wyniki tego badania zostały omówione w tej samej części książki. Niestety, organizacja OAEI nie udostępniła szczegółowych wyników dla poszczególnych narzędzi odwzorowujących konkretne pary ontologii na wybranych danych testowych - dostępne są jedynie uśrednione wyniki dla całego zbioru testowego. W celu porównania opracowanej metody z innymi narzędziami, obliczono zatem średnie wartości Precyzji, Kompletności i miary F dla każdej pary ontologii.

W przeprowadzonym eksperymencie omawiane narzędzie uzyskało bardzo dobrą średnią wartość Precyzji wynoszącą 0,87, jedynie o 0,01 gorszą od najwyższego wyniku 0,88. Natomiast średnia wartość miary Kompletności osiągnęła poziom 0,59, co jest wynikiem akceptowalnym w porównaniu z innymi narzędziami. W przypadku Miary F1, łączącej Precyzję i Kompletność, opisywane podejście uzyskało wynik 0,69, co plasuje je w czołówce analizowanych rozwiązań. Eksperyment dowiódł zatem, że opracowana metoda odwzorowywania klas przewyższa większość konkurencyjnych metod i dodatkowo, potwierdza skuteczność zastosowania logiki rozmytej.

⁴<https://jfuzzylogic.sourceforge.net/html/index.html>

⁵<https://oaei.ontologymatching.org/>

Metoda wyznaczania odwzorowań na poziomie relacji umożliwia znalezienie rozwiązania problemu formalnie zdefiniowanego jako: Dla danych dwóch ontologii $O_1 = (C_1, H_1, R^{C_1}, I_1, R^{I_1})$ i $O_2 = (C_2, H_2, R^{C_2}, I_2, R^{I_2})$ należy wyznaczyć odwzorowanie $Align_R(O_1, O_2)$ zdefiniowanego na równaniu [1]. Tabela [2] przedstawia funkcje podobieństwa użyte do obliczania stopnia pokrewieństwa par relacji $r_1 \in R_1$ i $r_2 \in R_2$. Zostały one formalnie zdefiniowane w Rozdziale 4.3.1 omawianej monografii habilitacyjnej.

Tabela 2: Funkcje podobieństwa pomiędzy relacjami

Funkcja	Opis
$sim_C(c_1, c_2)$	Wartość podobieństwa między dwiema klasami, która jest wykorzystywana do obliczania podobieństwa klas biorących udział w porównywanych relacjach z dwóch ontologii, stosowana jest dwukrotnie – raz dla każdej klasy na obu końcach relacji. Umożliwia to porównanie strukturalnego kontekstu danej relacji. Może to być dowolna funkcja podobieństwa klas dostępna w literaturze.
$sim_R(r_1, r_2)$	Wartość podobieństwa wartości semantyki relacji z Definicji 4.

Rozdział 4.3.2 zawiera opis eksperymentu przeprowadzonego w celu zweryfikowania skuteczności zaproponowanej metody. Rozpoczyna się on omówieniem funkcji przynależności, które zostały użyte do rozmywania obliczonych wartości podobieństwa, oraz zastosowanego zestawu reguł wnioskowania. Sam eksperyment składał się z dwóch etapów. W pierwszym z nich użyto referencyjnych odwzorowań klas dostarczonych przez OAEI. W drugim etapie wykorzystano zewnętrzne narzędzie LogMap, które, według literatury, jest jednym z najbardziej efektywnych systemów do odwzorowywania ontologii [30]. W obu etapach do określenia semantyki relacji, niezbędnej do obliczenia wartości funkcji sim_R , wykorzystano algorytm bazujący na teaurusie WordsAPI^[6], opracowany przez autora omawianej monografii i szczegółowo przedstawiony w Rozdziale 2.2. Został on opisany jako Osiągnięcie 2 w niniejszym autoreferacie.

Przeprowadzenie eksperymentu poświęconego poziomowi relacji niosło za sobą kilka trudnień z uwagi na charakter zestawu danych referencyjnych dostarczonych przez Ontology Alignment Evaluation Initiative. Ponadto, wiele rozwiązań odwzorowujących opisywanych w literaturze pomija aspekt relacji, koncentrując się głównie na poziomach klas lub instancji. Jest to potwierdzone w artykule [5], gdzie wykazano, że najlepiej oceniane systemy według rankingu OAEI w zakresie odwzorowywania relacji uzyskały średnią Precyzję na poziomie 0,36, średnią Kompletność wynoszącą 0,18 oraz średnią miarę F1 równą 0,21.

Wyniki wszystkich badań zostały zaprezentowane w tej samej części książki - w obu omawianych procedurach zaproponowane rozwiązanie zwróciło lepsze wyniki niż metody znane z literatury. Znalezione odwzorowania relacji cechowały się wysokimi wartościami Precyzji i Kompletności, co zostało skrótowo przedstawione na Tabeli [3], która zawiera uśrednione wartości tych miar.

⁶<https://www.wordsapi.com/>

Tabela 3: Wyniki eksperymentów dla poziomu relacji

	Precyzja	Kompletność	Miara F1
Eksperyment 1	0,53	0,29	0,37
Eksperyment 2	0,63	0,25	0,36
Dane OAEI z [5]	0,45	0,17	0,23

Metoda wyznaczania odwzorowań na poziomie instancji umożliwia znalezienie rozwiązania problemu formalnie zdefiniowanego jako: Dla danych dwóch ontologii $O_1 = (C_1, H_1, R^{C_1}, I_1, R^{I_1})$ i $O_2 = (C_2, H_2, R^{C_2}, I_2, R^{I_2})$ należy wyznaczyć odwzorowanie $Align_I(O_1, O_2)$ zdefiniowanego na równaniu [7]. Tabela 4 przedstawia siedem funkcji podobieństwa użytych do obliczania stopnia pokrewieństwa par $i_1 \in I_1$ i $i_2 \in I_2$. Zostały one formalnie zdefiniowane w Rozdziale 4.4.1 omawianej monografii habilitanta.

Tabela 4: Funkcje podobieństwa pomiędzy instancjami

Funkcja	Opis
$sim_{PR}(i_1, i_2)$	Funkcja podobieństwa właściwości instancji rozumianych jako atrybuty oraz relacje. Takie podejście umożliwia użycie opracowanego rozwiązania do porównywania instancji w ontologiach wyrażonych w języku OWL [16], w którym atrybuty oraz relacje są traktowane tak samo.
$sim_{AS}(i_1, i_2)$	Funkcja podobieństwa właściwości instancji podobna do sim_{PR} , ograniczona jednak tylko do atrybutów.
$sim_{RS}(i_1, i_2)$	Funkcja podobieństwa właściwości instancji podobna do sim_{PR} , ograniczona tylko do relacji.
$sim_{PC}(i_1, i_2)$	Funkcja podobieństwa obliczana jako stosunek właściwości (rozumianych zarówno jako relacje, jak i atrybuty), które zostały pomyślnie dopasowane pomiędzy instancjami i_1 i i_2 , do całkowitej liczby właściwości skojarzonych z tymi instancjami w ramach klas c_1 i c_2 .
$sim_{CC}(i_1, i_2)$	Funkcja podobieństwa oparta na obliczaniu stosunku liczby klas o maksymalnym stopniu podobieństwa (wyliczonego za pomocą dowolnej odpowiedniej funkcji) do których przypisane są porównywane instancje, do liczby wszystkich klas do których zostały przypisane.
$sim_{MS}(i_1, i_2)$	Funkcja podobieństwa obliczana jako stosunek liczby relacji łączących instancje i_1 i i_2 do łącznej liczby wszystkich relacji powiązanych z i_1 i i_2 .
$sim_{PS}(i_1, i_2)$	Funkcja podobieństwa jest obliczana jako stosunek liczby atrybutów wspólnych dla instancji i_1 i i_2 do łącznej liczby wszystkich atrybutów, które kiedykolwiek zostały użyte w obu tych instancjach, we wszystkich przypisanych im klasach (zgodnie z definicją, instancje mogą należeć do wielu klas, które mogą mieć różne zestawy atrybutów).

Rozdział 4.4.2 przedstawia funkcje przynależności zastosowane do rozmywania obliczonych podobieństw oraz reguły wnioskowania wykorzystane w eksperymencie. Wyniki tego badania omówiono w tej samej części książki. Podobnie jak w przypadku eksperymentów poświęconych poziomowi klas i relacji, do weryfikacji opracowanej metody odwzorowywania ontologii na poziomie instancji użyte zostały dane testowe przygotowane przez OAEI. Tym razem wykorzystano zbiór ISLab Instance Matching Benchmark (IIMB) dedykowany testowaniu odwzorowywania instancji. Zawiera on 79 ontologii będących wynikiem losowych zmian nanoszonych do pojedynczej ontologii źródłowej, które można podzielić na trzy główne kategorie - transformacje **wartości danych**,

struktur danych oraz semantyki danych. Zbiór zawiera zatem 80 par ontologii wraz z odwzorowaniami wzorcowymi. Dla każdej z nich wyznaczone zostało odwzorowanie, dla którego obliczone zostały przyjęte miary jakości jakimi są Precyzja, Kompletność oraz Miara F1.

Szczegółowe wyniki wraz z ich analizą statystyczną znaleźć można w Rodziale 4.4.2 monografii. Tabela 5 zawiera uśrednione wartości miar oraz uśrednione wartości miar obliczone dla odwzorowań wygenerowanych przez system LogMap([30]). Jak łatwo zauważyć opracowane rozwiązanie doskonale radzi sobie w grupie testów dotyczących transformacji semantyki danych. W pozostałych przypadkach, jakość zwracanych wyników jest również bardzo dobra. Analiza statystyczna wykazała, że uzyskane rezultaty nigdy nie były gorsze, a w dużej ilości przypadków lepsze niż uzyskane przez konkurencyjne rozwiązania z literatury.

Podsumowując, wszystkie opracowane metody odwzorowywania ontologii oparte na logice rozmytej zostały poddane eksperymentalnej weryfikacji, przeprowadzonej zgodnie z metodologią i przy użyciu danych testowych dostarczonych przez Ontology Alignment Evaluation Initiative. Wyniki jednoznacznie potwierdziły ich wysoką skuteczność. Istotnym atutem zaproponowanych metod jest ich pełna uniwersalność, ponieważ nie są zależne od konkretnego formatu zapisu ontologii - opierają się one wyłącznie na aparacie matematycznym. Jego uniwersalność została potwierdzona podczas weryfikacji eksperymentalnej, w której dane testowe były zapisane w formacie OWL. Zaproponowane metody z powodzeniem poradziły sobie z tym formatem, co świadczy o ich praktyczności.

Tabela 5: Podsumowanie wyników odwzorowywania ontologii na poziomie instancji

		Precyzja	Kompletność	F1
Transformacje wartości danych	Metoda rozmyta	0,9630	0,5280	0,6319
	LogMap	0,8964	0,8928	0,8894
Transformacje struktur danych	Metoda rozmyta	0,9241	0,4585	0,4811
	LogMap	0,9343	0,9856	0,9592
Transformacje semantyki danych	Metoda rozmyta	0,9488	0,9469	0,9388
	LogMap	0,8546	0,9466	0,8926
Transformacje mieszane	Metoda rozmyta	0,9243	0,0680	0,1095
	LogMap	0,9199	0,7581	0,8195

4. Formalny model ewolucji ontologii

Współczesne systemy komputerowe muszą sprostać dynamicznie zmieniającym się wymaganiom biznesowym, co oznacza, że ich kluczowe komponenty, takie jak ontologie, również mogą wymagać dostosowania. Ewolucja ontologii może wynikać z różnych czynników, takich jak aktualizacja wiedzy dziedzinowej, która poszerza lub precyzuje istniejące informacje, czy konieczność eliminacji błędnych informacji, które mogą wpływać na funkcjonowanie systemu.

Aby umożliwić skuteczne zarządzanie takimi zmianami oraz uwzględnienie ich w procesie aktualizacji odwzorowań ontologii, autor monografii habilitacyjnej opracował formalny model zarządzania ewolucją ontologii, umożliwiający monitorowanie i zarządzanie zmianami zachodzącymi w ontologiach. Podstawą opracowanego podejścia jest zestaw definicji, które zostały szcze-

gółowo opisane w Rozdziale 5 monografii. Wyniki przedstawione w tej części pracy są efektem badań prowadzonych w ramach projektu „Metody zarządzania ewolucją ontologii i ich odwzorowań”, finansowanego przez Narodowe Centrum Nauki w ramach programu SONATA (nr 2017/26/D/ST6/00251). Stanowią one istotny wkład autora w rozwiązanie problematyki poruszonej w drugim pytaniu badawczym, sformułowanym w sekcji [4.1.1](#)

Pierwszym krokiem było opracowanie metody elastycznego modelowania czasu, umożliwiającej precyzyjne wyrażanie stanu ontologii w czasie oraz zarządzanie wszelkimi zmianami, które mogą w nich zachodzić. Zostało to zrealizowane za pomocą uniwersalnej osi czasu reprezentowanej jako zbiór $TL = \{t_n | n \in \mathbb{N}\}$ zawierający dyskretne momenty w czasie. Porządek częściowy na zbiorze TL (zdefiniowany jako para $\langle TL, \prec \rangle$) jest możliwy dzięki istnieniu przeciwzrotnej, przechodniej i niesymetrycznej relacji \prec .

Definicja 7. Oś czasu ontologii O , oznaczona jako $TL(O)$ jest podzbiorem uniwersalnej osi czasu TL (formalnie, $TL(O) \subseteq TL$). Składa się z serii momentów, w których ontologia została w jakiś sposób zmodyfikowana. Każdy moment w $TL(O)$ odpowiada określonej wersji ontologii O

Definicja 8. Repozytorium ontologii O , oznaczone jako $Rep(O)$, jest zbiorem zawierającym kolekcję różnych stanów ontologii O utworzonych w czasie. Jest to uporządkowana sekwencja różnych wersji ontologii, gdzie każda wersja jest reprezentowana jako $O^{(m)}$. Formalnie jest to zdefiniowane jako

$$Rep(O) = \left\{ O^{(m)} \mid m \in TL(O) \right\} \quad (3)$$

Notacja $O^{(m)} = (C^{(m)}, H^{(m)}, R^{C(m)}, I^{(m)}, R^{I(m)})$ służy do odniesienia się do ontologii O w określonym momencie czasu $t_m \in TL$. Reprezentuje ona stan ontologii w tym momencie, uwzględniając jej klasy ($C^{(m)}$), hierarchię ($H^{(m)}$), relacje między pojęciami ($R^{C(m)}$), instancje ($I^{(m)}$) oraz relacje między instancjami ($R^{I(m)}$). Zapis $O^{(m-1)} \prec O^{(m)}$ oznacza, że ontologia $O^{(m-1)}$ jest wcześniejszą wersją ontologii w porównaniu do $O^{(m)}$. Notacja ta może być również stosowana do poszczególnych elementów ontologii, np. $c^{(m-1)} \prec c^{(m)}$ reprezentuje fakt, że klasa $c^{(m-1)}$ jest wcześniejszą wersją klasy $c^{(m)}$. Można jej również bezpośrednio użyć w Definicji 6 odwzorowania ontologii - $Align(O_1^{(m)}, O_2^{(n)})$ oznacza dopasowanie między ontologiami O_1 i O_2 w ich odpowiednich stanach w czasie m i n z uniwersalnej osi czasu TL . Każda oś czasu $TL(O)$ ma punkt początkowy, oznaczony jako 0 , który oznacza rozpoczęcie śledzenia ewolucji ontologii O . W tym początkowym momencie ontologia jest reprezentowana jako $O^{(0)} = (C^{(0)}, H^{(0)}, R^{C(0)}, I^{(0)}, R^{I(0)})$.

Efektywne zarządzanie ewolucją ontologii wymaga narzędzia, które nie tylko informuje o momencie wystąpienia zmian, ale także szczegółowo opisuje ich charakter. W tym celu wprowadzono Dziennik Ontologii (ang. *Ontology Log*), który zapewnia kompleksowy wgląd w wszystkie modyfikacje dokonane w ontologii, umożliwiając precyzyjne śledzenie i analizę wprowadzanych zmian.

Definicja 9. Dziennik ontologii to zbiór trójek utworzonych przy użyciu trzech funkcji $diff_C$, $diff_R$ i $diff_I$, które wyznaczają zmiany zastosowane do ontologii O odpowiednio na poziomie klas, relacji i instancji. Po ich zastosowaniu dla wszystkich kolejnych par sąsiadujących wersji ontologii na osi czasu $TL(O)$, wygenerowany zostaje zbiór:

$$\text{Log}(O) = \left\{ \left\langle \text{diff}_C(O^{(m-1)}, O^{(m)}), \text{diff}_I(O^{(m-1)}, O^{(m)}), \right. \right. \\ \left. \left. \text{diff}_{RC}(O^{(m-1)}, O^{(m)}) \right\rangle \mid m \in \text{TL}(O) \setminus \{0\} \right\} \quad (4)$$

Rozdział 5.2 monografii zawiera dokładne definicje funkcji diff_C , diff_R i diff_I . Przygotowany formalizm matematyczny pozwala na wyrażanie nie tylko podstawowych operacji dodawania lub usuwania elementów, ale również śledzenie bardziej złożonych modyfikacji, jak np. aktualizacji struktury klasy:

1. diff_C jest funkcją, która przyjmuje dwie wersje zbioru klas ontologii, $C^{(m-1)}$ i $C^{(m)}$, jako dane wejściowe i zwraca trzy zbiory wyjściowe: zbiór klas nowo dodanych do ontologii, zbiór klas z niej usuniętych oraz zbiór zmodyfikowanych klas, który zawiera pary tych samych klas w ich wcześniejszym i późniejszym stanie.
2. diff_R jest funkcją przyjmującą dwie wersje zbioru relacji R^C . Jej wynik obejmuje trzy zbiory, które opisują dodane relacje, usunięte relacje i zmodyfikowanych relacje.
3. diff_I jest funkcją, która przyjmuje jako dane wejściowe dwie wersje zbiorów klas i instancji. W wyniku zwrócone są trzy zbiory - zbiór identyfikatorów instancji, które zostały dodane do ontologii, zbiór identyfikatorów instancji, które zostały usunięte z ontologii, oraz zbiór obejmujący identyfikatory instancji, których wartościowania zostały zmodyfikowane.

Oczywistym jest spostrzeżenie, że nie wszystkie zmiany w ontologiach są tak samo istotne - czasem dotyczą one pojedynczej wartości w instancji, a czasem są to rozległe modyfikacje struktur wielu klas. Funkcje diff_C , diff_R i diff_I pozwalają na śledzenie zmian w ontologiach, nie pozwalają jednak na ocenę stopnia istotności modyfikacji ontologii. W tym celu zdefiniowane zostały trzy nowe funkcje, Ψ_C , Ψ_R i Ψ_I , służące do obliczania stopnia modyfikacji klas, relacji i instancji. Funkcje te przyjmują na wejściu uszeregowane czasowo opisy zmian ontologii, w wyniku zwracając wartość stopnia istotności zmian. Na poziomie klas funkcja Ψ_C musi spełnić następujące postulaty:

- **P1.** $\Psi_C(C^{(m-1)}, C^{(m)}) \in [0, 1]$
- **P2.** $\Psi_C(C^{(m-1)}, C^{(m)}) = 0 \iff \text{diff}_C(C^{(m-1)}, C^{(m)}) = \langle \emptyset, \emptyset, \emptyset \rangle$
- **P3.** $\Psi_C(C^{(m-1)}, C^{(m)}) = 1 \iff \text{diff}_C(C^{(m-1)}, C^{(m)}) = \langle C^{(m)}, C^{(m-1)}, \emptyset \rangle$

Pierwszy postulat **P1** zakłada, że funkcja Ψ_C musi przyjmować wartości w zakresie od 0 do 1. Kolejne dwa postulaty odnoszą się do przypadków brzegowych. Drugi postulat **P2** opisuje sytuację, w której zmiany w ontologii są minimalne, a wartość funkcji wynosi 0, co ma miejsce, gdy na poziomie klas nie wprowadzono żadnych modyfikacji – nie dodano, nie usunięto ani nie zmieniono żadnych klas. Postulat **P3** dotyczy przeciwnego przypadku - narzuca wymaganie, że wartość funkcji powinna wynosić 1, gdy cała zawartość ontologii na poziomie klas uległa zmianie.

Na podstawie tak opracowanych postulatów zdefiniowana została dokładna postać funkcji Ψ_C :

$$\Psi_C(C^{(m-1)}, C^{(m)}) = \frac{|new_C(C^{(m-1)}, C^{(m)})| + |del_C(C^{(m-1)}, C^{(m)})|}{|C^{(m)}| + |del_C(C^{(m-1)}, C^{(m)})|} + \frac{\sum_{(c_1, c_2) \in alt_C(C^{(m-1)}, C^{(m)})} d_s(ctx(c_1), ctx(c_2))}{|C^{(m)}| + |del_C(C^{(m-1)}, C^{(m)})|} \quad (5)$$

Funkcja składa się z dwóch składników: stosunków liczby zmian wprowadzonych do klas i całkowitej liczby klas w ontologii. Pierwszy składnik to stosunek nowo dodanych i usuniętych klas do sumy całkowitej liczby klas w ostatnim stanie i liczby usuniętych klas. Drugi składnik to stosunek sumy odległości pomiędzy wszystkimi parami tych sobie klas w ich przeszłym i obecnym stanie, które zostały zmienione (np. przemianowane, zmieniono ich hierarchię) do sumy całkowitej liczby klas w ostatnim stanie i liczby usuniętych klas. Ten komponent mierzy stopień, w jakim zmienione klasy zostały zmodyfikowane. W tym celu wykorzystana została funkcja d_s o sygnaturze $d_s : EXP_A \cup EXP_R \times EXP_A \cup EXP_R \rightarrow [0, 1]$, która mierzy odległość między dwoma wyrażeniami logicznymi z EXP_A (z Definicji [3]) lub EXP_R (z Definicji [4]). Opisana została ona w artykule [53]. Ze względu na fakt, że oba komponenty równania [5] mają ten sam mianownik, można je traktować jako jedno wyrażenie. Mianownik zawsze będzie wyższy od licznika, co zapewnia, że wynik Ψ_C zawsze będzie mieścił się w zakresie $[0, 1]$, co w konsekwencji spełnia wymagania narzucone przez postulat **P1**.

W sytuacji, w której nie zostaną wprowadzone żadne zmiany do koncepcji w ontologii, różnica między poprzednią wersją $C^{(m-1)}$ a bieżącą wersją $C^{(m)}$ będzie zbiorem pustym. W tym przypadku licznik równania Equation [5] będzie równy 0, co spowoduje, że całe równanie będzie równe 0, co spełnia postulat **P2**. Dla przeciwnego przypadku, jeśli każde pojęcie w ontologii zostanie zmodyfikowane, różnica między $C^{(m-1)}$ a $C^{(m)}$ będzie zawierać wszystkie klasy. W tym przypadku licznik i mianownik równania Equation [5] będą równe, co da w wyniku wartość 1, spełniając tym samym wymóg postulatu **P3**.

Funkcja Ψ_C z równania [5] pozwala na określenie momentu czasowego, w którym zmiany w ontologiach powinny znaleźć odzwierciedlenie w weryfikacji poprawności ich odwzorowań. Użytkowaną wartość można porównać z ustaloną wartością progową, co pozwala podejmować decyzje dotyczące potencjalnej dezaktualizacji odwzorowania ontologii. Wartość progowa, określana przez eksperta, może być dostosowana do specyfiki dziedziny i zastosowania ontologii oraz jej odwzorowań. Przeprowadzone badania wykazały wyraźną korelację między obliczoną istotnością zmian w ontologii a różnicami w odwzorowaniach wygenerowanych dla niej przed i po wprowadzeniu modyfikacji. W analogiczny sposób zdefiniowane zostały funkcje Ψ_R i Ψ_I , które umożliwiają obliczenie stopnia istotności zmian na poziomie relacji oraz instancji.

W niniejszej sekcji przedstawiono formalny model ewolucji ontologii, kładąc szczególny nacisk na elastyczne modelowanie czasu, śledzenie wprowadzonych modyfikacji oraz ocenę ich istotności. Zdefiniowanie uniwersalnej osi czasu (Definicja [7]) oraz repozytorium wersji ontologii (Definicja [8]) umożliwiło powiązanie kolejnych stanów ontologii z konkretnymi momentami w czasie, a dziennik ontologii (Definicja [9]) dostarcza zaś szczegółowych informacji o charakterze i zakresie wprowadzonych modyfikacji. Opracowane funkcje Ψ_C , Ψ_R i Ψ_I pozwalają na ilościowe określenie istotności zmian wprowadzanych odpowiednio w klasach, relacjach i instancjach, a porównanie ich wartości z przyjętymi wartościami progowymi umożliwia precyzyjne zarządzanie procesem aktualizacji ontologii oraz ich odwzorowań. Przedstawiony model stanowi zatem spójną i kompleksową

platformę do analizy i zarządzania ewolucją ontologii, zwiększając przydatność tych struktur wiedzy w zmiennych warunkach biznesowych.

5. Zestaw algorytmów do aktualizacji odwzorowań ontologii

Jak opisano we wcześniejszej sekcji niniejszego autoreferatu, ontologie mogą ulec zmianie w wyniku zmieniających się wymagań biznesowych. Na skutek takich zmian, wyznaczone zawczasu odwzorowanie dwóch ontologii może ulec dezaktualizacji i uwzględniać błędne informacje. W takiej sytuacji konieczne staje się ponowne uruchomienie procedury odwzorowywania ontologii i zastąpienie nieaktualnego dopasowania, nowym. W przypadku częstych modyfikacji ontologii takie podejście może jednak okazać się nieefektywne. Pojawia się zatem pytanie - czy istnieje możliwość aktualizacji odwzorowania ontologii w oparciu o opis zmian, które zostały wprowadzone do ontologii?

Rozdział 5.3 monografii habilitacyjnej [50] rozwija wcześniej przedstawiony formalny model zarządzania ewolucją ontologii, koncentrując się na metodach utrzymywania aktualności odwzorowań w obliczu zachodzących zmian. W tej części książki zaprezentowano zestaw algorytmów dedykowanych aktualizacji odwzorowań ontologii, które umożliwiają dostosowywanie istniejących mapowań do nowych warunków wynikających z modyfikacji ontologii. Podobnie jak w przypadku pozostałych rozdziałów monografii, opracowanego metody zostały podzielone na trzy grupy, odpowiadające trzem poziomom ontologii: klas, relacji i instancji. Dla poziomu klas opracowane zostały następujące procedury:

1. **Algorytm sprawdzający czy nowo dodane klasy mogą zostać odwzorowane (Rozdział 5.4.1.1 monografii)** umożliwiającą analizę nowych klas w ontologii i sprawdzenie, czy mogą one zostać poprawnie odwzorowane. Procedura polega na:
 - Wyznaczeniu nowych klas dodanych do ontologii (przy wykorzystaniu funkcji $diff_C$)
 - Określeniu potencjalnych mapowań nowych klas z klasami z drugiej ontologii poprzez zastosowanie odpowiednich funkcji podobieństwa między nowymi, a istniejącymi klasami w drugiej ontologii.
 - Aktualizacji odwzorowania poprzez dodanie nowych mapowań spełniających przyjęte kryteria.
2. **Algorytm przeprowadzający walidację mapowań odnoszących się do zmodyfikowanych klas (Rozdział 5.4.1.2 monografii)** umożliwiającą ponowną weryfikację mapowań klas, które zostały zmodyfikowane w wyniku zmian w ontologii. W szczególności algorytm ten:
 - Wyznacza klasy, które uległy zmianom (przy wykorzystaniu funkcji $diff_C$)
 - Waliduje istniejące mapowania zmodyfikowanych klas, sprawdzając, czy nadal są one poprawne.
 - Sprawdza czy zmienione klasy mogą zostać zamapowane z klasami z drugiej ontologii.
 - Aktualizacji odwzorowania poprzez dodanie nowych mapowań spełniających przyjęte kryteria lub usunięciu tych, które utraciły poprawność.
3. **Algorytm usunięcia mapowań nieistniejących klas (Rozdział 5.4.1.3)** jest najprostszym z algorytmów przeznaczonym do obsłużenia sytuacji, w których klasy są usuwane z ontologii. Wykorzystując funkcję $diff_C$ metoda wyznacza zbiór usuniętych klas i następnie usuwa powiązane mapowania, które stały się nieaktualne.

Eksperymenty, które symulowały ewolucję ontologii przy wykorzystaniu predefiniowanych scenariuszy testowych, potwierdziły wysoką jakość odwzorowań uzyskanych poprzez aktualizację na podstawie informacji o wprowadzonych zmianach. Procedura testowa opierała się na nowatorskich metodach oceny jakości odwzorowań, które analizują wyłącznie zawartość dopasowywanych ontologii. Te metody oraz ich zastosowanie zostaną szczegółowo opisane w Sekcji [4.1.3](#) niniejszego autoreferatu.

Przeprowadzone badania oraz analiza statystyczna wykazały, że jakość odwzorowań zaktualizowanych za pomocą zaproponowanych algorytmów była porównywalna z jakością odwzorowań generowanych od podstaw przy użyciu jednego z wiodących narzędzi do dopasowywania ontologii – systemu LogMap ([31]). Wyniki te potwierdzają skuteczność opracowanych algorytmów w zadaniu aktualizacji odwzorowań, które osiągają jakość na poziomie rozwiązań uznawanych za standard w dziedzinie.

Dodatkowo, w monografii szczegółowo przeanalizowano złożoność obliczeniową każdego z zaproponowanych algorytmów, dowodząc, że ich złożoność nie odbiega od metod dostępnych w literaturze. Co więcej, dzięki specyfice działania tych algorytmów, które przetwarzają zazwyczaj niewielki podzbiór elementów zaktualizowanych w wyniku ewolucji ontologii, ich efektywność jest wyższa w porównaniu do metod wymagających pełnego, ponownego wyznaczenia odwzorowań. Zaproponowane podejście pozwala zatem nie tylko na zachowanie wysokiej jakości dopasowań, ale również na znaczną oszczędność zasobów obliczeniowych.

Opracowane algorytmy stały się podstawą do kontynuowania prac. W Rozdziale 5.4.2 monografii omówiono zestaw algorytmów przeznaczonych do aktualizacji odwzorowań na poziomie relacji, które podobnie jak poprzednie rozwiązania, zostały podzielone na trzy odrębne procedury. Następnie, w Rozdziale 5.4.3 skupiono się na aktualizacji odwzorowań na poziomie instancji.

Podczas wyznaczenia odwzorowania dwóch ontologii za pomocą omawianych algorytmów, kluczowym celem jest osiągnięcie możliwie najwyższej jakości wyników. Jakość tę można ocenić przy użyciu metod przedstawionych w literaturze lub z wykorzystaniem miar opisanych w szóstym rozdziale monografii habilitacyjnej, szczegółowo omówionych w dalszej części tego autoreferatu jako Osiągnięcie 6. Takie odwzorowanie umożliwia opracowanie narzędzia wspierającego współpracę systemów komputerowych opartych na tych ontologiach, co w literaturze określane jest również mianem interoperacyjności [43].

Naturalny rozwój ontologii wejściowych, w tym ich aktualizacje, może jednak wpływać na jakość tej współpracy. W odpowiedzi na to wyzwanie, w Rozdziale 5.5 monografii habilitacyjnej zaproponowano miarę umożliwiającą bieżące monitorowanie zmian stopnia interoperacyjności dwóch ontologii, wynikających z modyfikacji ich odwzorowań. Dla poziomu klas miara ta została zdefiniowana jako funkcja $\Delta(O_1^{(m)}, O_2^{(m)}, O_1^{(n)}, O_2^{(n)}, Align_C(O_1^{(m)}, O_2^{(m)}), Align_C(O_1^{(n)}, O_2^{(n)}))$ przyjmującą na wejściu ontologie O_1 oraz O_2 w dwóch momentach czasu m i n takich, że $m < n$ wraz z odpowiadającymi im wersjami odwzorowaniami $Align_C$. Wykorzystuje ona pomocniczą funkcję μ o sygnaturze $\mu : \tilde{O} \times \tilde{O} \rightarrow \mathbb{R}^+$, która pozwala na obliczenie stopnia interoperacyjności (reprezentowanej przez dodatnią liczbę rzeczywistą) uzyskanego za pomocą zadanego odwzorowania. Musi ona spełnić następujące postulaty:

- **P1.** $\mu(O_1, O_2, Align_C(O_1, O_2)) = 0 \iff Align_C(O_1, O_2) = \emptyset$
- **P2.** $\mu(O_1, O_2, Align_C(O_1, O_2)) = 1 \iff \forall_{(c_{r1}, c_{r2}) \in Align_C(O_1, O_2)} (c_{r1} \in Root(O_1) \wedge c_{r2} \in Root(O_2))$

Postulat **P1** odnosi się do sytuacji, w której odwzorowanie, oznaczone jako $Align_C(O_1, O_2)$, między ontologiami O_1 i O_2 jest puste. W takiej sytuacji potencjał wiedzy wynikający z ich interoperacyjności nie występuje, ponieważ brak mapowań klas obu ontologii uniemożliwia jakąkolwiek wymianę informacji. Z tego powodu funkcja $\mu(O_1, O_2)$ powinna przyjąć wartość 0.

Postulat **P2** odnosi się do sytuacji, w której mapowania w odwzorowaniu $Align_C(O_1, O_2)$ łączą jedynie klasy z najwyższego poziomu hierarchii obu ontologii. Takie mapowania zapewniają podstawowe połączenie między O_1 a O_2 , nie łączą jednak bardziej szczegółowych klas położonych na niższych poziomach taksonomii. W takim przypadku funkcja μ powinna zwrócić wartość 1.

Zdefiniowana w Rozdziale 5.5 monografii habilitanta funkcja μ pozwala na zmierzenie stopnia interoperacyjności pomiędzy dwiema odwzorowanymi ontologiami, zakładając, że pozostają one niezmiennie w czasie. Została ona wykorzystana we wspomnianej funkcji Δ do pomiaru w jakim stopniu interoperacyjność dwóch ontologii zmieniła się w czasie w wyniku zmian wprowadzonych do tych ontologii oraz aktualizacji ich odwzorowania:

$$\Delta(O_1^{(m)}, O_2^{(m)}, O_1^{(n)}, O_2^{(n)}, Align_C(O_1^{(m)}, O_2^{(m)}), Align_C(O_1^{(n)}, O_2^{(n)})) = \left(\frac{\mu(O_1^{(m)}, O_2^{(m)}, Align_C(O_1^{(m)}, O_2^{(m)}))}{\mu(O_1^{(n)}, O_2^{(n)}, Align_C(O_1^{(n)}, O_2^{(n)}))} - 1 \right) * 100\% \quad (6)$$

Opracowana funkcja zwraca wartości z zakresu $\Delta \in [-100\%, \infty)$. Dodatnie wartości Δ wskazują na wzrost wiedzy w porównaniu do stanu wcześniejszego, co oznacza wzrost potencjału wiedzy. Natomiast ujemne wartości Δ wskazują na spadek wiedzy przechowywanej w dwóch ontologiach, co zmniejsza potencjał wiedzy. Brzegowa wartość -100% dotyczy sytuacji w której zmiany wprowadzone do ontologii spowodowały usunięcie wszystkich mapowań z odwzorowania i tym samym całkowitą utratę interoperacyjności.

W Rozdziale 5.5.2 monografii habilitanta opisano eksperyment mający na celu zweryfikowanie użyteczności funkcji Δ w analizie ewolucji ontologii. W ramach eksperymentu opracowano zestaw 10 scenariuszy symulujących dynamiczne zmiany ontologii, umożliwiające szczegółowe prześledzenie modyfikacji zachodzących w ich strukturze. Scenariusze te zostały zastosowane do par ontologii wybranych z testowego zbioru udostępnianego przez Ontology Alignment Evaluation Initiative (OAEI). Dla każdej pary ontologii, za pomocą narzędzia LogMap ([31]), wygenerowano dwa odwzorowania: jedno przed ewolucją ontologii, a drugie po jej dokonaniu. Następnie obliczono wartości funkcji Δ dla wszystkich odwzorowań w celu oceny wpływu zmian na interoperacyjność ontologii. Analiza statystyczna wyników wykazała wyraźną zbieżność wartości funkcji Δ ze stopniem rozległości modyfikacji wprowadzonych do ontologii. Uzyskane wyniki potwierdziły zatem skuteczność funkcji Δ jako narzędzia do monitorowania zmian w interoperacyjności ontologii. Dzięki temu funkcja ta okazuje się przydatnym elementem w zarządzaniu ewolucją ontologii, co czyni ją przydatnym narzędziem w zarządzaniu ewolucją ontologii w różnych zastosowaniach.

W artykule poświęconym integracji ontologii [61] zauważono, że zagadnienia związane z tym tematem pojawiają się w literaturze już od 2001 roku. Jak jednak wykazano w Rozdziale 3 monografii habilitacyjnej, który zawiera szczegółowy przegląd rozwiązań z zakresu integracji ontologii, dotychczasowe podejścia często pomijają kluczowe wyzwania związane z utrzymaniem spójności ontologii w warunkach ich ewolucji oraz nie zapewniają wszechstronnej analizy wszystkich aspektów tej problematyki. Metody przedstawione w Rozdziale 5 monografii [50] dostarczają natomiast kompleksowego spojrzenia na opisywany problem, wypełniając tym samym lukę pozostawioną

przez wcześniejsze badania.

6. Metody oceny jakości odwzorowań ontologii na poziomie klas i instancji

Jak wykazano w monografii [50], jednym z kluczowych elementów procesu tworzenia i utrzymywania odwzorowań ontologii jest ocena ich jakości. W literaturze jednym z podejść do tego zagadnienia jest metodologia opracowana przez Ontology Alignment Evaluation Initiative (OAEI), która opiera się na wykorzystaniu standardowych miar, takich jak Precyzja i Kompletność. Jednak aby obliczyć te miary, konieczne jest posiadanie referencyjnego odwzorowania zawierającego wyłącznie poprawne i kompletne mapowania. W praktycznych scenariuszach takie referencyjne odwzorowanie zazwyczaj nie istnieje, co uniemożliwia zastosowanie metodologii OAEI w rzeczywistych warunkach.

W sytuacji, gdy pojawia się potrzeba odwzorowania ontologii, istnieje możliwość wygenerowania mapowań przy użyciu różnych konkurencyjnych narzędzi. Niemniej jednak w literaturze nie opracowano dotąd kryteriów pozwalających na porównanie jakości takich odwzorowań bez dostępu do referencyjnego odwzorowania co wyklucza wykorzystanie podejścia OAEI w praktyce, podkreślając tym samym potrzebę opracowania alternatywnych metod oceny jakości, które bazowałyby wyłącznie na zawartości odwzorowanych ontologii.

W odpowiedzi na tę lukę badawczą, w Rozdziale 6 monografii habilitacyjnej zaproponowano cztery nowe metody oceny jakości odwzorowań, które nie wymagają odniesienia do referencyjnych mapowań. Proponowane rozwiązania stanowią istotny wkład autora w problematykę poruszaną w trzecim pytaniu badawczym sformułowanym w sekcji [4.1.1], oferując praktyczne i niezależne narzędzia do analizy jakości odwzorowań ontologii.

Zgodnie z Definicją [1], taksonomia klas odnosi się do poziomu szczegółowości reprezentowanego przez daną klasę, co ma bezpośrednie przełożenie na szczegółowość instancji tej klasy. Rozważmy ontologię, w której na najwyższym poziomie hierarchii znajdują się ogólne klasy. Klasy te są zwykle wysoce abstrakcyjne, pełniąc rolę ramowego opisu modelowanej dziedziny. W miarę schodzenia w dół hierarchii ontologii, napotykamy coraz bardziej szczegółowe podklasy, z których każda oferuje wyższy poziom szczegółowości.

To spostrzeżenie stało się podstawą metod oceny mapowań ontologii zaproponowanych w Rozdziale 6.1.1 monografii [53]. Metody te uwzględniają zarówno poziom klas (Λ_C) jak i poziom instancji (Λ_I), kładąc nacisk na znaczenie głębokości odwzorowanych elementów w hierarchii. Bazują one na założeniu, że mapowania obejmujące szczegółowe, wyspecjalizowane klasy powinny mieć większe znaczenie niż te, które dotyczą ogólnych, wysokopoziomowych klas. Wynika to z faktu, że odwzorowania szczegółowych klas odzwierciedlają precyzyjniejsze powiązania i bardziej złożone zależności między elementami ontologii.

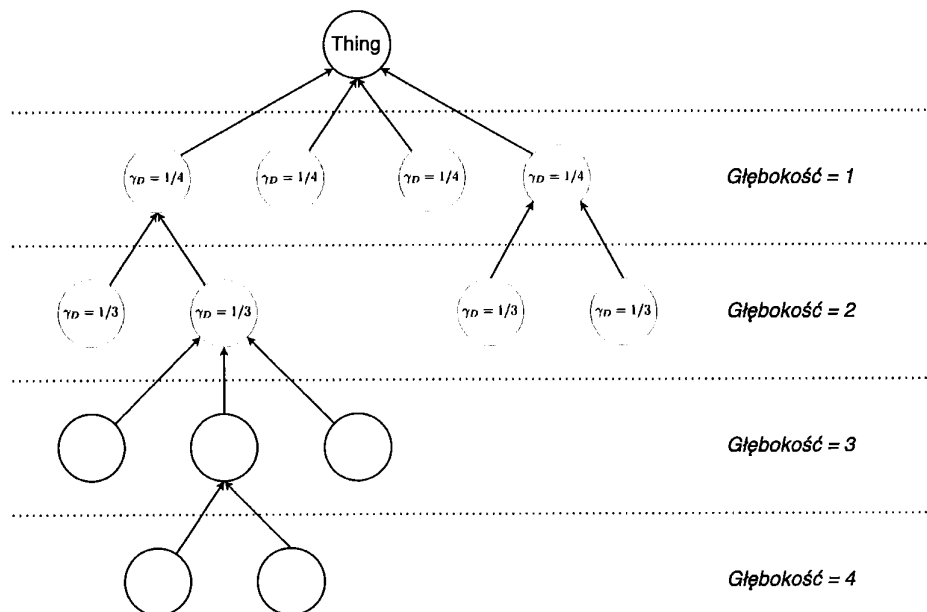
Obie zaproponowane metody wykorzystują funkcję γ_D , która przypisuje punktację mapowaniu $\langle c_1, c_2 \rangle$ na podstawie głębokości klas c_1 i c_2 . Dla dwóch ontologii $O_1 = (C_1, H_1, R^{C_1}, I_1, R^{I_1})$ i $O_2 = (C_2, H_2, R^{C_2}, I_2, R^{I_2})$ oraz odwzorowania $Align_C(O_1, O_2)$ (zdefiniowanego zgodnie z Definicją [6]) ma ona sygnaturę $\gamma_D : \tilde{O} \times \tilde{O} \times \tilde{AL}_C \times Align_C(O_1, O_2) \rightarrow \mathbb{R}^+$.

W monografii habilitanta przedstawiona została pełna definicja funkcji γ_D (uwzględniająca kryterium spójności wejściowego mapowania z pozostałymi mapowaniami z danego odwzorowania) oraz algorytm do obliczania jej wartości. Rysunek [3] pokazuje natomiast przykładowe wartości γ_D dla ontologii źródłowej zawartej w pewnym odwzorowaniu. Szare kółka oznaczają klasy, które

zostały zmapowane na odpowiadające im klasy w ontologii docelowej, której szczegóły nie są kluczowe dla zilustrowania głównej idei γ_D .

Dzięki funkcji γ_D do oceny pojedynczego mapowania możliwe stało się zdefiniowanie **funkcji Λ_C do oceny odwzorowań ontologii na poziomie klas w oparciu o głębokość mapowanych klas**. Posiada ona sygnaturę $\Lambda_C : \tilde{O} \times \tilde{O} \times \tilde{AL}_C \rightarrow \mathbb{R}^+$ oraz zwraca rzeczywistą wartość, która odzwierciedla stopień, w jakim dopasowane klasy są głęboko osadzone w ontologiach. Wyższe wartości Λ_C wskazują, że mapowania łączą klasy umieszczone głębiej w hierarchii, co oznacza bardziej szczegółowe i wyspecjalizowane dopasowanie. Z kolei niższe wartości sugerują, że mapowania obejmują bardziej ogólne i mniej szczegółowe klasy.

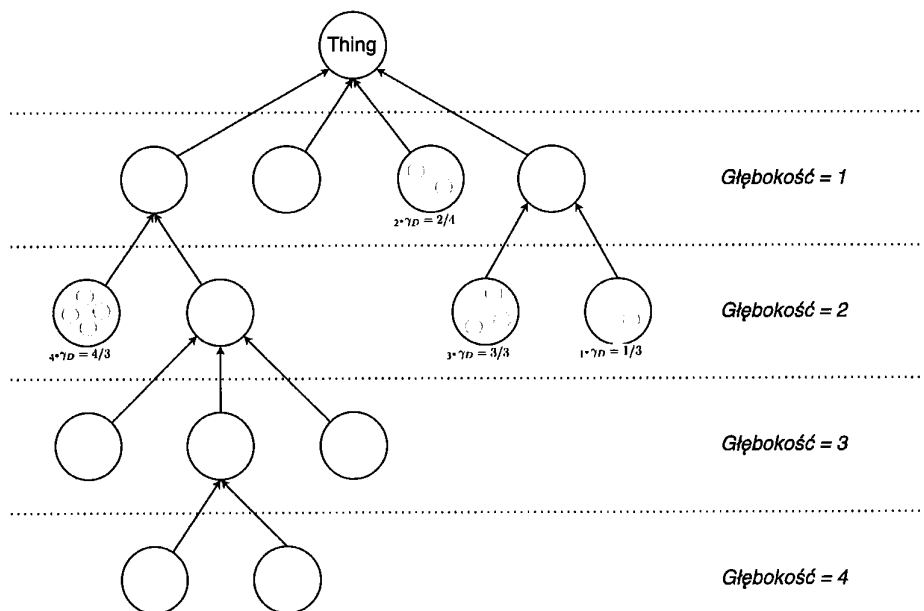
Użyteczność funkcji γ_D pozwoliła również na zdefiniowanie **funkcji Λ_I do oceny odwzorowań ontologii na poziomie instancji w oparciu o głębokość przypisanych klas** o sygnaturze $\Lambda_I : \tilde{O} \times \tilde{O} \times \tilde{AL}_I \rightarrow \mathbb{R}^+$. Podobnie jak Λ_C , Λ_I zwraca wartość rzeczywistą, która określa, czy mapowane instancje są powiązane z klasami znajdującymi się głęboko w hierarchii ontologii. Wyższe wartości Λ_I wskazują, że mapowane instancje są związane z bardziej wyspecjalizowanymi i szczegółowymi klasami, natomiast niższe wartości sugerują, że mapowania dotyczą instancji powiązanych z ogólniejszymi i mniej szczegółowymi klasami.



Rysunek 3: Przykład wartości funkcji γ_D dla klas z pojedynczej ontologii

Rysunek 4 pokazuje przykładowe wyniki przypisane do instancji klas w obrębie konkretnej ontologii. Szare kółka oznaczają instancje zmapowane na odpowiadające im instancje w drugiej ontologii, a ich otaczające pojęcia są przedstawione jako białe kółka wokół szarych. Rysunek pokazuje również skumulowane wartości γ_D , wskazujące wyniki oparte na głębokości dla każdej grupy instancji.

Drugą grupą funkcji oceny jakości odwzorowań zaproponowanych w monografii [50] w Rozdziale 6.1.2 są metody oparte na ciągłości odwzorowań. W sytuacji gdy pewne odwzorowanie ontologii zawiera mapowania klas, które nie mają wyraźnego związku, mogą się one wydawać niezorganizowane i chaotyczne, co sugeruje, że zostały utworzone przypadkowo. Przykładem takiej sytuacji mogą być dwie odrębne klasy mają te same nazwy (homonimy). W takich przypadku, jeśli wspólna nazwa jest jedynym wspólnym atrybutem tych klas, staje się mało prawdopodobne, aby ich klasy nadrzędne i podrzędne były jakkolwiek powiązane. W konsekwencji mapowanie między nimi wydaje się w najlepszym razie przypadkowe. Odwrotna sytuacja dotyczy mapowań klas, które są częścią tych samych poddrzew - tego rodzaju odwzorowania wydają się być bardziej spójne i tym samym, lepsze. Powyższe spostrzeżenia stały się podstawą do zdefiniowania funkcji Γ_C dotyczącej poziomu klas, oraz funkcja Γ_I dla poziomu instancji.

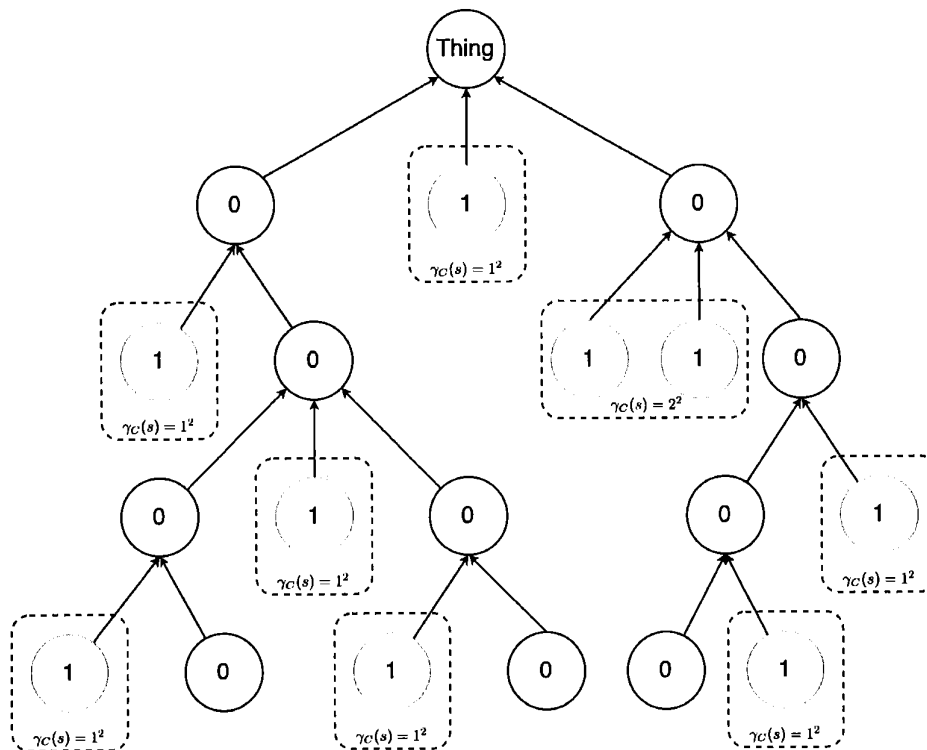


Rysunek 4: Przykład wykorzystania wartości γ_D od oceny odwzorowań instancji

Do zdefiniowania obydwu konieczne było wprowadzenie funkcji pomocniczych. Pierwsza z nich $subtrees_O(O_1)$ dla zadanej ontologii O_1 zwracała zbiór zbiorów klas ze zbioru C będące osobnymi poddrzewami taksonomii klas (bez korzenia). Każde z takich poddrzew można ocenić korzystając z drugiej z pomocniczych funkcji oznaczonej γ_C . Na Rysunkach 5 i 6 zostały przedstawione wartości funkcji γ_C dla wybranych poddrzew z przykładowej ontologii.

Założmy, że klasy zaznaczone na szaro na obu rysunkach reprezentują części dwóch odrębnych, konkurencyjnych odwzorowań — klasy te tworzą różne poddrzewa, zaznaczone liniami przerywanymi. Sformułowanie funkcji γ_C , umożliwiło opracowanie **funkcji Γ_C do oceny odwzorowań ontologii na poziomie klas w oparciu o ciągłość klas**. Posiada ona sygnaturę $\Gamma_C : \tilde{O} \times \tilde{O} \times \tilde{A}L_C \rightarrow \mathbb{R}^+$, a jej wyższe wartości wskazują, że klasy z ocenianego odwzorowania są położone blisko siebie.

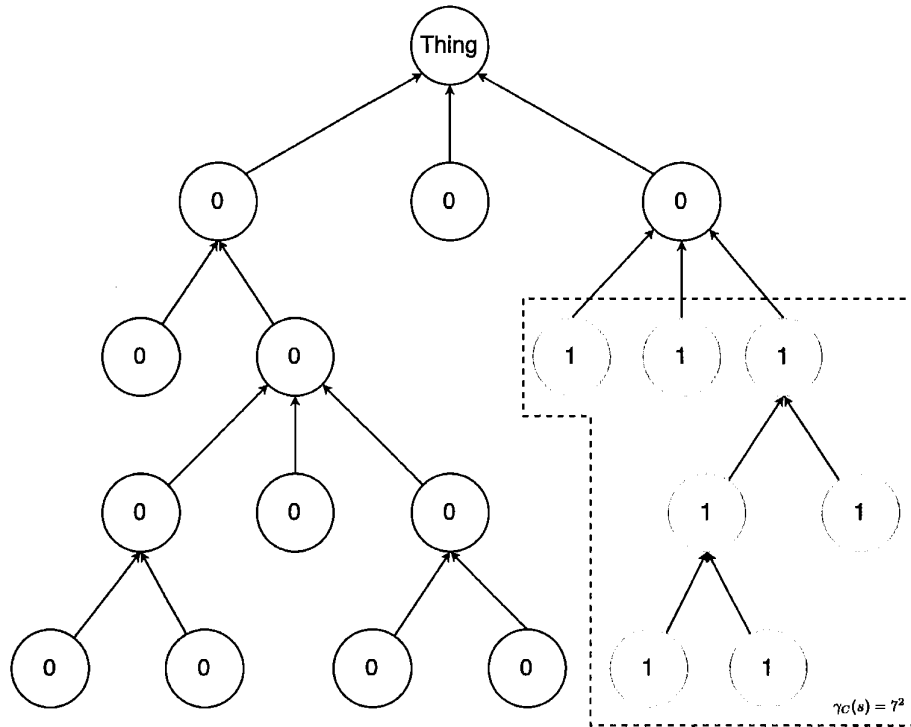
bie w hierarchii klas. Niższe wartości Γ_C oznaczają, że oceniane odwzorowanie zawiera mapowania klas rozmieszczonych losowo w całej hierarchii.



Rysunek 5: Przykład wartości funkcji γ_C dla klas z pojedynczej ontologii

W wyniku użycia Γ_C , odwzorowanie na Rysunku 5 uzyskuje wynik, wynoszący 11, podczas gdy odwzorowanie na Rysunku 6 osiąga znacznie wyższy rezultat równy 49. Ta różnica podkreśla kluczową rolę w ocenie odwzorowań, nie tylko na podstawie liczby mapowań, ale przede wszystkim poprzez analizę ich ciągłości, mierzonej funkcją γ_C . W efekcie, choć dopasowanie na Rysunku 5 zawiera więcej mapowań, jest uznawane za gorsze od tego na Rysunku 6, które wykazuje większą ciągłość między powiązаныmi klasami.

Druga z metod oceny jakości odwzorowania ontologii bazująca na ciągłości klas funkcjonuje na poziomie instancji. Klasy pełnią rolę strukturyzującą wybraną dziedzinę problemową, pozwalając na kategoryzację instancji według wyróżnionych wspólnych cech. Same instancje nie mają natomiast samodzielnego znaczenia. Ich wartość semantyczna ujawnia się dopiero w kontekście klas do których są przypisane.



Rysunek 6: Przykład wartości funkcji Γ_C dla klas z pojedynczej ontologii

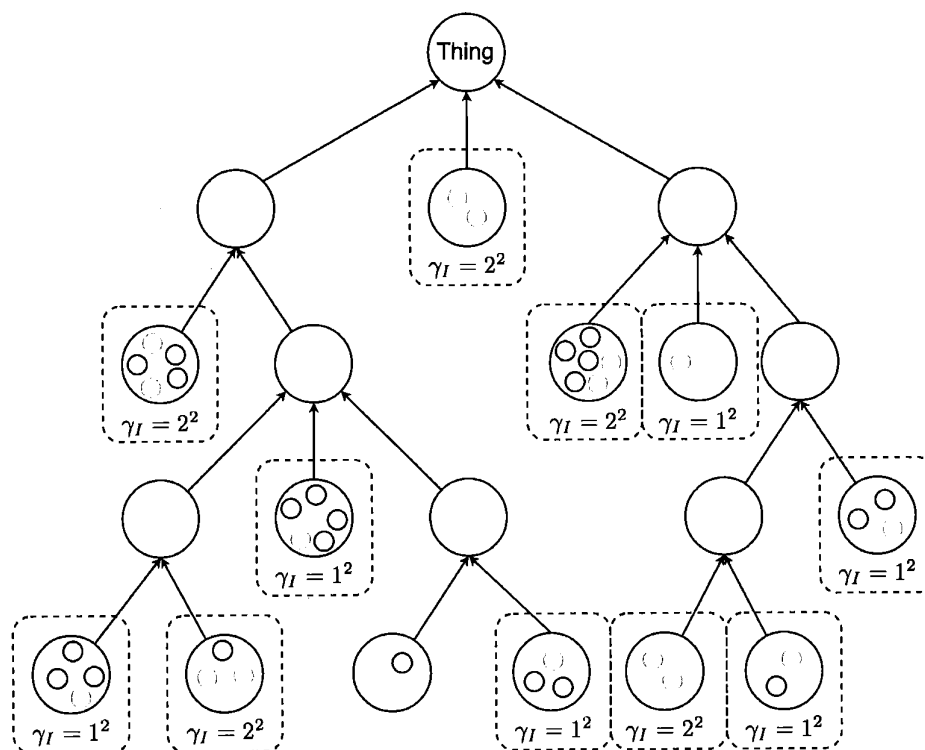
W Rozdziale 2 omawianej monografii habilitacyjnej znajduje się definicja odwzorowania ontologii O_1 i O_2 na poziomie instancji, oznaczona jako $Align_I(O_1, O_2)$. Jest to zbiór elementów $AL_{O_1, O_2}(c_1, c_2)$, obejmujący mapowania instancji klas c_1 i c_2 . Na tej podstawie autor zaproponował funkcję Γ_I do oceny odwzorowań ontologii na poziomie instancji, opartą na ciągłości klas, do których przypisane są mapowane instancje. Funkcja ta ma sygnaturę $\Gamma_I : \tilde{O} \times \tilde{O} \times \tilde{ALI} \rightarrow \mathbb{R}^+$ i umożliwia obliczenie wartości liczbowej, odzwierciedlającej poziom ciągłości klas, do których przypisane są mapowane instancje. Wyższe wartości Γ_I wskazują na większą spójność mapowanych instancji w ramach blisko powiązanych klas, natomiast niższe wartości sugerują bardziej przypadkowe, rozproszone mapowania. Funkcja Γ_I opiera się na funkcji γ_I , służącej do oceny jakości odwzorowania zbioru instancji $AL_{O_1, O_2}(c_1, c_2)$ dla wybranych klas c_1 i c_2 . Wszystkie te funkcje zostały szczegółowo opisane w Rozdziale 6.1.2 monografii habilitacyjnej.

Na Rysunkach 7 i 8 przedstawiono dwa różne odwzorowania tej samej ontologii. Instancje oraz przypisane do nich klasy zostały oznaczone jako małe kółka umieszczone wewnątrz większych okręgów, gdzie szare kółka symbolizują instancje uwzględnione w odwzorowaniu. Pod każdą klasą, której instancje zostały odwzorowane, umieszczono wyznaczoną wartość γ_I .

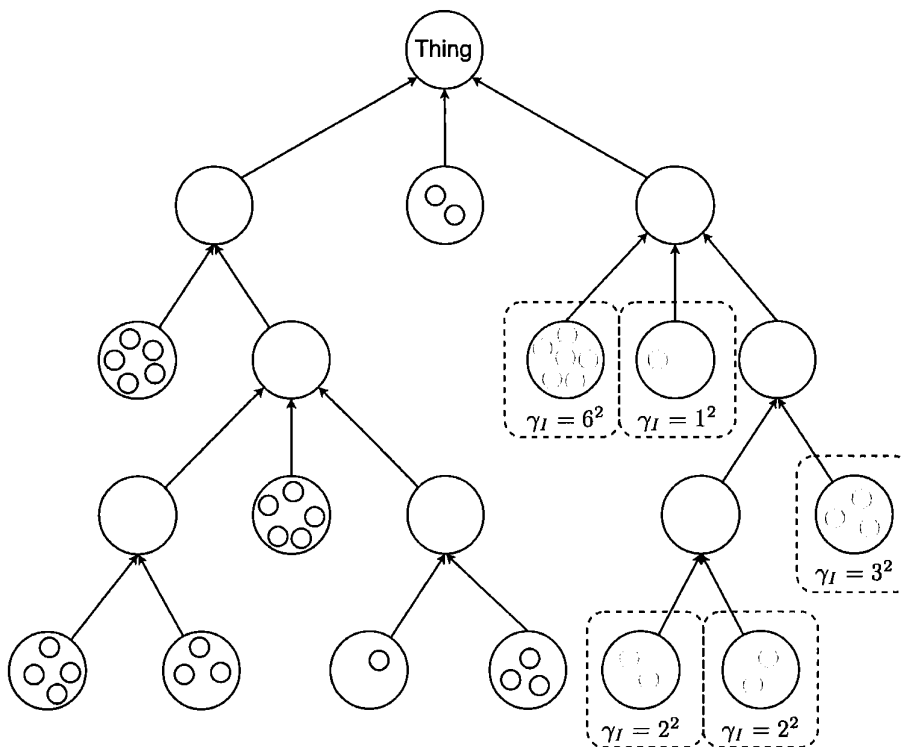
Rysunek 7 przedstawia przypadek, w którym co najwyżej dwie instancje z tej samej klasy zostały dopasowane, co prowadzi do rozproszonego odwzorowania. W rezultacie suma wartości γ_I wynosi 26. Natomiast Rysunek 8 ilustruje odwzorowanie, w którym wszystkie instancje kilku pojęć

zostały kompleksowo odwzorowane, co odzwierciedla bardziej spójne mapowanie. Suma wartości γ_I w tym przypadku wynosi 53, co podkreśla większą integralność odwzorowania w porównaniu do pierwszego przypadku.

Wyższa ocena odwzorowania przedstawionego na Rysunku 8 wynika z jego większej użyteczności z perspektywy użytkownika. Głównym celem dopasowania ontologii jest wzbogacenie i integracja wiedzy specjalistycznej, dlatego jakość odwzorowania powinna być oceniana nie tylko pod względem liczby mapowań. Bardziej ekspresywne odwzorowanie, takie jak to zaprezentowane na Rysunku 8, lepiej wspiera proces integracji wiedzy oraz dokładniej odzwierciedla złożoność modelowanej dziedziny, nawet jeśli obejmuje mniejszą liczbę elementów. To założenie stanowiło podstawę do zdefiniowania funkcji γ_I .



Rysunek 7: Przykład wartości funkcji Γ_I dla instancji z pojedynczej ontologii



Rysunek 8: Przykład wartości funkcji Γ_I dla instancji z pojedynczej ontologii

W Rozdziale 6.2 monografii przeprowadzono eksperymentalną weryfikację opracowanych funkcji. Ze względu na odmienny charakter funkcji Λ_C , Λ_I , Γ_C i Γ_I , ich bezpośrednie porównanie z metodami opartymi na odwzorowaniach referencyjnych nie jest możliwe. Obie grupy metod mają różne cele i zastosowania, co uniemożliwia jednoznaczną ocenę, która z nich jest lepsza. Dlatego też zdecydowano się na ocenę użyteczności zaproponowanych funkcji w dwóch scenariuszach: (1) w kontekście danych testowych udostępnianych przez Ontology Alignment Evaluation Initiative (OAEI) oraz (2) w zadaniach zarządzania ewolucją ontologii i ich odwzorowań. Taki wybór umożliwił ocenę skuteczności funkcji zarówno w standardowych scenariuszach dopasowania ontologii, jak i w bardziej dynamicznych warunkach, wymagających bieżącej aktualizacji odwzorowań.

W pierwszym scenariuszu, dla par ontologii z zestawu danych testowych OAEI, wygenerowano odwzorowania przy użyciu narzędzia LogMap ([31]), które według literatury ([30]) jest jednym z wiodących rozwiązań w tej dziedzinie. Następnie dla każdego wygenerowanego odwzorowania obliczono standardowe miary jakości: Precyzję i Kompletność, a także wartości funkcji Λ_C , Λ_I , Γ_C i Γ_I . Uzyskane wyniki poddano analizie statystycznej w celu zbadania współzależności między poszczególnymi miarami.

Analiza wykazała istotne wartości współczynnika korelacji międzyklasowej pomiędzy Kompletnością a funkcjami Λ_C i Γ_C , co sugeruje, że funkcje te dobrze odzwierciedlają pełność odwzorowań. Natomiast Precyzja wykazała niskie wartości korelacji z Λ_C i Γ_C , co wskazuje, że te funkcje nie są silnie powiązane z aspektami dokładności odwzorowań. W przypadku funkcji Λ_I i Γ_I zaobserwowano dodatnią korelację z Kompletnością, przy czym Precyzja miała słabą korelację z Λ_I ,

ale umiarkowaną dodatnią korelację z Γ_I . Dokładny opis procedury badawczej oraz szczegółowe omówienie wyników i analizy statystycznej zostały przedstawione w Rozdziale 6.2.1 monografii.

Drugi etap weryfikacji eksperymentalnej dotyczył oceny przydatności funkcji Λ_C , Λ_I , Γ_C i Γ_I w kontekście ewolucji ontologii. W ramach tego etapu, na podstawie ontologii pobranych ze zbioru testowego opracowanego przez OAEI, przygotowano ich zmodyfikowane wersje symulujące proces ewolucji. Następnie dla ontologii w ich początkowym stanie oraz po wprowadzeniu zmian wygenerowano odwzorowania od podstaw, korzystając z narzędzia LogMap ([30]).

W kolejnym kroku zastosowano algorytmy opisane w Rozdziale 5 monografii habilitacyjnej (omówione wcześniej w autoreferacie jako Osiągnięcie 5), co pozwoliło na uzyskanie zaktualizowanych wersji odwzorowań, uwzględniających zmiany wprowadzone podczas ewolucji ontologii. Ostatnim krokiem była ocena jakości wszystkich uzyskanych odwzorowań z wykorzystaniem funkcji Λ_C , Λ_I , Γ_C i Γ_I .

Przeprowadzona analiza statystyczna wyników wykazała, że jakość odwzorowań generowanych od podstaw oraz tych zaktualizowanych za pomocą zaproponowanych algorytmów była porównywalna. Wyniki te potwierdzają zarówno skuteczność algorytmów aktualizacji odwzorowań przedstawionych w Rozdziale 5 monografii, jak i funkcji oceny jakości odwzorowań opisanych w Rozdziale 6.1.

Eksperyment osiągnął dwa główne cele. Po pierwsze, dostarczył kompleksowej oceny skuteczności metod utrzymania odwzorowań ontologii w obliczu ich ewolucji, co omówiono w Rozdziale 5 monografii habilitacyjnej [50]. Badanie wykazało, że możliwe jest skuteczne zaktualizowanie odwzorowania ontologii na podstawie informacji o ich zmianach, bez konieczności ponownego uruchamiania czasochłonnych procesów mapowania wymagających przetwarzania całych ontologii. Po drugie, eksperyment potwierdził praktyczną użyteczność funkcji oceny jakości odwzorowań ontologii. Jest to szczególnie istotne w sytuacjach, gdy tradycyjne miary, takie jak Precyzja czy Kompletność, nie mogą zostać zastosowane, np. z powodu braku referencyjnych odwzorowań.

Podsumowując, opracowane funkcje Λ_C , Λ_I , Γ_C i Γ_I stanowią nowatorskie narzędzie do porównywania odwzorowań ontologii, opierając się wyłącznie na zawartości samych ontologii. Ich główną zaletą jest niezależność od wcześniej przygotowanych odwzorowań referencyjnych, co pozwala na przewyższenie ograniczeń dotychczasowych metod i tradycyjnych miar, takich jak Precyzja i Kompletność. Dzięki temu opracowane podejście znacząco wzbogaca zestaw narzędzi dostępnych w dziedzinie zarządzania ontologiami i ich dopasowaniami.

4.1.4 Podsumowanie

Monografia habilitacyjna zatytułowana „Formal Methods for Managing and Processing Ontology Alignments” ([50]) stanowi podsumowanie wieloletnich badań autora, prowadzonych na przestrzeni ostatnich dziesięciu lat. Wyniki tych prac były publikowane w renomowanych czasopiśmie o wysokim wskaźniku IF oraz prezentowane na prestiżowych konferencjach co najmniej z listy B rankingu Core. Opracowane w monografii rozwiązania odpowiadają na pytania badawcze, sformułowane w sekcji [\[4.1.1\]](#) niniejszego autoreferatu i powtórzonych poniżej:

- *Q1*: W jaki sposób wyznaczyć odwzorowanie ontologii na poziomie klas, relacji i instancji?
- *Q2*: W jaki sposób zarządzać procesem ewolucji ontologii tak, aby możliwe było utrzymanie poprawności ich odwzorowań?

- *Q3*: W jaki sposób ocenić jakość odwzorowań ontologii bazując jedynie na zawartości ontologii?

Pierwsza część monografii, obejmująca opis osiągnięć 1 i 2, przedstawia formalny model ontologii, obejmujący definicję wszystkich kluczowych elementów, takich jak klasy, relacje i instancje, wraz z ich wewnętrznymi strukturami. Wprowadzono również innowacyjne pojęcia semantyki atrybutów i relacji, bazujące na rachunku zdań. Definicje te są rozwinięciem wcześniejszych prac autora, zapoczątkowanych w artykule [53]. W monografii opisano metody przypisywania semantyki atrybutom (przedstawione w [24]) oraz relacjom (zaprezentowane w [25]), które stanowią podstawę dla skutecznego odwzorowywania ontologii. Kolejnym istotnym osiągnięciem jest formalna definicja odwzorowań ontologii oraz kryteriów wykrywania niespójności w ich strukturach. Osiągnięcia 1 i 2 stanowią zatem podstawę dla całej monografii.

Kolejna część książki, obejmująca opis osiągnięcia 3, skupia się na metodach odwzorowywania ontologii z wykorzystaniem logiki rozmytej. Metody te, wcześniej opisane w pracach [25], [26] i [39], obejmują procesy odwzorowywania na poziomie klas, relacji i instancji. Wprowadzono różnorodne funkcje podobieństwa z towarzyszącymi im funkcjami rozmywania oraz regułami wnioskowania rozmytego. Eksperymentalna weryfikacja tych metod potwierdziła ich skuteczność w porównaniu z innymi rozwiązaniami obecnymi w literaturze. Osiągnięcie 3 jest zatem bezpośrednim wkładem autora w problematykę poruszoną w pierwszym pytaniu badawczym *Q1*.

Następnie zaprezentowano model ewolucji ontologii, uwzględniający definicję repozytoriów, dzienników zmian oraz funkcji opisujących wprowadzone modyfikacje. W tej części zawarto również definicje funkcji oceniających istotność zmian na poziomie klas, relacji i instancji, bazujących na wcześniejszych pracach autora ([36],[35], [41], [51]). Kluczowym osiągnięciem jest opracowanie metod walidacji i aktualizacji odwzorowań w oparciu o opisy ewolucji ontologii ([37], [38]). Autor proponuje również metodę pomiaru przyrostu wiedzy o interoperacyjności ontologii oraz jej zastosowanie w ocenie jakości odwzorowań ([40]). Osiągnięcia 4 i 5 opisane w tej części monografii są bezpośrednią odpowiedzią na pytanie badawcze *Q2*.

W ostatnim rozdziale monografii przedstawiono metody oceny jakości odwzorowań ontologii (stanowiące osiągnięcie 6), w tym miary oparte na głębokości klas oraz ciągłości odwzorowań. Metody te, opisane w pracach [52] oraz [27], zostały zweryfikowane eksperymentalnie w porównaniu do tradycyjnych metryk, takich jak Precyzja i Kompletność. Wykazano ich przydatność w procesach zarządzania ewolucją ontologii, szczególnie w kontekście aktualizacji odwzorowań. Osiągnięcie 6 jest zatem odpowiedzią habilitanta na pytanie badawcze *Q3*.

Monografia przedstawia kompleksowy zestaw metod zarządzania i przetwarzania odwzorowań ontologii, oferując narzędzia wspierające formalne modelowanie, procesy aktualizacji oraz oceny jakości odwzorowań. Dzięki zastosowanemu formalnemu podejściu opracowane rozwiązania cechują się uniwersalnością, co umożliwi ich zastosowanie w różnych środowiskach, niezależnie od używanego formatu reprezentacji ontologii czy specyfiki dziedziny, której dotyczą. Autor monografii wskazuje również na potencjalne kierunki dalszych badań. W szczególności zwraca uwagę na możliwość wykorzystania mechanizmów opartych na głębokich sieciach neuronowych i uczeniu maszynowym do automatyzacji procesu przypisywania semantyki atrybutów i relacji

Lista publikacji autora związanych z tematem monografii

- [24] Hnatkowska, A. Kozierekiewicz, M. Pietranik, "Semi-automatic definition of attribute semantics for ontology integration," *IEEE Access*, 8, 107272–107284, 2020. **Punktacja MNiSW: 100, Impact Factor (2020): 3.367, Impact Factor (2023): 3.9**
- [25] B. Hnatkowska, A. Kozierekiewicz, M. Pietranik, "Fuzzy based approach to ontology relations alignment," 2021 IEEE International Conference on Fuzzy Systems, 1–7, 2021. **Punktacja MNiSW: 70, Klasyfikacja CORE: B**
- [26] B. Hnatkowska, A. Kozierekiewicz, M. Pietranik, "Fuzzy logic framework for ontology instance alignment," *ICCS 2022: International Conference On Computational Science*, 653–666, Springer, 2022. **Punktacja MNiSW: 140, Klasyfikacja CORE: A**
- [27] B. Hnatkowska, A. Kozierekiewicz, M. Pietranik, H. B. Truong, "Assessing ontology alignments on the level of instances" *ICCCI 2021: International Conference on Computational Collective Intelligence*, 42–52, Springer, 2021. **Punktacja MNiSW: 70, Klasyfikacja CORE: B**
- [35] A. Kozierekiewicz, M. Pietranik, "A formal framework for ontology evolution," *ACIIDS 2019: Asian Conference on Intelligent Information and Database Systems*, 16–27, Springer, 2019. **Punktacja MNiSW: 70, Klasyfikacja CORE: B**
- [36] A. Kozierekiewicz, M. Pietranik, "Triggering Ontology Alignment Revalidation Based on the Degree of Change Significance on the Ontology Concept Level" *BIS 2019: International Conference on Business Information Systems*, 137–148, Springer, 2019. **Punktacja MNiSW: 70, Klasyfikacja CORE: B**
- [37] A. Kozierekiewicz, M. Pietranik, "Updating ontology alignment on concept level based on evolution," *ADBIS 2019: European Conference on Advances in Databases and Information Systems*, 201–214, Springer, 2019. **Punktacja MNiSW: 70, Klasyfikacja CORE: B**
- [38] A. Kozierekiewicz, M. Pietranik, "Updating ontology alignment on relation level based on evolution," *ENASE 2020: International Conference on Evaluation of Novel Approaches to Software Engineering*, 241–248, SciTePress, 2020. **Punktacja MNiSW: 70, Klasyfikacja CORE: B**
- [39] A. Kozierekiewicz, M. Pietranik, W. Jankowiak, "Fuzzy logic framework for ontology concepts alignment," *ACIIDS 2023: Asian Conference on Intelligent Information and Database Systems*, 16–27, Springer, 2023. **Punktacja MNiSW: 70, Klasyfikacja CORE: B**
- [40] A. Kozierekiewicz, M. Pietranik, K. Kania, "A Method for Estimating Potential Knowledge Increase after Updating Ontology Mapping," *ENASE 2021: International Conference on Evaluation of Novel Approaches to Software Engineering*, 173–180, SciTePress, 2021. **Punktacja MNiSW: 70, Klasyfikacja CORE: B**
- [41] A. Kozierekiewicz, M. Pietranik, L. T. Nguyen, "Updating ontology alignment on instance level based on evolution," *DEXA 2020: International Conference*, 301–311, Springer, 2020. **Punktacja MNiSW: 70, Klasyfikacja CORE: B**
- [50] A. Kozierekiewicz, M. Pietranik, "Methods of managing the evolution of ontologies and alignments," *Applied Intelligence*, 2023. **Punktacja MNiSW: 100, Impact Factor (2023): 5.3**
- [51] M. Pietranik, A. Kozierekiewicz, M. Wesołowski, "Assessing ontology mappings on concepts and instances," *IEEE Access*, 8, 174845–174859, 2020. **Punktacja MNiSW: 100, Impact Factor (2019): 4.098, Impact Factor (2023): 3.9**

- [53] Pietranik, N. T. Nguyen, “A multi-attribute based framework for ontology aligning,” *Neurocomputing*, 146, 276–290, 2014. **Punktacja MNiSW za 2022-2025: 140, Impact Factor (2014): 2.083, Impact Factor (2023): 5.5**

Wszystkie powyższe publikacje powstały po uzyskaniu stopnia doktora.

Literatura

- [1] S. M. Alghamdi, F. Zhapa-Camacho, and R. Hoehndorf. A-lion - alignment learning through inconsistency negatives of the aligned ontologies. In *Proceedings of the 17th International Workshop on Ontology Matching (OM 2022) co-located with the 21th International Semantic Web Conference (ISWC 2022), Hangzhou, China, held as a virtual conference, October 23, 2022*, volume 3324 of *CEUR Workshop Proceedings*, pages 137–144. CEUR-WS.org, 2022.
- [2] Y. An, A. Kalinowski, and J. Greenberg. Otmaponto: optimal transport-based ontology matching. In *Proceedings of the 16th International Workshop on Ontology Matching co-located with the 20th International Semantic Web Conference (ISWC 2021), Virtual conference, October 25, 2021*, volume 3063 of *CEUR Workshop Proceedings*, pages 185–192. CEUR-WS.org, 2021.
- [3] A. Azim, L. Clark, C. Lau, M. Cobb, and K. Jenner. Grounding ontologies with pre-trained large language models for activity based intelligence. In *Signal Processing, Sensor/Information Fusion, and Target Recognition XXXIII*, volume 13057, pages 219–229. SPIE, 2024.
- [4] S. D. Cardoso, M. Da Silveira, and C. Pruski. Construction and exploitation of an historical knowledge graph to deal with the evolution of ontologies. *Knowledge-Based Systems*, 194:105508, 2020.
- [5] M. Cheatham, C. Pesquita, D. Oliveira, and H. B. McCurdy. The properties of property alignment on the semantic web. *International Journal of Metadata, Semantics and Ontologies*, 13(1):42–56, 2018.
- [6] H. Chen. Large knowledge model: Perspectives and challenges. *arXiv preprint arXiv:2312.02706*, 2023.
- [7] J. Chen, P. Hu, E. Jimenez-Ruiz, O. M. Holter, D. Antonyrajah, and I. Horrocks. Owl2vec*: Embedding of owl ontologies. *Machine Learning*, 110(7):1813–1845, 2021.
- [8] G. O. Consortium. The gene ontology (go) database and informatics resource. *Nucleic acids research*, 32(suppl_1):D258–D261, 2004.
- [9] V. T. Da Silva, J. S. Dos Santos, R. Thiago, E. Soares, and L. G. Azevedo. Owl ontology evolution: understanding and unifying the complex changes. *The Knowledge Engineering Review*, 37:e10, 2022.
- [10] J. David and J. Euzenat. On fixing semantic alignment evaluation measures. In *Proc. 3rd ISWC workshop on ontology matching (OM)*, pages 25–36, 2008.
- [11] S. Efeoglu. Graphmatcher: a graph representation learning approach for ontology matching. In *Proceedings of the 17th International Workshop on Ontology Matching (OM 2022) co-located with the 21th International Semantic Web Conference (ISWC 2022), Hangzhou, China, held as a virtual conference, October 23, 2022*, volume 3324 of *CEUR Workshop Proceedings*, pages 174–180. CEUR-WS.org, 2022.

- [12] J. Euzenat. Semantic precision and recall for ontology alignment evaluation. In *IJCAI 2007, Proceedings of the 20th International Joint Conference on Artificial Intelligence, Hyderabad, India, January 6-12, 2007*, pages 348–353, 2007.
- [13] D. Faria, C. Pesquita, E. Santos, M. Palmonari, I. F. Cruz, and F. M. Couto. The agreement-makerlight ontology matching system. In *On the Move to Meaningful Internet Systems: OTM 2013 Conferences: Confederated International Conferences: CoopIS, DOA-Trusted Cloud, and ODBASE 2013, Graz, Austria, September 9-13, 2013. Proceedings*, pages 527–541. Springer, 2013.
- [14] D. Fleischhacker and H. Stuckenschmidt. Implementing semantic precision and recall. In *Proceedings of the 4th International Workshop on Ontology Matching (OM-2009) collocated with the 8th International Semantic Web Conference (ISWC-2009) Chantilly, USA, October 25, 2009*, volume 551 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2009.
- [15] G. Flouris, D. Manakanatas, H. Kondylakis, D. Plexousakis, and G. Antoniou. Ontology change: classification and survey. *The Knowledge Engineering Review*, 23(2):117–152, 2008.
- [16] C. Golbreich, E. K. Wallace, and P. F. Patel-Schneider. Owl 2 web ontology language new features and rationale. *W3C working draft, W3C (June 2009) <http://www.w3.org/TR/2009/WD-owl2-new-features-20090611>*, 2009.
- [17] F. Gosselin and A. Zouaq. SORBET: A siamese network for ontology embeddings using a distance-based regression loss and BERT. In *The Semantic Web - ISWC 2023 - 22nd International Semantic Web Conference, Athens, Greece, November 6-10, 2023, Proceedings, Part I*, volume 14265 of *Lecture Notes in Computer Science*, pages 561–578. Springer, 2023.
- [18] B. C. Grau, I. Horrocks, B. Motik, B. Parsia, P. Patel-Schneider, and U. Sattler. Owl 2: The next step for owl. *Journal of Web Semantics*, 6(4):309–322, 2008. Semantic Web Challenge 2006/2007.
- [19] T. R. Gruber. A translation approach to portable ontology specifications. *Knowledge acquisition*, 5(2):199–220, 1993.
- [20] B. G. H. Happi, G. F. Pelap, D. Symeonidou, and P. Larmande. Dlinker results for OAEI 2022. In *Proceedings of the 17th International Workshop on Ontology Matching (OM 2022) co-located with the 21th International Semantic Web Conference (ISWC 2022), Hangzhou, China, held as a virtual conference, October 23, 2022*, volume 3324 of *CEUR Workshop Proceedings*, pages 166–173. CEUR-WS.org, 2022.
- [21] M. Hartung, A. Groß, and E. Rahm. Conto-diff: generation of complex evolution mappings for life science ontologies. *Journal of biomedical informatics*, 46(1):15–32, 2013.
- [22] M. Hartung, T. Kirsten, and E. Rahm. Analyzing the evolution of life science ontologies and mappings. In *Data Integration in the Life Sciences: 5th International Workshop, DILS 2008, Evry, France, June 25-27, 2008. Proceedings 5*, pages 11–27. Springer, 2008.
- [23] S. Hertling and H. Paulheim. Olala: Ontology matching with large language models. In *Proceedings of the 12th Knowledge Capture Conference 2023*, pages 131–139, 2023.
- [24] B. Hnatkowska, A. Kozierekiewicz, and M. Pietranik. Semi-automatic definition of attribute semantics for the purpose of ontology integration. *IEEE Access*, 8:107272–107284, 2020.
- [25] B. Hnatkowska, A. Kozierekiewicz, and M. Pietranik. Fuzzy based approach to ontology relations alignment. In *2021 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, pages 1–7. IEEE, 2021.
- [26] B. Hnatkowska, A. Kozierekiewicz, and M. Pietranik. Fuzzy logic framework for ontology instance alignment. In *Computational Science-ICCS 2022: 22nd International Conference*,

- London, UK, June 21–23, 2022, *Proceedings, Part II*, pages 653–666. Springer International Publishing Cham, 2022.
- [27] B. Hnatkowska, A. Kozierekiewicz, M. Pietranik, and H. B. Truong. Assessing ontology alignments on the level of instances. In *Computational Collective Intelligence - 13th International Conference, ICCCI 2021, Rhodes, Greece, September 29 - October 1, 2021, Proceedings*, volume 12876 of *Lecture Notes in Computer Science*, pages 42–52. Springer, 2021.
- [28] L. Hollink, M. Van Assem, S. Wang, A. Isaac, and G. Schreiber. Two variations on ontology alignment evaluation: Methodological issues. In *The Semantic Web: Research and Applications: 5th European Semantic Web Conference, ESWC 2008, Tenerife, Canary Islands, Spain, June 1-5, 2008 Proceedings 5*, pages 388–401. Springer, 2008.
- [29] Y. Jiang, X. Wang, and H.-T. Zheng. A semantic similarity measure based on information distance for ontology alignment. *Information Sciences*, 278:76–87, 2014.
- [30] E. Jimenez-Ruiz. Logmap family participation in the oaei 2021. In *16th International Workshop on Ontology Matching (OM 2021)*, volume 3063, pages 175–177, January 2021. Copyright © 2021 for the individual papers by the papers’ authors.
- [31] E. Jiménez-Ruiz and B. Cuenca Grau. Logmap: Logic-based and scalable ontology matching. In *The Semantic Web–ISWC 2011: 10th International Semantic Web Conference, Bonn, Germany, October 23-27, 2011, Proceedings, Part I 10*, pages 273–288. Springer, 2011.
- [32] A. M. Khattak, K. Latif, and S. Lee. Change management in evolving web ontologies. *Knowledge-Based Systems*, 37:1–18, 2013.
- [33] A. M. Khattak, Z. Pervez, W. A. Khan, A. M. Khan, K. Latif, and S. Lee. Mapping evolution of dynamic web ontologies. *Information Sciences*, 303:101–119, 2015.
- [34] D. Kossack, N. Borg, L. Knorr, and J. Portisch. TOM matcher results for OAEI 2021. In *Proceedings of the 16th International Workshop on Ontology Matching co-located with the 20th International Semantic Web Conference (ISWC 2021), Virtual conference, October 25, 2021*, volume 3063, pages 193–198, 2021.
- [35] A. Kozierekiewicz and M. Pietranik. A formal framework for the ontology evolution. In *Intelligent Information and Database Systems: 11th Asian Conference, ACIIDS 2019, Yogyakarta, Indonesia, April 8–11, 2019, Proceedings, Part I 11*, pages 16–27. Springer International Publishing, 2019.
- [36] A. Kozierekiewicz and M. Pietranik. Triggering ontology alignment revalidation based on the degree of change significance on the ontology concept level. In *Business Information Systems: 22nd International Conference, BIS 2019, Seville, Spain, June 26–28, 2019, Proceedings, Part I 22*, pages 137–148. Springer International Publishing, 2019.
- [37] A. Kozierekiewicz and M. Pietranik. Updating ontology alignment on the concept level based on ontology evolution. In *Advances in Databases and Information Systems: 23rd European Conference, ADBIS 2019, Bled, Slovenia, September 8–11, 2019, Proceedings 23*, pages 201–214. Springer International Publishing, 2019.
- [38] A. Kozierekiewicz. and M. Pietranik. Updating ontology alignment on the relation level based on ontology evolution. In *Proceedings of the 15th International Conference on Evaluation of Novel Approaches to Software Engineering - ENASE*, pages 241–248. INSTICC, SciTePress, 2020.
- [39] A. Kozierekiewicz, M. Pietranik, and W. Jankowiak. Fuzzy logic framework for ontology concepts alignment. In *Intelligent Information and Database Systems: 15th Asian Conference*,

- ACIIDS 2023, Phuket, Thailand, July 24–26, 2023, Proceedings, Part I 11*, pages 16–27. Springer International Publishing, 2023.
- [40] A. Kozierekiewicz., M. Pietranik., and K. Kania. A method for estimating potential knowledge increase after updating ontology mapping. In *Proceedings of the 16th International Conference on Evaluation of Novel Approaches to Software Engineering - ENASE*, pages 173–180. INSTICC, SciTePress, 2021.
- [41] A. Kozierekiewicz, M. Pietranik, and L. T. Nguyen. Updating ontology alignment on the instance level based on ontology evolution. In *Database and Expert Systems Applications: 31st International Conference, DEXA 2020, Bratislava, Slovakia, September 14–17, 2020, Proceedings, Part II 31*, pages 301–311. Springer International Publishing, 2020.
- [42] C. D. Manning, P. Raghavan, and H. Schütze. *Introduction to information retrieval*. Cambridge University Press, 2008.
- [43] F. McNeill, P. Besana, J. Pane, and F. Giunchiglia. Service integration through structure-preserving semantic matching. In *Cases on Semantic Interoperability for Information Systems Integration: Practices and Applications*, pages 64–82. IGI Global, 2010.
- [44] C. Meilicke. *Alignment incoherence in ontology matching*. PhD thesis, University of Mannheim, 2011.
- [45] D. Miller. An ontology for future airspace system architectures. In *2017 IEEE/AIAA 36th Digital Avionics Systems Conference (DASC)*, pages 1–8. IEEE, 2017.
- [46] S. Neutel and M. H. de Boer. Towards automatic ontology alignment using bert. In *AAAI Spring Symposium: Combining Machine Learning with Knowledge Engineering*, 2021.
- [47] N. F. Noy, M. A. Musen, et al. Algorithm and tool for automated ontology merging and alignment. In *Proceedings of the 17th National Conference on Artificial Intelligence (AAAI-00)*. Available as *SMI technical report SMI-2000-0831*, volume 115. sn, 2000.
- [48] D. Panneer, K. Rangunathan, M. Ramalingam, and L. K. Narayanan. Comparative study on ontology matching tools and methods. In *AIP Conference Proceedings*, volume 2802. AIP Publishing, 2024.
- [49] P.-H. Paris, F. Hamdi, and S. S.-s. Cherfi. A study about the use of owl 2 semantics in rdf-based knowledge graphs. In *The Semantic Web: ESWC 2020 Satellite Events: ESWC 2020 Satellite Events, Heraklion, Crete, Greece, May 31–June 4, 2020, Revised Selected Papers 17*, pages 181–185. Springer, 2020.
- [50] M. Pietranik. *Formal Methods for Managing and Processing Ontology Alignments*. CRC Press, 2024.
- [51] M. Pietranik and A. Kozierekiewicz. Methods of managing the evolution of ontologies and their alignments. *Applied Intelligence*, 2023.
- [52] M. Pietranik, A. Kozierekiewicz, and M. Wesółowski. Assessing ontology mappings on a level of concepts and instances. *IEEE Access*, 8:174845–174859, 2020.
- [53] M. Pietranik and N. T. Nguyen. A multi-attribute based framework for ontology aligning. *Neurocomputing*, 146:276–290, 2014.
- [54] J. Portisch, M. Hladik, and H. Paulheim. Wiktionary matcher. In *CEUR Workshop Proceedings*, volume 2536, pages 181–188. RWTH Aachen, 2020.
- [55] J. Portisch and H. Paulheim. Alod2vec matcher. In *Proceedings of the 13th International Workshop on Ontology Matching co-located with the 17th International Semantic Web Conference, OM@ISWC 2018, Monterey, CA, USA, October 8, 2018*, volume 2288 of *CEUR Workshop Proceedings*, pages 132–137. CEUR-WS.org, 2018.

- [56] C. E. Quintero-Narvaez and R. Monroy. Integrating knowledge graph data with large language models for explainable inference. In *Proceedings of the 17th ACM International Conference on Web Search and Data Mining*, pages 1198–1199, 2024.
- [57] E. Rahm and P. A. Bernstein. A survey of approaches to automatic schema matching. *the VLDB Journal*, 10:334–350, 2001.
- [58] J.-M. Rodrigues, S. Schulz, A. Rector, K. Spackman, B. Üstün, C. G. Chute, V. Della Mea, J. Millar, and K. B. Persson. Sharing ontology between icd 11 and snomed ct will enable seamless re-use and semantic interoperability. In *MEDINFO 2013*, pages 343–346. IOS Press, 2013.
- [59] C. Rosse and J. L. Mejino Jr. The foundational model of anatomy ontology. In *Anatomy ontologies for bioinformatics: principles and practice*, pages 59–117. Springer, 2008.
- [60] P. Roussille and O. Teste. TOMATO: results of the 2022 OAEI evaluation campaign. In *Proceedings of the 17th International Workshop on Ontology Matching (OM 2022)*, volume 3324 of *CEUR Workshop Proceedings*, pages 210–215. CEUR-WS.org, 2022.
- [61] J. S. Salamon, C. C. Reginato, and M. P. Barcellos. Ontology integration approaches: A systematic mapping. In *ONTOBRAS*, pages 161–172, 2018.
- [62] N. Sassi, W. Jaziri, and S. Alharbi. Supporting ontology adaptation and versioning based on a graph of relevance. *Journal of Experimental & Theoretical Artificial Intelligence*, 28(6):1035–1059, 2016.
- [63] A. Sharma, S. Jain, and A. Patel. Large scale ontology matching system (Ismatch). *Recent Advances in Computer Science and Communications*, 16:1–11, 2023.
- [64] G. Sousa, R. Lima, R. Vieira, and C. Trojahn. Using BERT models to automatically classify domain concepts into DOLCE top-level concepts: A study of the OAEI ontologies. In *Proceedings of the Joint Ontology Workshops 2023 Episode IX: The Quebec Summer of Ontology co-located with the 13th International Conference on Formal Ontology in Information Systems (FOIS 2023)*, Sherbrooke, Quebec, Canada, July 19-20, 2023, volume 3637 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2023.
- [65] M. Tounsi Dhouib, C. Faron Zucker, and A. G. Tettamanzi. An ontology alignment approach combining word embedding and the radius measure. In *Semantic Systems. The Power of AI and Knowledge Graphs: 15th International Conference, SEMANTiCS 2019, Karlsruhe, Germany, September 9–12, 2019, Proceedings 15*, pages 191–197. Springer International Publishing, 2019.
- [66] M. Uschold. *OWL Limitations*, pages 163–184. Springer International Publishing, Cham, 2018.
- [67] J. Vela and J. Gracia. Cross-lingual ontology matching with CIDER-LM: results for OAEI 2022. In *Proceedings of the 17th International Workshop on Ontology Matching (OM 2022) co-located with the 21th International Semantic Web Conference (ISWC 2022)*, Hangzhou, China, held as a virtual conference, October 23, 2022, volume 3324 of *CEUR Workshop Proceedings*, pages 158–165. CEUR-WS.org, 2022.
- [68] Z. Wang and I. F. Cruz. Agreementmakerdeep results for oaei 2021. In *OM@ ISWC*, pages 124–130, 2021.
- [69] V. E. Yamamoto and J. C. dos Reis. Updating ontology alignments in life sciences based on new concepts and their context. In *SeWeBMeDa@ ISWC*, pages 16–30, 2019.
- [70] L. Yang, K. Cormican, and M. Yu. Ontology-based systems engineering: A state-of-the-art review. *Computers in Industry*, 111:148–171, 2019.

- [71] A. Zekri, Z. Brahmia, F. Grandi, and R. Bouaziz. τ owl: A framework for managing temporal semantic web documents. In *Proceedings of the 8th International Conference on Advances in Semantic Processing (SEMAPRO 2014), Rome, Italy*, pages 33–41, 2014.
- [72] A. Zekri, Z. Brahmia, F. Grandi, and R. Bouaziz. Temporal schema versioning in τ owl: a systematic approach for the management of time-varying knowledge. *Journal of Decision Systems*, 26(2):113–137, 2017.
- [73] T. Zhang, D. Xu, and J. Chen. Application-oriented purely semantic precision and recall for ontology mapping evaluation. *Knowledge-Based Systems*, 21(8):794–799, 2008.
- [74] H. Zhao, H. Chen, F. Yang, N. Liu, H. Deng, H. Cai, S. Wang, D. Yin, and M. Du. Explainability for large language models: A survey. *arXiv preprint arXiv:2309.01029*, 2023.
- [75] L. Zhou and P. Hitzler. Aroa results for oaei 2020. In *OM@ ISWC*, pages 161–167, 2020.

4.2 Inne osiągnięcia

4.2.1 Zastosowanie głębokiego uczenia w uprawie winorośli

1. Franczyk B., Hernes M., Kozierekiewicz A., Kozina A., Pietranik M., Roemer I., Schieck M.: Deep learning for grape variety recognition. Knowledge-Based and Intelligent Information & Engineering Systems: Proceedings of the 24th International Conference KES 2020, s. 1211-1220. Lokalizacja elektroniczna: <http://dx.doi.org/10.1016/j.procs.2020.09.117>. **Punktacja MNiSW: 70, Klasyfikacja CORE: B**
2. Schieck M., Krajsic P., Loos F., Hussein A., Franczyk B., Kozierekiewicz A., Pietranik M.: Comparison of deep learning methods for grapevine growth stage recognition. Computers and Electronics in Agriculture. 2023, vol. 211, s. 1-9. Lokalizacja elektroniczna: <https://doi.org/10.1016/j.compag.2023.107944>. **Punktacja MNiSW: 100; Impact Factor 2023: 7.7**

Osiągnięcie naukowe, zrealizowane w ramach dwóch stażów badawczych na Uniwersytecie w Lipsku w ramach projektu Express⁷, obejmuje opracowanie zaawansowanych metod wykorzystujących uczenie głębokie oraz techniki rozpoznawania obrazu w celu wspierania cyfryzacji sektora winiarskiego i nowoczesnego zarządzania uprawami winorośli.

W pierwszej fazie projektu stworzono metodę ExtResNet, zaprojektowaną do identyfikacji odmian winogron na podstawie zdjęć liści i krzewów winorośli. Metoda ta opiera się na hybrydowej architekturze, która łączy głęboką sieć neuronową ResNet z wielowarstwowym perceptronem. Eksperymenty wykazały wyjątkową skuteczność rozwiązania, osiągając dokładność klasyfikacji na poziomie 99%. Wynik ten znacząco przewyższa istniejące w literaturze metody, co jednoznacznie potwierdza skuteczność zaproponowanego podejścia w precyzyjnej identyfikacji odmian winorośli z wykorzystaniem cyfrowych technologii.

W drugiej fazie projektu opracowano metodę klasyfikacji fenologicznych etapów wzrostu winorośli. Przeprowadzono szczegółowe badania porównawcze różnych architektur sieci neuronowych, takich jak ResNet, DenseNet i InceptionV3, uzyskując dokładność klasyfikacji na poziomie 88,1%. Wyniki te potwierdzają, że zaproponowane podejście umożliwia precyzyjne monitorowanie rozwoju winorośli, co jest kluczowe dla optymalizacji procesów zarządzania uprawami.

⁷<https://www.digitalisierung-landwirtschaft.de/ueber-express/>

Obie metody stanowią ważny wkład w rozwój Rolnictwa 4.0, dostarczając rolnikom zaawansowanych narzędzi wspomagających automatyzację i podejmowanie decyzji. Dzięki temu przyczyniają się do zrównoważonej, ekologicznej i efektywnej produkcji żywności, wspierając innowacyjność oraz zrównoważony rozwój sektora rolniczego, a także promując transformację cyfrową w rolnictwie.

4.2.2 Ocena ekspresywności danych w federacyjnych hurtowniach danych

1. Rafał J. Kern, Adrianna Kozierekiewicz, Marcin M. Pietranik.: The data richness estimation framework for federated data warehouse integration. *Information Sciences*. 2020, vol. 513, s. 397-411. Lokalizacja elektroniczna: <http://dx.doi.org/10.1016/j.ins.2019.10.046> **Punktacja MNiSW: 200, Impact Factor (2020): 6.795, Impact Factor (2024): 8.1**

Osiągnięcie naukowe przedstawione w artykule dotyczy opracowania nowatorskich metod i miar oceny ekspresywności danych w federacyjnych hurtowniach danych. Autorzy przedstawili formalne podstawy funkcjonowania hurtowni danych i ich federacji, a następnie zaproponowali zaawansowane algorytmy umożliwiające szczegółową analizę ekspresywności tych struktur. Kluczowym rezultatem badań było sformułowanie miar ekspresywności oraz analityczne udowodnienie ich właściwości, co stanowi istotny wkład w rozwój metod oceny jakości danych w złożonych systemach federacyjnych.

Eksperymenty miały na celu ocenę wpływu dołączania kolejnych hurtowni danych na zdolność federacji do poprawnego odpowiadania na zapytania. Analiza statystyczna, oparta na modelu wielorakiej regresji liniowej, wykazała silną zależność między opracowanymi miarami ekspresywności a efektywnością integracji danych. Wyniki te wskazują, że opracowane miary mogą być skutecznie wykorzystane do precyzyjnego oszacowania efektywności procesu integracji danych w oparciu o wartości tych miar.

Dodatkowo przeprowadzono porównanie zaproponowanych metod z podejściem ACCU, przy użyciu współczynnika korelacji wewnątrzklasowej. Wyniki potwierdziły wysoką użyteczność zaproponowanych narzędzi oraz ich potencjał w poprawie jakości integracji danych w systemach federacyjnych.

5. Informacja o wykazywaniu się istotną aktywnością naukową albo artystyczną realizowaną w więcej niż jednej uczelni, instytucji naukowej lub instytucji kultury, w szczególności zagranicznej.

Prowadzone przeze mnie prace badawcze powstały w wyniku aktywności naukowej realizowanej na kilku uczelniach zarówno polskich, jak i zagranicznych. Wśród aktywności można wyróżnić:

Pobyt stażowy w Leipzig University, 15.08 - 16.09.2022. W trakcie pobytu na Uniwersytecie w Lipsku prowadzone były prace badawcze na temat zastosowania warstwowej architektury głębokich sieci uczenia do rozpoznawania obrazów. Pobyt stażowy został zakończony publikacją:

- Franczyk B., Kozierekiewicz A., Pietranik M., Roemer I., Schieck M., Krajsic P., Comparison of deep learning approaches to recognition of grape growth development stage. **IF: 7.7, Punktacja MNiSW z 2019-2022: 100 pkt**

Pobyt stażowy w Leipzig University, 17.02 - 1.03.2020. W trakcie pobytu na Uniwersytecie w Lipsku prowadzone były prace badawcze na temat zastosowania głębokich sieci uczenia do rozpoznawania obrazów. Pobyt stażowy został zakończony publikacją:

- Franczyk B., Hernes M., Kozierekiewicz A., Kozina A., Pietranik M., Roemer I., Schieck M., 2020, Deep learning for grape variety recognition, *Procedia Computer Science*. 2020, vol. 176, s. 1211-1220, **Punktacja MNiSW z 2019-2022: 70 pkt, ranking CORE: B**

Pobyt naukowy w Vietnam National University, 16.03 - 31.03.2018. W trakcie pobytu nawiązana została współpraca z pracownikami naukowymi z uczelni. Zaowocowała ona szeregiem publikacji:

- Bogumiła Hnatkowska, Adrianna Kozierekiewicz, Marcin M. Pietranik, Hai Bang. Truong*: Assessing ontology alignments on the level of instances / Bogumiła Hnatkowska [i in]. W: *Computational Collective Intelligence : 13th International Conference, ICCCI 2021, Rhodes, Greece, September 29 - October 1, 2021 : proceedings* / eds. Ngoc Thanh Nguyen [i in.]. Cham : Springer, cop. 2021., s. 42-52,

(Lecture Notes in Computer Science. Lecture Notes in Artificial Intelligence, ISSN 0302-9743; 12876)), **Punktacja MNiSW z 2019-2022: 70pkt, ranking CORE: B**

- Kozierekiewicz A., Pietranik M., Nguyen L.T.T, 2020, *Updating ontology alignment on the instance level based on ontology evolution*, W: Database and Expert Systems Applications: 31st International Conference, DEXA 2020, Bratislava, Czech Republic, September 14-17, 2020: proceedings. Pt 2 / Eds. Sven Hartmann [i in.]. Cham: Springer, cop. 2020. s. 301-311, (Lecture Notes in Computer Science, ISSN 0302-9743; vol. 12392), **Punktacja MNiSW z 2019-2022: 70pkt, ranking CORE: B**

Współpraca z Uniwersytetem Ekonomicznym we Wrocławiu. Nawiązana została współpraca z pracownikami naukowymi uczelni, w wyniku której powstał szereg publikacji.

- Karolina Mateńczuk*, Agata Kozina*, Aleksandra Markowska*, Kateryna Czerniachowska*, Klaudia Kaczmarczyk*, Paweł Golec*, Marcin Hernes*, Krzysztof Lutosławski*, Adrianna Kozierekiewicz, Marcin M. Pietranik, Artur Rot*, Mykola Dyvak*: *Financial time series forecasting: comparison of traditional and spiking neural networks* / Karolina Mateńczuk [i in.]. W: Knowledge-Based and Intelligent Information & Engineering Systems: Proceedings of the 25th International Conference KES 2021 / KES International. Amsterdam : Elsevier, cop. 2021. s. 5023-5029, (Procedia Computer Science, ISSN 1877-0509; vol. 192), **Punktacja MNiSW z 2019-2022: 70pkt, ranking CORE: B**
- Kateryna Czerniachowska*, Krzysztof Lutosławski*, Agata Kozina*, Karolina Mateńczuk*, Aleksandra Markowska*, Adrianna Kozierekiewicz, Marcin M. Pietranik, Ingolf Roemer*, Martin Schieck*: *Shelf space allocation problem with horizontal shelf division* / Kateryna Czerniachowska [i in.]. W: Knowledge-Based and Intelligent Information & Engineering Systems: Proceedings of the 25th International Conference KES 2021 / KES International. Amsterdam : Elsevier, cop. 2021. s. 1550-1559, (Procedia Computer Science, ISSN 1877-0509; vol. 192), **Punktacja MNiSW z 2019-2022: 70pkt, ranking CORE: B**
- Hernes M., Kozierekiewicz A., Pietranik M., Sobieska-Karpińska J., 2018, *A new distance function for consensus determination in decision support systems*. Computational collective intelligence : 10th International Conference, ICCCI 2018, Bristol, UK, September 5-7, 2018: proceedings. Pt. 1 / Ngoc Thanh Nguyen [i in.] (Eds.). Cham : Springer, cop. 2018. s.155-165. **Punktacja MNiSW z 2019-2022:**

20pkt, ranking CORE: C, ranking CORE (rok 2021): B

- Hernes M., Chojnacka-Komorowska A., Kozierekiewicz A., Pietranik M., 2018, *Agents' knowledge conflicts' resolving in cognitive integrated management information system - case of budgeting module.*, Computational collective intelligence: 10th International Conference, ICCCI 2018, Bristol, UK, September 5-7, 2018: proceedings. Pt. 1 / Ngoc Thanh Nguyen [i in.] (Eds.). Cham: Springer, cop. 2018. s. 297-308. **Punktacja MNiSW z 2019-2022: 20pkt, ranking CORE: C, ranking CORE (rok 2021): B**

- Hernes M., Kozierekiewicz A., Pietranik M., 2019, *The agents' selection methods for a consensus-based investment strategy in a multi-agent financial decisions support system. W: New Trends in Databases and Information Systems, ADBIS 2019 Short Papers, Workshops: BBIGAP, QAUCA, SemBDM, SIMPDA, M2P, MADEISD and Doctoral Consortium: Bled, Slovenia, September 8-11, 2019: proceedings / eds. Tatjana Welzer [i in.]. Cham: Springer, cop. 2019. s. 60-67. Punktacja MNiSW z 2019-2022: 70pkt, ranking CORE: B*

6. Informacja o osiągnięciach dydaktycznych, organizacyjnych oraz popularyzujących naukę lub sztukę.

6.1 Podręczniki i inne pomoce dydaktyczne

- Opracowanie materiałów dydaktycznych w języku angielskim dla kursu "*Projektowanie baz danych*" w ramach projektu „ZPR PWr – Zintegrowany Program Rozwoju Politechniki Wrocławskiej”, nr WND-POWR.03.05.00-00-Z301/17

6.2 Opieka nad pracami inżynierskimi i magisterskimi

Pod moim nadzorem powstały oraz zostały obronione liczne prace inżynierskie i magisterskie:

- Prace magisterskie: 12
- Prace inżynierskie: 21
- Recenzje: 44
- Prace naukowe powstałe we współpracy ze studentami i doktorantami: 14

7. Oprócz kwestii wymienionych w pkt. 1-6, wnioskodawca może podać inne informacje, ważne z jego punktu widzenia, dotyczące jego kariery zawodowej.

- Opiekun koła naukowego WMS_DEV.



.....
(podpis wnioskodawcy)