

POLITECHNIKA OPOLSKA

WYDZIAŁ ELEKTROTECHNIKI, AUTOMATYKI I INFORMATYKI

INSTYTUT: Automatyki i Informatyki



Rozprawa doktorska pt.

**Inteligentne wyszukiwanie wiedzy diagnostycznej
wykorzystujące rozmytą logikę opisową
na przykładzie wybranej klasy obiektów diagnozowanych**

Autor: mgr inż. Anna Bryniarska

Promotor: prof. dr hab. inż. Anna Walaszek-Babiszewska

OPOLE 2014

*Składam serdeczne podziękowania Pani
prof. dr hab. inż. Annie Walaszek-Babiszewskiej
za życzliwość, zrozumienie, cenne uwagi
i okazane wsparcie przy pisaniu niniejszej pracy.*

Spis treści

1. OPIS PROBLEMU BADAWCZEGO	5
1.1. Motywacja dysertacji	5
1.2. Umieszczenie tematu dysertacji w dyscyplinie automatyka i robotyka	6
1.3. Cel i teza pracy	7
1.4. Zakres pracy	9
2. PODSTAWY TEORETYCZNE WYSZUKIWANIA WIEDZY Z ZASTOSOWANIEM SIECI SEMANTYCZNYCH I ROZMYTEJ LOGIKI OPISOWEJ	11
2.1. Pojęcie systemu diagnostycznego w diagnostyce technicznej	11
2.1.1. Reprezentacja wiedzy w systemach ekspertowych	12
2.1.2. Diagnoza techniczna jako pomiar w systemie diagnostycznym	13
2.1.3. Wnioskowanie diagnostyczne	19
2.1.4. Wnioskowanie diagnostyczne z zastosowaniem logiki rozmytej	20
2.2. Modele wiedzy diagnostycznej w systemach wyszukiwania informacji	21
2.2.1. Formalnojęzykowa struktura świata rzeczywistego a sieci semantyczne	22
2.2.2. Agenci w systemach komunikowania się	29
2.2.3. Modele wiedzy diagnostycznej w systemach danych diagnostycznych	33
2.2.4. Logika atrybutowa	37
2.2.5. Wieloargumentowe atrybuty i tablice decyzyjne dla tych atrybutów	39
2.2.6. Sieć semantyczna wyznaczona przez język atrybutów wieloargumentowych ...	40
2.2.7. Reprezentacja wiedzy w sieci semantycznej Web	43
2.2.8. Ontologia i tezaurus	48
2.2.9. Modele wiedzy nieadekwatnej i niedoskonałej w systemach danych	52
2.2.10. Modele wiedzy diagnostycznej w informatycznych systemach danych	56
2.3. Wybrane zagadnienia z teorii zbiorów rozmytych	63
2.3.1. Teoria zbiorów rozmytych	63
2.3.2. Podstawowe definicje i operacje na zbiorach rozmytych	65
2.3.3. Normy trójkątne w logice rozmytej	66
2.3.4. Wnioskowanie przybliżone w systemach wyszukiwania wiedzy	69
2.4. Postulaty znaczeniowe logiki wyszukiwania informacji	72
2.4.1. Dokładność wyniku rozmytego wyszukiwania informacji	73
2.4.2. Język logiki IRL	75
2.4.3. Propozycja postulatów rozmytości logiki IRL	77
2.5. Logika wyszukiwania informacji w sieciach semantycznych Web	80
2.5.1. Operacja rozmywania w sieci semantycznej	80
2.5.2. Semantyka logiki IRL	82
2.5.3. Proponowana logika wyszukiwania informacji IRL	83
2.5.4. Proponowana interpretacja wyostrzania w logice IRL	84

2.6. Metoda inteligentnego wyszukiwania wiedzy diagnostycznej z wykorzystaniem logiki IRL	85
3. KONCEPCJA INTELIGENTNEGO SYSTEMU WYSZUKIWANIA WIEDZY DIAGNOSTYCZNEJ	88
3.1. Koncepcja implementacji	88
3.1.1. Wyniki badań teoretycznych umożliwiające opracowanie koncepcji.....	88
3.1.2. Procesy decyzyjne ekspertów prowadzące do ustalenia zakresu ufności	91
3.1.3. Opis koncepcji systemu wyszukiwania wiedzy w sieciach semantycznych	92
3.2. Mapowanie reprezentacji wiedzy w języku OWL DL.....	98
3.3. Wybrane przykłady wyszukiwania wiedzy w sieciach semantycznych	99
3.3.1. Klasyfikator konstrukcyjno-technologiczny dla systemu produkcji	100
3.3.2. Implementacja w języku OWL DL pomiaru ciśnienia sprężania w silniku samochodowym.....	103
3.4. Przykład wyszukiwania wiedzy diagnostycznej	107
3.4.1. Opis diagnozowanego obiektu	108
3.4.2. Schemat realizacji projektu	109
3.4.3. Tabela diagnostyczna reprezentująca rozmyte dane diagnostyczne	110
3.4.4. Diagnostyczna tablica decyzyjna	117
3.4.5. Pozyskiwanie eksperckiej wiedzy diagnostycznej	118
3.4.6. Klasyfikator konstrukcyjno-technologiczny kotła gazowego	125
3.4.7. Powiązanie klasyfikatora ze stanami określającymi usterki i awarie.....	127
3.4.8. Działanie aplikacji	130
3.4.9. Oprogramowanie wykorzystane do aplikacji	136
4. PODSUMOWANIE I WNIOSKI KOŃCOWE	139
LITERATURA	142
SPIS RYSUNKÓW	151
SPIS TABEL	153
DODATEK A.....	154
DODATEK B	171

1. OPIS PROBLEMU BADAWCZEGO

1.1. Motywacja dysertacji

Urządzenia techniczne cechuje coraz większa złożoność. Rośnie również złożoność procesów technologicznych i produkcyjnych, prowadzących do wytworzenia tych urządzeń. Powoduje to, że przy rozwiązywaniu pewnych zadań technicznych i inżynierskich dostępna wiedza jest niepewna, niejasna lub niepełna. Dotyczy to również **diagnostyki technicznej** oraz wiedzy o symptomach stanów, w których pozostają urządzenia techniczne. Do takich stanów zaliczamy wady: konstrukcyjne, technologiczne, użytkowe, uszkodzenia, zewnętrzne blokady czy niedostosowanie do środowiska użytkowania. Zdarza się często, że dysponując niepełną wiedzą o symptomach stanów urządzeń technicznych, próbujemy ustalić ich stan: wcześniejszy, obecny lub przyszły [kor02, s. 6].

Wiedza, którą posiadamy z danej dziedziny, składa się nie tylko z poszczególnych pojęć, ale również zawiera informację o związkach, jakie zachodzą pomiędzy tymi pojęciami. W niniejszej pracy będzie to wiedza z zakresu diagnostyki technicznej. Możemy ją reprezentować za pomocą sieci informacyjnej, łączącej stany wyszukiwania wiedzy o obiektach, zwanej **siecią semantyczną** [ber01], [bak09], [sow91]. W takiej sieci stany związków pomiędzy obiektami opisane są za pomocą danych o tych obiektach oraz związkach pomiędzy tymi obiektami (opisach obiektów i ich związków).

Dane o obiektach, dotyczące na przykład procesu diagnostycznego, mogą być przetwarzane komputerowo. Jeżeli oprócz danych mamy dostępne również informacje o powiązaniach pomiędzy tymi obiektami, to możemy użyć sieci semantycznych do opisu takiej wiedzy. Współcześnie wiedza techniczna coraz częściej pozyskiwana jest z globalnych sieci komputerowych. Do takich sieci należą między innymi sieć semantyczna Web, określona przez projekt W3C [w3c14].

Również jeżeli dostępna wiedza, którą posiadamy lub wyszukujemy, jest niepewna, niejasna lub niepełna, możemy używać sieci semantycznych do reprezentacji tej wiedzy. W tym celu wykorzystuje się język **rozmytej logiki opisowej** (ang. *Fuzzy Description Logic – fuzzyDL*) [baa03], [bob08], [str98] opartej na teorii logiki rozmytej. Wtedy w strukturze sieci do poszczególnych pojęć i związków między nimi możemy określać ich stopień rozmycia. Natomiast podczas wyszukiwania informacji uwzględniane są zarówno stopnie rozmycia wiedzy, jak i związki występujące pomiędzy pojęciami.

Motywacją do rozpoczęcia badań w tej dziedzinie stało się poszukiwanie odpowiedzi na pytanie: *jak precyzyjnie określić proces wyszukiwania wiedzy (informacji) reprezentowanej w sposób rozmyty/lingwistyczny w sieciach semantycznych, w wyniku którego otrzyma się wiedzę dokładną?* Tak dokładnie, precyzyjnie określone dane mogą zostać użyte w procesach technologicznych i diagnostycznych [gal04]. Natomiast od metod przetwarzania tych danych zależy sposób projektowania i wytwarzania obiektów technicznych (w tym automatów czy robotów) oraz diagnozowania ich stanów. Należy zwrócić uwagę na to, że bez teoretycznego

rozwiązania problemu diagnozowania obiektów technicznych, wykorzystującego wiedzę niepewną, nie będzie można opracować ogólnych algorytmów automatyzacji procesów wytwarzania tych obiektów czy algorytmów sterowania nimi w środowisku rozmytym. Ten aspekt badań jest motywacją do rozpoczęcia rozważań nad problemem wyszukiwania wiedzy z zakresu diagnostyki technicznej w sieciach semantycznych przy wykorzystaniu rozmytej logiki opisowej.

W niniejszej dysertacji została przedstawiona koncepcja inteligentnego systemu ekspertowego, służącego do wyszukiwania wiedzy niepewnej z dziedziny diagnostyki technicznej. System ten wykorzystuje sieci semantyczne do opisu wiedzy. Natomiast rozmyta logika opisowa, oparta na teorii logiki rozmytej, została wykorzystana w celu opisu bazy wiedzy. Dzięki takiemu opisowi można w trakcie procesów wnioskowania i wyostrzenia otrzymać dokładną wiedzę, która jest wyszukiwana.

1.2. Umiejscowienie tematu dysertacji w dyscyplinie automatyka i robotyka

Inteligentny system wyszukiwania, którego koncepcja jest przedstawiona w niniejszej pracy, można zaliczyć do **systemów ekspertowych**. Systemy ekspertowe służą do rozwiązania specjalistycznych problemów, co przyczynia się do polepszenia jakości produkowanych wyrobów [mul96]. System ekspertowy jest pewnym programem komputerowym, który wykorzystuje wiedzę ekspercką z danej dziedziny, zapisaną za pomocą bazy wiedzy, w celu wyciągnięcia pewnych wniosków i podejmowania decyzji [mul96]. Przedstawiona koncepcja wyszukiwania wiedzy opiera się na pozyskiwaniu i przetwarzaniu wiedzy zapisanej w formie lingwistycznej, dlatego można ją również zaliczyć do dziedziny zwanej **inżynierią wiedzy**.

Systemy ekspertowe zaliczamy do narzędzi z dziedziny **sztucznej inteligencji** (ang. *Artificial Intelligence – AI*), która silnie wspiera badania w zakresie urządzeń technicznych [paw96]. W metodach sztucznej inteligencji następuje przejście od przetwarzania danych do przetwarzania wiedzy. Systemy te określa się jako systemy oparte na wiedzy zapisanej w postaci bazy wiedzy [nie00]. Metody algorytmiczne charakterystyczne dla przetwarzania proceduralnego w sztucznej inteligencji zostają zastąpione przeszukiwaniem inteligentnym.

Systemy ekspertowe, jako pewne systemy komputerowe, wspomagają automatyzację procesów i funkcji, w których wykorzystywana jest praca umysłowa człowieka. Dziedziny, w których te systemy mają szerokie zastosowanie, to przetwarzanie i statystyczna obróbka danych; prace badawcze, a w szczególności eksperymentalne; projektowanie; sterowanie procesami wytwórczymi i działaniem systemów technicznych; zarządzanie i wiele innych [bub90]. Z tego powodu systemy ekspertowe znajdują się na pograniczu dwóch dziedzin nauki: *informatyki i automatyki*.

W automatyce zastosowanie systemów ekspertowych jest coraz większe i istotnie wpływa na rozwój tej dziedziny. Dzięki wykorzystaniu zasobów komputerowych można przyspieszyć wiele procesów automatyzacji, zwłaszcza gdy do przetworzenia jest duża ilość danych [bub90], [roj00]. W niniejszej pracy koncepcja systemu ekspertowego została przygotowana tak, by przetwarzać bardzo dużo danych lingwistycznych z dziedziny diagnostyki technicznej. Ze

względu na szerokie zastosowanie systemu w procesie automatyzacji wyszukiwania wiedzy – praca jest umiejscowiona w dyscyplinie **automatyka i robotyka**.

Przedmiot badań jest zgodny ze współczesnymi możliwościami technologii informacyjnej, które silnie wspomagają metody diagnostyczne. Technologia ta zarówno w życiu codziennym, jak i w procesach technologicznych coraz częściej wykorzystywana jest do komputerowego przetwarzania danych o obiektach diagnostycznych. Dane te mogą być reprezentowane przez sieci semantyczne.

W pracy rozważane będą sieci semantyczne, w których wiedza będzie wykorzystywana w procesie diagnostycznym dla wybranych obiektów technicznych (inżynierskich). Ze względu na wykorzystanie wiedzy do diagnozowania jej reprezentacja dotyczy *obiektów diagnostycznych*. Koncepcja systemu została wykorzystana do stworzenia aplikacji umożliwiającej wyszukiwanie wiedzy diagnostycznej z tezauryza zapisanego za pomocą sieci semantycznej na przykładzie wiedzy diagnostycznej dotyczącej kotłów gazowych małej mocy.

Metody pozyskiwania i wyszukiwania wiedzy z sieci semantycznych [ceg06], [sim08] obecnie mają duże znaczenie dla nauki, w tym również dla systemów diagnostycznych. W trakcie diagnozowania stanu urządzenia wyszukiwanie wiedzy na ten temat, w szczególności gdy jest ona niepełna lub niedokładna, jest bardzo istotnym aspektem. Fakt ten miał decydujący wpływ na wybór zakresu badań w niniejszej dysertacji, tzn. na wybór zagadnień wyszukiwania wiedzy reprezentowanej w sieciach semantycznych zgodnie z zasadami określonymi w tzw. **rozmytej logice opisowej** [bob07], [bob08], [bob09], [pan08], [sim10], [str98] dla systemów diagnostyki technicznej.

1.3. Cel i teza pracy

Przedmiotem badań niniejszej pracy jest wyszukiwanie wiedzy i wnioskowanie diagnostyczne w systemach diagnostyki technicznej wykorzystujących niepełną i niejasną wiedzę o obiektach diagnozowanych.

W świetle przeprowadzonej analizy aktualnego stanu wiedzy i badań w ustalonym wyżej zakresie jako **cel główny** pracy postawiono:

Utworzenie koncepcji systemu wyszukiwania wiedzy diagnostycznej w sieciach semantycznych jako systemu dokonującego klasyfikacji obiektów diagnozowanych na podstawie metody wyostrzania danych w języku rozmytej logiki opisowej.

W praktyce wyszukiwana wiedza w sieciach semantycznych, w szczególności z zakresu diagnostyki technicznej, jest często wiedzą niepewną, niejasną lub niepełną. Systemy wyszukiwania tej wiedzy są **systemami rozmytymi** o strukturze składającej się z:

- rozmytej bazy wiedzy,
- rozmytej reprezentacji wiedzy w sieci semantycznej,

- automatu wyszukującego wiedzę w oparciu o rozmytą bazę wiedzy i jej reprezentację w sieci semantycznej,
- procesu wyostrzania wyników wyszukiwania wiedzy.

Myśl ta zainspirowana jest koncepcją systemów rozmytych [bob08], [yag95], [łęś08].

Jak dotąd w literaturze poświęconej sieciom semantycznym precyzyjnie określono proces rozmywania w sieci semantycznej oraz rozmytą bazę wiedzy. Kompleksowe i precyzyjne opisanie rozmytych baz wiedzy, które częściowo zostanie wykorzystane, znajduje się w artykule [bob08], a aktualne tendencje badań z tej dziedziny prezentowane są w pracy [sim10]. Natomiast trudno odnaleźć w literaturze przedmiotu zagadnienia wyszukiwania wiedzy reprezentowanej w sposób rozmyty w sieciach semantycznych oraz wyostrzania tej reprezentacji dla wyróżnionej przez ekspertów klasy interpretacji rozmywania i przyjętych kryteriów ufności dla tej wiedzy. Rozważane jest jedynie tworzenie języków rozmytych zapytań [bob08], [pan08], [sim08], [sim10], tj. pytań o wiedzę, bez analiz metod i algorytmów wyszukiwania odpowiedzi na te pytania.

W niniejszej pracy zajmiemy się opisaniem aparatu pojęciowego potrzebnego do sformułowania problemu wyszukiwania wiedzy niepewnej w sieciach semantycznych oraz otrzymania dokładnej wiedzy w wyniku procesu wyostrzania. Do postawienia i rozwiązywania problemu zostały wykorzystane dostępne publikacje i prace naukowe, prezentowane w ostatnich latach na światowych konferencjach z zakresu sztucznej inteligencji i automatyki.

W pracy ponadto zostanie pokazane, jak wiedzę z dziedziny diagnostyki technicznej można reprezentować w sieciach semantycznych. Określone zostanie też, jak ustalać zakres ufności dla rozmytej interpretacji wiedzy. Sformułowane definicje mogą być implementowane w OWL (ang. *Web Ontology Language*) lub XML (ang. *Extensible Markup Language*), co również zostanie zaprezentowane. Umożliwi to określenie metody wyszukiwania wiedzy w Internecie (zbudowania wyszukiwarki) w zakresie diagnostyki technicznej, tak aby można było dokonać ostrej klasyfikacji obiektów opisywanych przez wiedzę niepełną lub niedokładną wyszukiwaną w sieciach semantycznych.

Prowadzone badania, w powiązaniu z analizą aktualnego stanu wiedzy i badań, pozwoliły na sformułowanie następującej **tezy rozprawy doktorskiej**:

Interpretacje rozmyte i wyostrzające wyrażen rozmytej logiki opisowej dają możliwość dokonania ostrej klasyfikacji obiektów diagnostycznych opisywanych na podstawie wiedzy wyszukiwanej w sieciach semantycznych.

Do uzasadnienia podanej tezy oraz realizacji celu głównego zostały określone cele szczegółowe, które podzielono na zadania teoretyczne i praktyczne. **Zadania teoretyczne**, odpowiadające także opisanej wcześniej problematyce, to:

- 1) Przedstawienie aktualnych badań związanych z tematyką dysertacji w zakresie systemów ekspertowych, diagnostyki technicznej, modelowania wiedzy oraz logiki rozmytej.
- 2) Formalny opis rozmytego modelu danych dla wyszukiwania wiedzy w sieciach semantycznych.
- 3) Formalny opis rozmytej logiki opisowej.
- 4) Metody wnioskowania i wyostrzania wiedzy niepełnej dla wyszukiwania w sieciach semantycznych.

Określono również **zadania praktyczne**, które składają się na koncepcję systemu wyszukiwania wiedzy diagnostycznej w sieciach semantycznych:

- 1) Opracowanie algorytmu tworzenia tezaury i ontologii dla inżynierskich obiektów diagnostycznych.
- 2) Opracowanie algorytmu wyszukiwania i wyostrzania wiedzy niepełnej w sieci semantycznej Web zgodnie z tezaurem i ontologią.
- 3) Projekt interfejsu umożliwiającego tworzenie wyszukiwarki internetowej.
- 4) Aplikacja tezaury dla wybranego obiektu diagnostycznego oraz aplikacja wyszukiwania wiedzy diagnostycznej w stworzonym tezaurysie.

1.4. Zakres pracy

Praca została podzielona na cztery główne rozdziały. W rozdziale 2. przedstawiono podstawy teoretyczne wyszukiwania i modelowania wiedzy diagnostycznej z zastosowaniem sieci semantycznych oraz rozmytej logiki opisowej. Ma on charakter teoretyczny i wprowadzający do dziedziny badawczej. Rozdział 3. jest praktyczny i zawiera opis autorskiej koncepcji systemu i jej aplikacji. Podsumowanie pracy oraz omówienie dalszych możliwych kierunków badań zostało przedstawione w rozdziale 4.

Rozdział 2. zawiera analizę podstaw teoretycznych do utworzenia koncepcji systemu diagnostycznego wykorzystującego sieci semantyczne oraz rozmytą reprezentację wiedzy. W podrozdziale 2.1. omówiono charakterystykę systemów ekspertowych, a w szczególności ekspertowych systemów diagnostycznych.

W rozdziale tym przedstawiono również podstawy teoretyczne systemów diagnostycznych, ze szczególnym uwzględnieniem wiedzy diagnostycznej oraz jej modeli w systemach wyszukiwania informacji (rozd. 2.2.). Rozdział ten zawiera również podstawy logiki rozmytej (rozd. 2.3.), sieci semantycznych i logiki wyszukiwania rozmytych informacji w sieci semantycznej.

W dalszej części w podrozdziale 2.4. opisano podstawy rozmytej logiki opisowej, operację rozmywania w sieciach semantycznych oraz semantykę języka rozmytej logiki opisowej. Następnie w podrozdziale 2.5. przedstawiono autorskie ujęcie operacji wyostrzania i wyszukiwania informacji w sieci semantycznej Web z uwzględnieniem niepewności i niedokładności wiedzy. Podrozdział 2.6. stanowi podsumowanie rozdziału 2. oraz prezentuje

autorską metodą wyszukiwania wiedzy niepewnej w sieciach semantycznych. Opisana wiedza teoretyczna pozwoliła na utworzenie koncepcji systemu wyszukiwania informacji.

Rozdział 3. zawiera autorską koncepcję systemu diagnostycznego jako propozycję wykorzystującą rozmytą logikę opisową do opisu wiedzy niepewnej i niedokładnej w sieciach semantycznych Web (rozdz. 3.1.). Koncepcja ta zawiera algorytmy tworzenia tezaury i ontologii dla obiektów diagnostycznych. Zaproponowano również metodę implementacji operacji rozmywania, wyostrażania i wyszukiwania danych rozmytych w sieciach semantycznych Web.

W podrozdziale 3.2. przybliżono również język programowania OWL służący do zapisu sieci semantycznych oraz jego powiązania z rozmytą logiką opisową przedstawioną w części teoretycznej (rozdz. 2.4. i 2.5.). Natomiast podrozdział 3.3. zawiera przykładowe zastosowania algorytmów zawartych w koncepcji systemu przy wykorzystaniu języka OWL.

W podrozdziale 3.4. została opisana aplikacja tezaury dla wybranego obiektu diagnostycznego oraz aplikacja wyszukiwania wiedzy diagnostycznej w stworzonym teaurusie. Wybrany obiektem diagnostycznym w niniejszej pracy jest kocioł gazowy małej mocy. Przedstawiona aplikacja pozwala na diagnostykę kotłów na podstawie wiedzy lingwistycznej zapisanej w teaurusie, przy uwzględnieniu niepewności występowania stanów diagnostycznych. Krótko scharakteryzowano także oprogramowanie, które zostało wykorzystane przy tworzeniu aplikacji.

Rozdział 4. zawiera podsumowanie pracy oraz wskazuje na dalsze kierunki badań. Ukazano oryginalne pomysły autorki prezentowane w publikacjach i w niniejszej pracy oraz to, jakie elementy wiedzy zostały przedstawione w dysertacji.

Wskazano dwa główne nurty dalszych badań. Pierwsza możliwość to rozszerzenie stworzonej aplikacji do diagnostyki kotłów gazowych o wiedzę dostępną z innych źródeł internetowych. W tym celu należałoby zaimplementować interfejs określający ontologię diagnozy oraz agenta wyszukującego automatycznie informacje w zasobach internetowych. Odpowiednie algorytmy zostały już przedstawione w koncepcji systemu (rozdz. 3.1.). Drugi nurt dalszych badań to rozszerzenie aplikacji na inne typy kotłów gazowych, takie jak przykładowo kotły kondensacyjne czy kotły o większej mocy. Można też w aplikacji uwzględnić kotły olejowe czy na paliwo stałe lub podobne pod względem budowy obiekty diagnostyczne, np. termy gazowe.

2. PODSTAWY TEORETYCZNE WYSZUKIWANIA WIEDZY Z ZASTOSOWANIEM SIECI SEMANTYCZNYCH I ROZMYTEJ LOGIKI OPISOWEJ

Niniejszy rozdział przybliża podstawy teoretyczne tematyki rozprawy. Został on opracowany na podstawie przeglądu literatury z takich dziedzin, jak systemy ekspertowe, diagnostyka techniczna, wyszukiwanie wiedzy, logika rozmyta, logika opisowa, sieci semantyczne oraz pokrewnych. Ponadto, zaproponowano postulaty rozmytości logiki wyszukiwania informacji, logikę wyszukiwania informacji IRL oraz interpretację wyostrzenia wiedzy w tej logice. Rozdział kończy opis autorskiej metody inteligentnego wyszukiwania wiedzy diagnostycznej z wykorzystaniem logiki IRL.

Systemy ekspertowe są przedmiotem wielu badań w dziedzinie automatyki i robotyki. Jednym z zastosowań systemów ekspertowych jest diagnostyka techniczna. Dane przetwarzane w procesie diagnostycznym są *nośnikiem informacji diagnostycznej* oraz *źródłem wiedzy diagnostycznej* [moc02, s. 67], [cho93]. Zwykle wiedza diagnostyczna reprezentowana jest za pomocą konkretnych wartości, uzyskanych na przykład podczas pomiaru wartości fizycznych badanego obiektu. Jednak nie zawsze mamy możliwość dokonania pomiaru lub wiedza, którą posiadamy, jest wyrażona za pomocą języka naturalnego. Ma to miejsce na przykład wtedy, gdy wiedzę uzyskujemy od eksperta lub specjalisty w danej dziedzinie. Niestety, taka wiedza jest obciążona pewną niedokładnością, niepewnością lub też niepełnością. Powoduje to, że system diagnostyczny wykorzystujący taką wiedzę jest systemem niedeterministycznym.

Wiedza jest reprezentowana za pomocą modeli wiedzy. Model wiedzy diagnostycznej może również uwzględniać niepewność i niedokładność, która powstaje wskutek przetwarzania opisanej wiedzy. W celu uwzględnienia niepewności danych można zastosować między innymi przybliżone reprezentacje wiedzy, modelowanie wiedzy w sieci przekonań lub modelowanie wiedzy w ujęciu rozmytym [cho02].

Jedną z metod reprezentowania wiedzy są sieci semantyczne. Zastosowanie sieci semantycznych pozwala na uwzględnienie powiązań pomiędzy różnymi pojęciami zapisanymi w tej sieci oraz uwzględnienie semantycznych znaczeń opisanych pojęć. W sieciach semantycznych może również znajdować się wiedza z zakresu diagnostyki technicznej, która będzie reprezentowana w sposób rozmyty. W tym celu wykorzystuje się język rozmytej logiki opisowej (*fuzzyDL*).

2.1. Pojęcie systemu diagnostycznego w diagnostyce technicznej

Zakres wyszukiwanej wiedzy, rozpatrywanej w niniejszej pracy, ograniczony jest do diagnostyki technicznej, dlatego pod pojęciem wiedzy diagnostycznej będzie rozumiana wiedza z zakresu diagnostyki technicznej. W rozdziale tym przedstawiono podstawowe informacje z zakresu reprezentacji wiedzy w systemach ekspertowych, diagnostyki technicznej oraz wnioskowania diagnostycznego, w tym rozmytego.

2.1.1. Reprezentacja wiedzy w systemach ekspertowych

Inżynieria wiedzy pozwala na określenie problemów za pomocą wiedzy i wnioskowania. Wtedy komputer jest traktowany jako ekspert, natomiast system komputerowy realizujący takie zadania jest **systemem ekspertowym** [bub90]. Systemy ekspertowe służą do automatyzacji procesów i zadań wykonywanych przez człowieka, starając się jak najlepiej odwzorować mechanizm działania ludzi, którzy są ekspertami w danej dziedzinie. Dlatego w tych systemach, w przeciwieństwie do innych, pojawia się etap rozumowania. Rozumowanie pozwala na wyciągnięcie wniosków i podjęcie decyzji, wspomagając pracę człowieka [bub90].

Według Jana Muławki [mul96] system ekspertowy jest „programem komputerowym, który wykonuje złożone zadania o dużych wymaganiach intelektualnych i robi to tak dobrze jak człowiek będący ekspertem w tej dziedzinie”. Należy jednak podkreślić, że system ekspertowy nie musi całkowicie zastępować eksperta, ale może służyć jako narzędzie pomocne przy podjęciu decyzji. Również systemy ekspertowe bywają omylne, tak jak eksperci, na których są wzorowane [bub90].

Systemy ekspertowe można podzielić, między innymi, na:

- doradcze,
- podejmujące decyzje bez kontroli człowieka,
- krytykujące.

Niektóre systemy doradcze przedstawiają pewne rozwiązania, które człowiek może zaakceptować albo odrzucić i zażądać innego rozwiązania problemu, inne systemy podejmują decyzje bez kontroli człowieka. Takie podejście jest szczególnie istotne przy sterowaniu obiektami, gdzie należy szybko podjąć decyzję, a dostęp człowieka jest utrudniony. Systemy krytykujące, natomiast, oceniają i analizują decyzję podjętą przez człowieka w odniesieniu do danego problemu [mul96].

Struktura systemu ekspertowego składa się z następujących elementów [paw96]:

- baza wiedzy,
- mechanizmy wnioskowania (metoda rozwiązania problemu),
- komponenty wyjaśniające,
- komponenty nabywania wiedzy,
- moduły kontaktu z użytkownikiem (interfejs).

Jednak baza wiedzy i mechanizm wnioskowania determinują to, czy dany system jest systemem ekspertowym, a reszta elementów ma znaczenie wspomagające [paw96].

Metody wnioskowania można podzielić na słabe i mocne, w zależności od zastosowanego sposobu rozwiązywania problemów. Słabe metody mają bardziej otwartą strukturę, przez co reprezentowana przez nie wiedza może być bardziej różnorodna. Natomiast mocne metody są mniej elastyczne, bardziej specjalistyczne w danej dziedzinie, co jest pomocne przy pozyskiwaniu wiedzy [paw96].

Wiedzą, w systemie ekspertowym, nazywamy dowolne zbiory informacji o obiektach, czyli klasy typów danych, które są jednakowo używane do opisu tych obiektów. Informacje są tym,

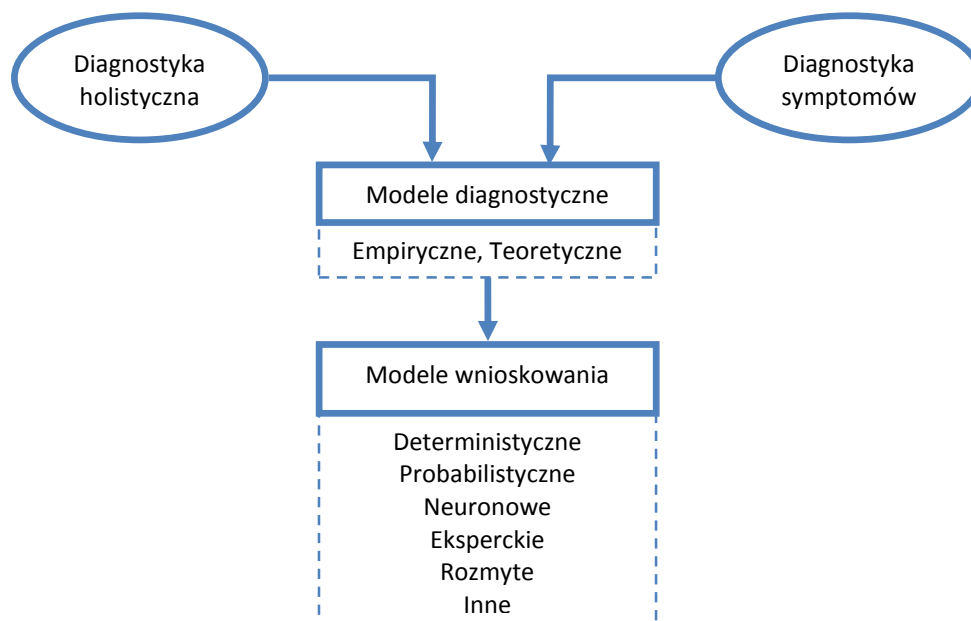
co wiemy o identyfikowanych obiektach, lub tym, co przenoszone jest przez sygnały w procesach sterowania. W tym sensie opisywane są przez ilościową teorię informacji (np. teorię Shannona) [fla11]. Informacje te identyfikują pewne obiekty lub sterują nimi oraz są reprezentowane przez pewne dane lingwistyczne (zwane też opisami). O tych danych mówimy, że reprezentują wiedzę. Zbiory rozumiane są jako pewne systemy identyfikacji obiektów – grupowania obiektów zgodnie z pewną formalnie ustaloną metodą: np. zbiory abstrakcyjne, kolektywne, zdarzenia probabilistyczne, multizbiory (ang. *multi sets*) w sensie Blizarda, zbiory przybliżone (ang. *rough sets*) w sensie Pawlaka, zbiory rozmyte (ang. *fuzzy sets*) w sensie Zadeha. Dla wymienionych typów pojmowania zbiorów wiedzą są bezpośrednio wskazywane obiekty jako „te obiekty”. Pojęcia, jako zbiory obiektów o danej nazwie, również określają pewną wiedzę. Rozważane są też zbiory wyuczonych wiadomości lub zbiory informacji jako listy wskazań urządzeń pomiarowych i listy danych czy opisów sygnałów oraz informacji przetwarzanych przez komputer [fla11], [nie00].

Wiedzę, która jest precyzyjnie (najczęściej jednoznacznie) określona poprzez zbiory abstrakcyjne za pomocą formuł logicznych lub matematycznych, np. funkcji liczbowych, równań różniczkowych, wektorowych, macierzowych lub tensorowych, nazywamy **wiedzą ostrą, pewną** lub **dokładną** [nie00]. Gdy wiedzy nie można dokładnie określić jako zbiór jednakowo używanych typów danych, w ramach jakiejś lub każdej metody identyfikacji obiektów, to nazywamy ją **wiedzą niepełną, niejasną** lub **niepewną** (ang. *incomplete, vague, uncertain*). Wiedzę niepewną stosuje się w opisie probabilistycznym, wskaźnikowym lub rozmytym [bub01]. Natomiast proces wyostrzenia reprezentacji tej wiedzy prowadzi od wiedzy niepewnej do wiedzy dokładnej. W niniejszej pracy system ekspertowy będzie służył do wyszukiwania i precyzowania wiedzy niepewnej z zakresu diagnostyki technicznej.

2.1.2. Diagnostyka techniczna jako pomiar w systemie diagnostycznym

Dziedzina nauki zajmująca się środkami i sposobami rozpoznawania zdarzeń, a więc stanów i ich zmian na podstawie ich charakterystycznych objawów – symptomów, nazywana jest **diagnostyką** (gr. *diagnosis* – rozpoznanie, określenie). Dziedzina diagnostyki całościowo ujmująca stany obiektów, od ich powstania do momentu ich przeminięcia, nazywana jest diagnostyką holistyczną, natomiast dziedzina pozwalająca ustalić symptomy stanów nazywana jest diagnostyką symptomową (rysunek 2.1.).

W niniejszej pracy, mówiąc o diagnostyce, będziemy mieli na myśli **diagnostykę techniczną** (gr. *diagnostike techne* – sztuka rozróżniania, sztuka stawiania diagnozy). Diagnostyka techniczna oznacza naukę o rozpoznaniu stanów obiektów technicznych (bez ich demontażu) na podstawie symptomów tych stanów [kor02]. Zajmuje się oceną stanu technicznego maszyn poprzez badanie własności procesów roboczych i towarzyszących pracy maszyny, a także poprzez badanie własności wytworów maszyny [zól12], [cho93].



Rysunek 2.1. Schemat ogólny procesu diagnostycznego. Źródło: rys. 6.17 w [źól96].

Gdy badany obiekt znajduje się w stanie niezdatności, czyli występuje w obiekcie usterka lub awaria, to obserwowanymi symptomami są różnego typu wady, uszkodzenia czy blokady. Na podstawie tych symptomów, próbujemy ustalić stan urządzenia nie tylko obecny, ale również wcześniejszy i przyszły. Jednak często wiedza, którą posiadamy jest niepełna i określenie tych stanów może być kłopotliwe [kor02].

Rozpoznawanie stanu nazywa się w diagnostyce: generowaniem, diagnozowaniem, prognozowaniem [kor02, s. 6]. Np. przekazany mechanikowi (lub producentowi samochodów) subiektywny opis zachowania się samochodu przez kierowcę tego samochodu może pozwolić na ustalenie usterki technicznej w tym samochodzie i jej lokalizację lub wskazać na pewne wady konstrukcyjne. W takim procesie diagnozowania występują kolejno po sobie odpowiednie fazy badania stanu urządzenia technicznego [kor02, s. 9], takie jak:

- detekcja – wykrycie, czyli zauważenie powstania uszkodzenia w obiekcie i określenie chwili, w której tego dokonano,
- lokalizacja – określenie rodzaju: miejsca i czasu wystąpienia uszkodzenia,
- identyfikacja – określenie rozmiaru i charakteru zmienności uszkodzenia w czasie.

W wyniku stosowania pewnych metod inżynierii wiedzy (tak je prezentuje Antoni Ligęza w [kor02]) otrzymuje się wiedzę diagnostyczną. Natomiast wnioskowanie odpowiadające tym metodom pozwala wypracować decyzje w procesie diagnozowania. Zgodnie z rysunkiem 2.1. przyjęty model diagnostyczny umożliwia jawne lub niejawne odwzorowanie relacji **symptom** – **stan** oraz pozwala na ustalenie stanu obiektu na podstawie symptomów. Wymienione wnioskowanie zwane jest **wnioskowaniem diagnostycznym**. Podstawowy łańcuch przyczynowo-skutkowy rozważany podczas wnioskowania diagnostycznego obejmuje:

- diagnozowany stan – usterki, wady konstrukcyjne, wady technologiczne, wady użytkowe, uszkodzenia, zewnętrzne blokady lub niedostosowanie do środowiska użytkownika,
- proces wyjściowy,
- sygnał diagnostyczny,
- symptom stanu,
- diagnozę.

Tak rozumiane wnioskowanie będzie możliwe, gdy będzie istniał system pomiarowy rozumiany jako zbiór środków pomiarowych, konieczny i wystarczający do przeprowadzenia danego eksperymentu pomiarowego – **diagnozy** [roz82, s. 38].

Taki system pomiarowy można rozszerzyć o logikę rozmytą, jeżeli będzie on umożliwił rozmytą reprezentację wiedzy za pomocą egzemplarzy danych uzyskanych w tym systemie. System ten będzie w niniejszej pracy nazywany **systemem diagnostycznym** [żół96].

W diagnostyce, szczególnie heurystycznej, może mieć zastosowanie logika rozmyta. W diagnostyce heurystycznej podstawowymi typami obiektów są pewne symptomy (cechy) i diagnozy (rozwiązania) oraz reguły typu: cecha c wskazuje na rozwiązanie x [paw96, s. 23]. Można zauważyć podobieństwo takiej reguły do reguł logiki rozmytej. Również symptomy w takiej diagnostyce są rozmyte. Na przykładzie silnika samochodowego symptomy opisane w sposób rozmyty to: czarne spaliny, za wysokie zużycie paliwa, nietypowe odgłosy silnika, za małe przyśpieszenie itp. Natomiast rozwiązaniem (diagnozą) dla tego typu problemu może być np. nieszczelny system zasysania powietrza, złe ustawienie zapłonu, zanieczyszczony filtr powietrza.

Wnioskowanie w diagnostycznych systemach pomiarowych ma specyficzną problematykę wyznaczoną przez informatyczne aspekty pomiaru [roz82]. Z tego względu potrzebne jest przyjęcie ogólnej procedury przetwarzania danych diagnostycznych, zgodnej z ogólną strukturą systemu pomiarowego (procedury przetwarzania tych danych zgodnej z ogólnym modelem procesu pomiarowego).

Zgodnie ze schematem na rysunku 2.2. sygnały o obiekcie diagnozowanym są interpretowane na poziomie wejściowym systemu diagnostycznego. Zostają określone takie procedury, jak [roz82]:

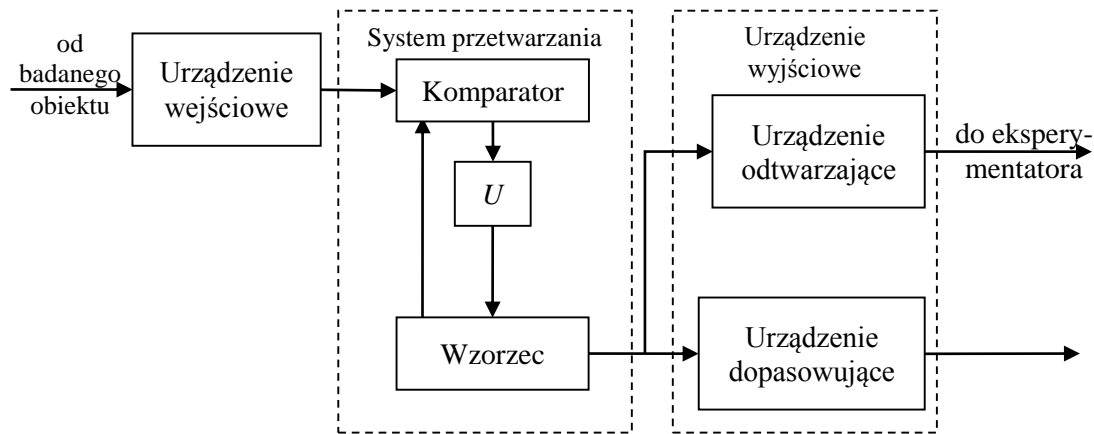
Pomiar – wyznaczenie wartości wielkości parametrów symptomów stanu diagnozowanego obiektu za pomocą odpowiednich środków pomiarowo-informacyjnych.

Metoda pomiarowa – sposób wykorzystania zasad i środków pomiarowych.

Środki pomiarowe – narzędzia informacyjno-techniczne wykorzystywane przy pomiarach i mające znormalizowane charakterystyki metrologiczne.

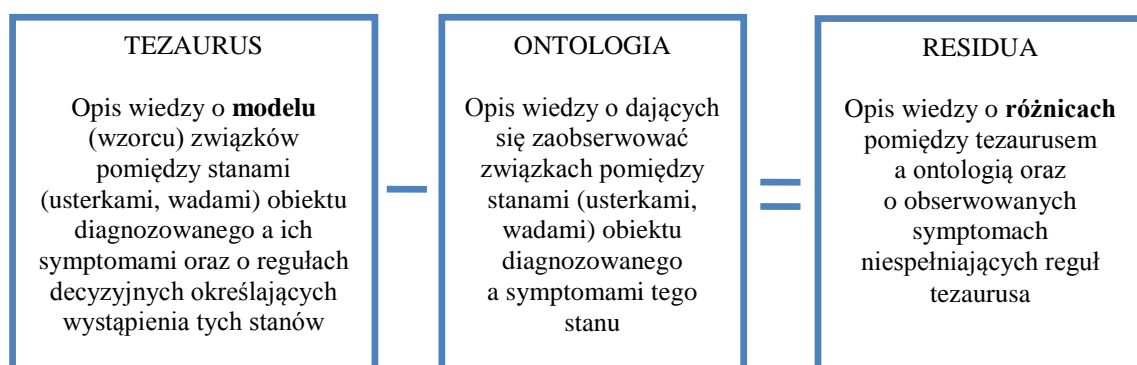
Dokładność pomiarów – odchylenie wartości pomiaru od rzeczywistej wartości wielkości mierzonej.

Rzeczywista wartość wielkości mierzonej – wartość parametru symptomu stanu obiektu diagnozowanego, która w sposób dokładny odzwierciedla rzeczywistą własność tego obiektu pod względem ilościowym i jakościowym (miary i struktury zachowania obiektu).



Rysunek 2.2. Schemat struktury systemu pomiarowego [roz82].

Wymienionym procedurom odpowiada opis ich modeli, a więc tezaurus procesu diagnostycznego. Bieżące opisy atrybutów pomiaru, jako wyników zastosowania wymienionych procedur, składają się na ontologię procesu diagnostycznego. Dane zawarte w ontologii są przetwarzane w systemie przetwarzania, który zawiera: wzorzec, komparator i układ sterowania. Wzorzec to system danych, który odtwarza dane fizycznie lub informatycznie (np. komputerowo) z tezausa. Komparator realizuje pewną operację porównania danych ontologii ze wzorcem lub wzorców ze sobą w zależności od przyjętej metryki, wykorzystując do tego pewne algorytmy ustalone przez ekspertów dla procesów deterministycznych (określonych dokładnie, jednoznacznie) i dla procesów niedeterministycznych: statystycznych, Bayesa, Dempstera–Shafera, rozmytych oraz hybrydowych (stosujących różne z tych metod jednocześnie) [roz82].



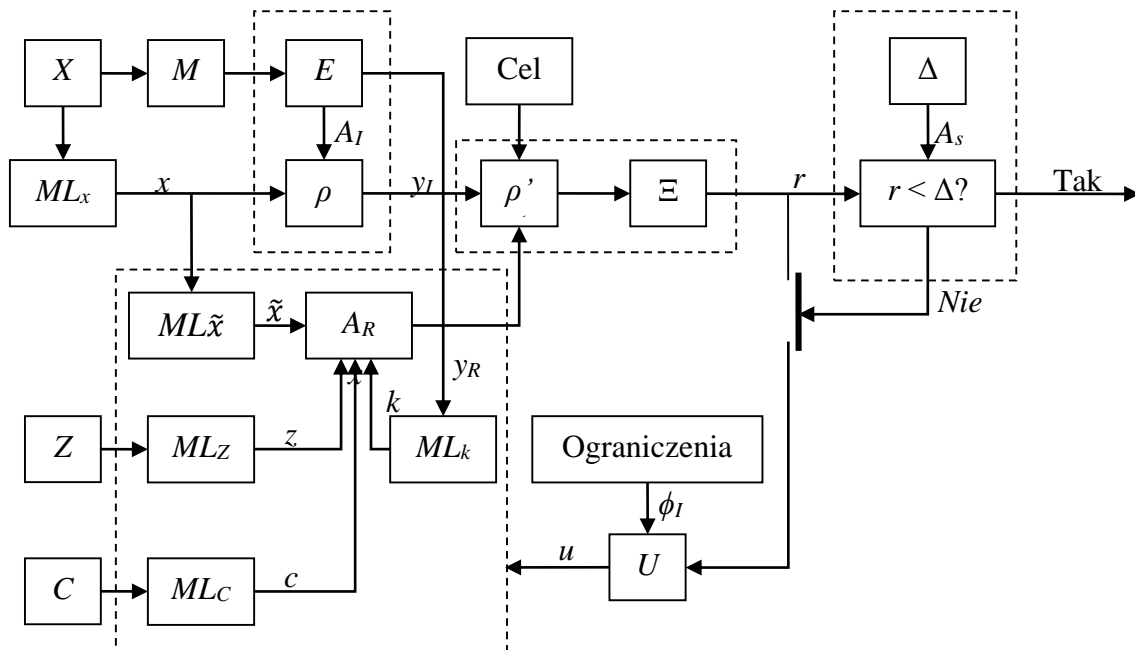
**Rysunek 2.3. Typy przetwarzanej wiedzy w systemie diagnostycznym.
Opracowanie własne na podstawie [kor02].**

Na rysunku 2.3. przedstawiono zależności pomiędzy tezauresem, ontologią a residuum. Tezaurus opisuje dane modelowe. Natomiast ontologia opisuje dane zebrane podczas

obserwacji rzeczywistego obiektu. Z kolei residua wyznaczają różnice pomiędzy tezauresem a ontologią.

Ustaleniem najlepszego wyniku porównania ontologii z tezauresem, zgodnie z przyjętym algorytmem, zajmuje się układ sterowania (U). Przetworzone dane skierowane są do urządzenia wyjściowego, w którym są odtworzone autonomicznie w urządzeniu odtwarzającym, według potrzeb eksperymentatora, lub nieautonomicznie dopasowane są w urządzeniu dopasowującym do udostępnienia w innych systemach (rysunek 2.2.) [kor02].

Na rysunku 2.4. przedstawiono model procesu pomiarowego.



Rysunek 2.4. Model procesu pomiarowego [roz82].

Analizując system diagnostyczny jako pewien system pomiarowy, przyjęto, że procedura przetwarzania danych diagnostycznych powinna być zgodna z modelem procesu pomiarowego przedstawionym na rysunku 2.4., gdzie:

X – jest procedurą ustalającą zbiór egzemplarzy danych pomiarowych.

E – jest zbiorem etalonów, czyli wzorcowych wartości atrybutów (konceptów i ról) z tezausa.

Na mechanizmy pomiaru składają się następujące procedury:

M – tezaurus określony przez zbiór atrybutów (parametrów), których wartości są mierzone w procedurze X .

ML_x – procedura tworzenia ontologii poprzez losowe przypisywanie poszczególnym atrybutom tezausa pewnych wartości x ustalonych przez procedurę X .

A_I – procedura ustalająca zgodnie z kryterium ρ etalony (wzorcowe wartości) odpowiadające atrybutom ontologii i ich losowo wybranym wartościom.

$ML_{\bar{x}}$ – procedura analizująca proces losowania wartości \mathbf{x} ; procedura ta określa statystyczny charakter problemu pomiarowego, przekazując dane statystyczne do porównania z danymi modelowymi za pomocą procedury A_R .

ML_Z – procedura ustalająca parametry z reprezentujące warunki przeprowadzania pomiaru (diagnozy); uzyskane dane mogą mieć wpływ, zgodnie z metodami Bayesa, na dane statystyczne.

ML_c – procedura ustalająca wiarygodność danych niepewnych, określoną przez odchylenia c wyników pomiaru od danych statystycznych $ML_{\bar{x}}$. Odchylenie jest wywołane rozrzutem technologicznym, zużyciem środków pomiarowych, przestarzałymi lub niepasującymi metodami pomiaru, subiektywną interpretacją sygnałów o stanie obiektu diagnozowanego (np. przekonaniem eksperymentatora) i innymi wpływami na wartość pomiaru niezwiązanymi bezpośrednio z procesem pomiaru.

ML_k – procedura ustalająca odchylenie k wartości parametrów wzorca w wyniku niedoskonałości w przekazywaniu w systemie diagnostycznym poprawnej miary etalonów czy w wyniku niekompletności, niepełności teaurusu lub niedoskonałości (niekiedy błędności) stosowanej teorii. Ta niepewność w przekazywaniu poprawnej miary etalonów prowadzi do wiedzy niepewnej, gdzie odchylenie k mierzy pewien stopień rozmycia reprezentacji tej wiedzy.

A_R – procedura ustalająca optymalną wartość pomiaru y_R na podstawie wyniku procedur $ML_{\bar{x}}$, ML_Z , ML_c i ML_k przy parametrach sterujących u przyjętych dla ograniczeń środków pomiarowych φ .

ρ' – kryterium porównania, zgodnie z celem pomiaru, ustalające parametry zwane *residuami*. Jest to potrzebne do wyznaczenia przez procedurę E błędu pomiaru r . Jest to możliwe, gdyż rozkład losowy przyjęty w $ML_{\bar{x}}$ zostaje skorygowany przez procedury ML_Z , ML_c i ML_k . Staje się wtedy realnym rozkładem losowym występującym w procesie pomiaru i zgodnie z metrologią można poprawnie określić błąd pomiaru. Residua reprezentują wiedzę o różnicy między danymi systemu pomiarowego (diagnostycznego) opisanymi przez teaurus a danymi opisanymi przez ontologię (rys. 2.4.).

A_s – procedura prowadząca do porównania błędu pomiaru r z tolerancją (dopuszczalnym błędem) Δ , czyli odpowiadana na pytanie, czy $r < \Delta$.

U – układ sterujący. Jest on włączany, gdy błąd pomiaru jest niedopuszczalny ($r > \Delta$).

Podsumowując, należy stwierdzić, że wyróżnione wyżej mechanizmy pomiaru: M , $ML_{\bar{x}}$, ML_Z , ML_c , ML_k , wyznaczają odpowiednio dane:

- *deterministyczne* – modelowe,
- *statystyczne* – zależne od obiektów mierzonych, a niezależne od systemu pomiarowego,
- *sieci Bayesa* – zależne od warunków pomiaru,
- *niepewne* – zależne od określonego przez ekspertów środowiska pomiaru, natomiast ich wiarygodność jest mierzona np. metodą Dempstera–Shafera,

- *rozmyte* – pozostałe niedeterministyczne dane, zależne od nieokreśloności występującej w przekazywaniu poprawnej miary etalonów, ze względu na możliwość zastosowania reguł wnioskowania.

W systemie pomiarowym wymienione kolejno mechanizmy pomiaru wyznaczają dane reprezentujące wiedzę z różnymi poziomami niepewności, ale równocześnie pozwalają tak zmodyfikować dane statystyczne, aby je jak najbardziej zbliżyć do danych modelowych [roz82].

Niepewność danych w diagnostyce technicznej może wynikać z różnych przyczyn. Dane diagnostyczne mogą być [moc02]:

- *niekompletne*, gdy tor pomiarowy jest niesprawny np. uszkodzony jest czujnik;
- *błędne*, gdy nie uwzględnimy błędów pomiaru lub sprzeczności przy gromadzeniu danych;
- *obciążone niepewnością*, gdy pojawiają się szумы na wejściu lub wyjściu;
- *niedokładne*, gdy kwantujemy wartości cech ciągłych, gromadzimy dane jakościowe lub wynikające z własności układów pomiarowych oraz techniki określania własności cech;
- *reprezentowane za pomocą wartości jakościowych*, gdy są formułowane przez człowieka, eksperta z danej dziedziny.

System diagnostyczny pozwala na ustalenie stanu obiektu na podstawie symptomów w procesie wnioskowania. Przejdziemy teraz do dokładniejszego omówienia wnioskowania diagnostycznego.

2.1.3. Wnioskowanie diagnostyczne

Diagnostyka to proces wnioskowania, który definiuje pewne właściwości (takie jak wystąpienie usterek, nieoczekiwany pomiar lub odchylenie od normalnego zachowania) i który jest utrzymywany w pewnym czasie w systemie (diagnostyka usterek) lub w środowisku (diagnostyka sytuacji). Proces ten zmierza do identyfikacji przyczyny danego problemu [cho93]. W diagnostyce systemów używa się różnych metod reprezentacji wiedzy. W systemach diagnostycznych z bazą wiedzy (ang. *knowledge-based diagnosis system*), w przeciwieństwie do systemów opartych na symulacjach, istotne jest, jak reprezentować wiedzę w modelu diagnostycznym oraz jak tworzyć silnik wnioskowania diagnostycznego [lig04].

W diagnostyce często, aby stwierdzić, czy diagnoza jest prawidłowa, najpierw należy ustalić, jak dany obiekt powinien działać. W tym celu stosuje się teoretyczne modele diagnostyczne oparte na wnioskowaniu diagnostycznym, które tworzą pewien schemat działania danego urządzenia. Porównując działanie rzeczywistego obiektu z jego modelem matematycznym, można uzyskać informację, czy obiekt rzeczywisty jest zgodny z obliczonym modelem, a co za tym idzie – czy model jest prawidłowy [dav92]. W tym celu stosuje się również metadiagnostykę, która służy do diagnostyki systemu diagnostycznego [bell1].

Wnioskowanie diagnostyczne należy do metod sztucznej inteligencji zastosowanych w diagnostyce obiektów. Wnioskowanie diagnostyczne jest istotne z wielu względów, szczególnie z powodu: bezpieczeństwa (głównie w nadzorowanych systemach), automatyzacji

(automatyczne decyzje inteligentnych agentów w ich środowisku) czy reorganizacji biznesowych strategii po usterce. Wnioskowanie diagnostyczne może zostać również wykorzystane w celu weryfikacji wyników symulacji. Tworzenie modeli diagnostycznych z wykorzystaniem różnych metod wnioskowania jest tematem wielu publikacji, między innymi [reg83], [reg85], [rei87], [tza89], [lam06], [dav92], [poo88]. Metody wnioskowania diagnostycznego mają również zastosowanie, gdy analizujemy system z wieloma usterkami jednocześnie [lig08].

Systemy diagnostyczne oparte na modelach mają wiele zalet, ale również wiele wad. Jedną z nich jest to, że wyniki otrzymywane z systemu nie zawsze są zrozumiałe dla użytkownika, który nie ma specjalistycznej i zaawansowanej wiedzy z danej dziedziny. W celu interpretacji otrzymanych wyników można wykorzystać ontologię oraz opis lingwistyczny wiedzy [kit99]. Jeśli bazę wiedzy przedstawimy za pomocą sieci semantycznej, to do opisu tej wiedzy można użyć logiki opisowej (ang. *Description Logic* – DL) opartej na ontologicznej reprezentacji wiedzy. Wnioskowanie diagnostyczne na podstawie DL zostało przedstawione w pracach [qi09], [du08], [du11].

2.1.4. Wnioskowanie diagnostyczne z zastosowaniem logiki rozmytej

Logika rozmyta jest szeroko wykorzystywana w diagnostyce. W literaturze można znaleźć wiele przykładów zastosowania wnioskowania rozmytego do automatycznej diagnostyki procesów [ter78], [uli96], detekcji błędów systemu [tin08] oraz identyfikacji nagłych zmian w diagnozowanym obiekcie [sar11]. Wnioskowanie rozmyte ma również zastosowanie w przypadku rozproszonych systemów diagnostycznych [koś08].

Bardzo istotne są także badania, w których uwzględnia się stochastyczny i probabilistyczny charakter diagnozowanych procesów. Wtedy rozmyta baza wiedzy uwzględnia prawdopodobieństwo wystąpienia rozmytych zmiennych lingwistycznych na wejściu lub wyjściu systemu. W takim diagnostycznym systemie ekspertowym powstaje probabilistyczno-rozmyta baza wiedzy [wal11a], [wal11b], [wal13], [rud11]. Wykorzystanie logiki rozmytej i regułowej bazy wiedzy, opisującej wiedzę niepewną i niepełną w języku naturalnym, uwzględnia wiedzę eksperta, natomiast wykorzystanie danych empirycznych pozwala na określenie prawdopodobieństwa stwierdzeń warunkowych utworzonego modelu.

Obecnie wiele badań dotyczy systemów diagnostycznych z bazami wiedzy (ang. *knowledge-based systems*). Jednak wiele trudności sprawia pozyskiwanie wiedzy do takiego systemu. Jednym z powodów, dla których występują te trudności, jest kierunkowość reguł. Reguły wiedzy pozyskiwane od ekspertów mają zwykle postać: „Jeżeli wydarzy się jakaś usterka, to pojawią się określone symptomy”. Systemy ekspertowe wymagają, aby reguły były zapisane w drugą stronę, tj.: „Jeżeli pojawia się symptom, to występuje dana usterka”. Oznacza to, że baza wiedzy jest budowana w odwrotny sposób, niż podana jest wiedza przez eksperta. Aby uniknąć tego problemu, w diagnostyce stosuje się rozumowanie abdukcyjne (ang. *abductive reasoning*) [pen90], [pop73]. Rozumowanie to pozwala na podstawie obserwowanych symptomów stworzyć hipotezę lub hipotezy, które określą przyczynę danego symptomu. Empiryczne dane

uzyskiwane w trakcie obserwacji obiektu diagnostycznego dają podstawę do formułowania bazy reguł systemu diagnostycznego. Możliwa jest również modyfikacja tych reguł, jeśli zostaną dostarczone nowe dane do systemu.

We wnioskowaniu diagnostycznym często stosuje się logikę rozmytą do określenia stopni nasilenia danych symptomów [yam95a], [yam95b], [yam95c]. Gdy do opisu wiedzy niepewnej używamy logiki rozmytej, to wnioskowanie diagnostyczne oparte na logice opisowej należy poszerzyć o logikę rozmytą [zad65]. W tym celu używana jest rozmyta logika opisowa (ang. *Fuzzy Description Logic – fuzzyDL*) [baa03], [bob08], [str98], która dokładniej została opisana w rozdziale 2.4.

2.2. Modele wiedzy diagnostycznej w systemach wyszukiwania informacji

Wychodząc od pojęcia wiedzy w diagnostyce technicznej, można zauważyć za W. Moczulskim [kor02, s. 624-625], że: „W odniesieniu do człowieka wiedza to ogół tego, co dany człowiek wie. Wiedza w danej dziedzinie dotyczy obiektów (jak maszyny i ich zespoły) i klas obiektów, związków pomiędzy obiektami i ich klasami. Wiedza ta obejmuje także umiejętności, rozumienie praw ogólnych, procedur postępowania itp. [...] W diagnostyce technicznej celowe jest także rozpatrywanie zgromadzonego doświadczenia i umiejętności. [...] termin «wiedza» oznaczać będzie zarówno wiedzę, jak i praktyczne doświadczenie specjalisty. [...] wiedza diagnostyczna dotyczy zarówno faktów, jak i procesów. Z tego i wielu innych jeszcze względów, jak np. reprezentacji i stosowania wiedzy, celowe jest rozróżnienie wiedzy deklaratywnej i wiedzy proceduralnej”.

Można się zgodzić z W. Moczulskim [kor02, s. 626], że analiza dostępnych prac dotyczących **pozyskiwania wiedzy** z zakresu diagnostyki technicznej wskazuje na: nadmierne wykorzystywanie pośrednictwa „inżyniera wiedzy”; dominację uczenia maszynowego na podstawie przykładów diagnostycznych wstępnie sklasyfikowanych; małą intensywność wykorzystywania wiedzy specjalistów oraz brak ogólnie uznanej metodologii. Z tego powodu metody pozyskiwania wiedzy powinny obejmować [kor02, s. 627]: wybór sposobów reprezentacji danych; wybór metod reprezentowania wiedzy; wybór metod pozyskiwania wiedzy i ich weryfikacji. Zaproponowany w przywołanej pracy aparat pojęciowy sprowadza się do określenia wspólnych dla specjalistów procedur przetwarzania typów danych diagnostycznych, w wyniku czego uzyskuje się wiedzę diagnostyczną obejmującą to, co się wie w sensie zarówno dokładnym, jak i niedokładnym (niedoskonałym) [fla11, rozdz. 12 i 13] oraz czego się doświadcza w procesie diagnozy. Poniżej zostanie pokazane, że takie podejście jest bliskie ujęciu informatycznemu, które silnie wspiera automatyzację procesów.

Podstawowymi pojęciami informatyki, od jej początków, charakteryzującymi **systemy danych**, są: dane; egzemplarze danych, w tym dane jako obiekty; dane wskazujące konkretne obiekty; dane jako reprezentacje wiedzy o obiektach; typy danych; agenci naturalni i sztuczni (jako podmioty przetwarzania danych); wiedza oraz atrybuty wyznaczone przez klasyfikację obiektów zgodnie z wyróżnioną wiedzą [abe02], [tur76], [wai89]. W świetle badań nad sztuczną inteligencją, w tym nad sieciami semantycznymi Web i komputerowym

wspomaganiem projektowania technicznego, poszukuje się pewnych sposobów algorytmizacji rozpoznawania wiedzy. Istotne jest wyszukiwanie danych badawczych i technicznych w warunkach, w których pozyskana wiedza może być uważana za niepewną, niejasną lub nieostrą, gdzie wykorzystana jest logika rozmyta. Takie badania, jak w [gal04], [kno02], wymagają uściślenia aparatu teoretycznego opisującego systemy danych. W tych badaniach istotne są następujące, najczęściej występujące, aplikacje systemów danych: tablice decyzyjne i sieci semantyczne. **Tablice decyzyjne** określone są przez *język atrybutów* obiektów (lub stanów) [lig05], [lig10, s. 261-283]. **Sieci semantyczne** są zbiorami konceptów i ról oraz są ilustrowane jako graf indeksowany [kow79b], [mag78]. Natomiast koncepty są zbiorami obiektów (danych), a role są relacjami pomiędzy tymi obiektami.

2.2.1. Formalnojęzykowa struktura świata rzeczywistego a sieci semantyczne

Przyjmijmy za Jeffreyem Ullmanem [ull88, s. 25], że encjami jest wszystko to, co w rzeczywistości może istnieć, jest rozróżnialne i jednoznacznie jest rozpoznawalne. Innymi słowy, encje to wszystko to, czego istnienie można rozpoznać przez wskazanie, tj. odniesienie się do tego, identyfikację tego. Encje są oznakami czegoś. Dzięki encjom rozpoznajemy określoną teoriomnogościowo strukturę dowolnych obiektów rzeczywistości. Zgodnie z modelem Ullmana, świata rzeczywistego [ull88, s. 24-31], obiekty rozpoznaje się jako encje, encje encji (ich krotki), zbiory/kolekcje encji lub krotek encji, krotki tych zbiorów, zbiory takich wyróżnionych struktur itd. Elementy tej struktury także ujawniają się jako encje, np. konkretne napisy z teorii mnogości lub języka systemów danych, obrazy, diagramy lub fotografie. W modelu Ullmana uniwersum wszystkich obiektów można opisać następująco:

Niech zbiór *Encje* będzie zbiorem wszystkich encji, a zbiory *Obiekty*^(k), dla $k = 1, 2, 3, \dots$ zbiorami obiektów rzędu k . Określmy:

$$\begin{aligned} \text{Obiekty}^{(1)} &= \text{Encje} \cup \text{Encje} \times \text{Encje} \cup \dots \cup \text{Encje} \times \text{Encje} \times \dots \times \text{Encje} \cup \dots \\ \text{Obiekty}^{(n+1)} &= \text{Obiekty}^{(n)} \cup \wp(\text{Obiekty}^{(n)}) \cup \dots \\ &\dots \cup \wp(\text{Obiekty}^{(n)}) \times \wp(\text{Obiekty}^{(n)}) \times \dots \times \wp(\text{Obiekty}^{(n)}) \cup \dots \end{aligned} \quad (1)$$

Uniwersum wszystkich obiektów jest zbiór:

$$\text{Obiekty} = \text{Obiekty}^{(1)} \cup \text{Obiekty}^{(2)} \cup \dots \cup \text{Obiekty}^{(k)} \cup \dots \quad (2)$$

Obserwacja 2.1.

Ponieważ w świecie rzeczywistym można wskazać każde rozpoznawanie obiektu czy oznakę obiektu, stąd też rozpoznawanie/oznaka obiektu, o ile istnieje, jest encją.

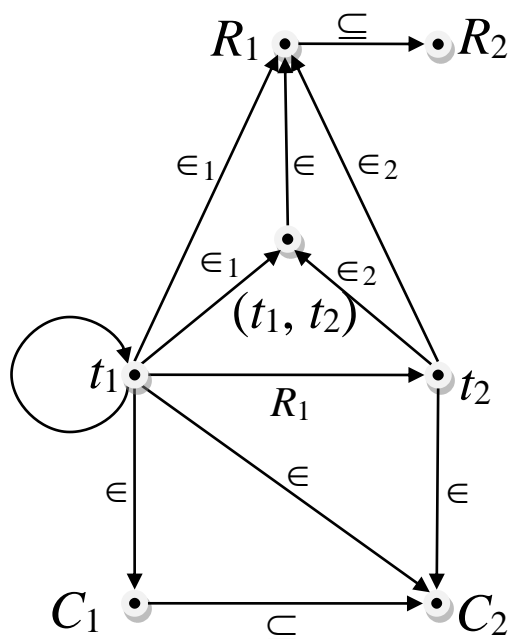
Jak rozpoznawana jest rzeczywistość?

Obserwacja 2.2.

Rzeczywistość to pewien zbiór obiektów, który rozpoznajemy, rozpoznając struktury relacyjne określone w zbiorze *Obiekty*. Rozpoznanie struktur relacyjnych polega na tym, że

wskazane, wyróżnione struktury zamieniane są na pewne encje. Każda rozpoznana cecha, relacja oraz obiekty posiadające daną cechę lub wchodzące w daną relację zastąpione są przez encję identyfikującą wymieniony obiekt (encję będącą oznaką wymienionego obiektu), np. przez egzemplarz nazwy takiego obiektu. Podobnie zbiory obiektów i relacje zamienia się na pewne encje, np. egzemplarze nazw cech i atrybutów, nazw wartości atrybutów lub nazw relacji (własności) czy zapis danych lub typów danych. Wszelkie egzemplarze nazw są encjami (oznakami) nazywanymi obiektów (desygnatów nazw), w tym każdy znak na tej kartce papieru jest encją tego, co można na niej wyczytać. W tym sensie jako fragment rzeczywistości rozpoznawane są wyniki pomiarów w systemach diagnostycznych oraz opisy symptomów i stanów diagnozowanego obiektu.

Wyróżnione w Obserwacji 2.2. encje dla rozpoznawanej struktury relacyjnej tworzą strukturę nazywaną **siecią semantyczną** [fla11, s. 96-103]. Zbiory encji wskazujących obiekty, posiadające daną cechę, nazywane są *konceptami*. Elementy konceptów zwane są *wystąpieniami tych konceptów*. Zbiory krotek encji wskazujących na obiekty pozostające w danej relacji (mające daną własność) nazywane są *rolami*, a encje wskazujące na ich elementy (encje krotek encji) – *wystąpieniami ról*. Przykładowy teoriomnogościowy schemat sieci semantycznej przedstawia rysunek 2.5.



Rysunek 2.5. Schemat sieci semantycznej. Opracowanie własne.

Symbole na rysunku mogą przybierać różne nazwy oraz występować jako różne encje, tj. egzemplarze nazw: pojęć $\{C_1\}$, obiektów $\{C_1, C_2, R_1, R_2, t_1, t_2, (t_1, t_2)\}$ i ról $\{R_1, E, E_1, E_2, \subseteq\}$.

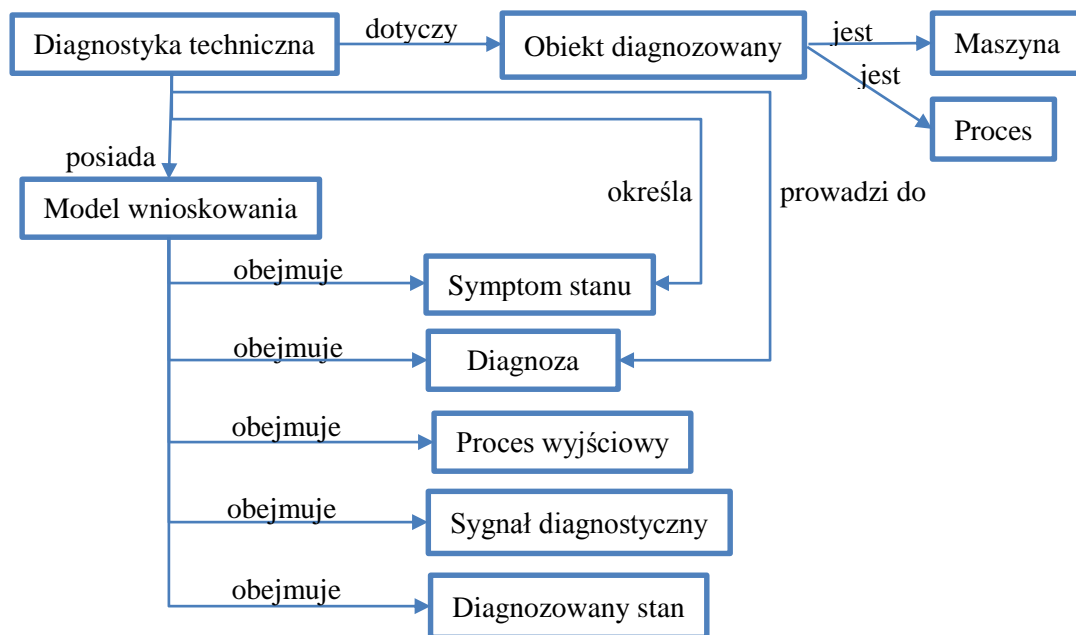
Zwrotna strzałka C_1 skierowana do t_1 oznacza, że „ t_1 jest wystąpieniem konceptu C_1 ” („pojęcie C_1 odnosi się do obiektu t_1 ”), gdzie t_1 i t_2 są nazwami indywiduów, a C_1 i C_2 są

nazwami konceptów. $t_1 \in C_1$ oznacza, że „ t_1 jest C_1 ”, „ t_1 jest wystąpieniem (przypadkiem wystąpienia) konceptu C_1 ” lub „ t_1 ma/posiada cechę C_1 ”.

Natomiast R_1, R_2 są nazwami ról. Strzałka R_1 od t_1 do t_2 oznacza, że t_1 jest w roli R_1 z t_2 . Zapis $(t_1, t_2) \in R_1$ oznacza, że „pomiędzy obiektami t_1 i t_2 zachodzi związek będący wystąpieniem roli R_1 ”. Oznaczenia przedstawione na rysunku są rozumiane jako:

- $t_1 \in_1 R_1$ – „ t_1 jest poprzednikiem w relacji R_1 ”,
- $t_2 \in_2 R_1$ – „ t_2 jest następnikiem w relacji R_1 ”,
- $t_1 \in_1 (t_1, t_2)$ – „ t_1 jest pierwszym elementem pary (t_1, t_2) ”,
- $R_1 \subseteq R_2$ – „to, co jest w relacji R_1 , jest w relacji R_2 ”,
- $(t_1, t_2) \in R_1$ – „para t_1, t_2 spełnia relację R_1 ”,
- $C_1 \subseteq C_2$ – „to, co jest C_1 , jest C_2 ”.

Podane zwroty używane są do opisu fragmentów realnego świata w stosownych znaczeniach językowych. W realizacjach schematów sieci semantycznej symbole wystąpień konceptów i ról zastąpione są ich opisami (np. egzemplarzami ich nazw). Często wystąpienie konceptu ilustruje się prostokątami zawierającymi opisy tych konceptów, natomiast wystąpienie roli oznaczane jest strzałką (rysunek 2.6).



Rysunek 2.6. Przykładowa sieć semantyczna dla diagnostyki technicznej. Opracowanie własne.

Obserwacja 2.3.

W świecie rzeczywistym Ullmana dla dowolnego zbioru X encji istnieje taki zbiór Y encji, że dla dowolnego ciągu encji ze zbioru X istnieje dokładnie jedna encja ze zbioru Y , będąca oznaką tego ciągu, a więc wskazująca ten ciąg.

Zgodnie z Obserwacją 2.3. można stwierdzić, że:

Obserwacja 2.4.

Dla encji występujących w dowolnej sieci semantycznej istnieje najmniejszy zbiór A encji taki, że zbiór A^* wszystkich encji wskazujących na dowolne ciągi encji ze zbioru A zawiera encje tej sieci semantycznej. Zbiór A zwany jest zazwyczaj alfabetem, a jego elementy – symbolami. Podzbiory A^* zwane są **językami abstrakcyjnymi**, a elementy zbioru A^* są **słowami** [kow79b, s. 10]. Symbol wskazujący na ciąg pusty zwany jest słowem pustym. W językach, w których encje są linearnymi napisami, słowa są „sklejonymi” ze sobą linearnie, jedna za drugą, encjami alfabetu. Są to tzw. konkatenacje symboli. Na przykład dla ciągu symboli: ‘A’, ‘d’, ‘a’, ich konkatenacją, tj. encją ciągu tych symboli, jest napis „Ada”. Jak widać, encje nie są tu „zabrudzonymi” na czarno miejscami na białym tle, ale połączeniem takich miejsc z procedurami identyfikowania wzorców wymienionych liter napisu „Ada” oraz identyfikowania słowa „Ada”. Identyfikacja ta może być przeprowadzona przez człowieka lub program komputerowy.

Uzasadnione jest wprowadzenie następujących definicji:

Definicja 2.1. [kow79b, s. 8, 9]

Niech A będzie dowolnie wyróżnionym niepustym zbiorem symboli zwanym alfabetem, a Λ niech będzie pewnym wyróżnionym symbolem wskazującym na ciąg pusty, nienależącym do A , zwanym słowem pustym. Zbiór A^* nazywamy zbiorem słów nad alfabetem A , jeśli jest najmniejszym zbiorem X spełniającym warunki:

- 1) $\Lambda \in X$ oraz $A \subset X$,
- 2) jeżeli $x \in X, x \neq \Lambda$ oraz $v \in A$, to napis $vx \in X$,
- 3) żaden inny napis nie należy do X .

Dowolny podzbiór L zbioru A^* nazywamy językiem L nad alfabetem A .

Długość $\iota(s)$ słowa $s \in A^*$ określamy rekurencyjnie:

$$\iota(\Lambda) = 0; \text{ dla } v \in A, x \in A^*, \iota(vx) = 1 + \iota(x) \quad (3)$$

Dwuargumentową operację na słowach, zwaną konkatenacją (oznaczaną przez xy , dla słów x, y) określamy rekurencyjnie (dla dowolnych słów x, y): $\Lambda x = x\Lambda = x$. Jeśli $\iota(y) > 0$, to istnieje symbol v i słowo z takie, że $\iota(z) = \iota(y) - 1$, a napis vz jest równy (rozpoznany jako równokształtny) słowu y , wtedy: $yx = v(zx)$. Dla dowolnych słów $x, y, z \in A^*$, $x(yz) = (xy)z$.

Wprowadzone pojęcia nie są wystarczające do ustalenia tego, czym są rozpoznawane obiekty, tj. do ustalenia wiedzy o tych obiektach, ale wystarczają do zbudowania modelu językowej identyfikacji encji obiektów (egzemplarzy opisów tych obiektów), zwanego **gramatyką bezkontekstową**.

Gramatyka bezkontekstowa [kow79b, s. 14]

Gramatyką bezkontekstową nazywamy układ $G = \langle V, T, P, S \rangle$, gdzie:

- 1) V jest zbiorem symboli zmiennych (symboli nieterminalnych),

- 2) T jest zbiorem symboli terminalnych,
- 3) P jest zbiorem reguł produkcji, określonym dla wszystkich symboli zmiennych,
- 4) S jest symbolem wyjściowym (zwanym też początkowym lub głową).

Symbole zmiennych oznaczamy dużymi literami: $A, B, \dots, S \dots$, a symbole terminalne małymi literami: a, b, \dots, s, \dots . W konkretnych przykładach tym symbolom przypisuje się odpowiednio wyróżnione ich nazwy. Spełnione są warunki:

- 1) $V \cap T = \emptyset$ – rozłączność zbiorów zmiennych i symboli terminalnych,
- 2) $(V \cup T) \setminus \{S\} = A$ – alfabet jest zbiorem zmiennych i symboli terminalnych bez symbolu wyjściowego,
- 3) $S \in V$ i $P \subseteq V \times A^*$ – symbol wyjściowy jest zmienną, a reguły produkcji są relacją pomiędzy zmiennymi i słowami ze zbioru A^* .

Regułę produkcji $\langle M, x \rangle \in P$ zazwyczaj zapisuje się jako $M \rightarrow x$.

Słowo y wynika bezpośrednio ze słowa x (piszemy $x \rightarrow y$), jeżeli słowo $x = pMk$, dla pewnych słów p, k oraz symbolu zmiennej M , a $y = puk$ oraz $M \rightarrow u$. Słowo y wynika pośrednio ze słowa x (co piszemy: $x \rightarrow^* y$) [kow79b, s. 15], jeżeli $x = y$, bądź istnieje taki ciąg słów $u_0, u_1, \dots, u_n, u_0 = x, u_n = y$, że:

$$u_0 \rightarrow u_1, u_1 \rightarrow u_2, \dots, u_{n-1} \rightarrow u_n \quad (4)$$

Dla dowolnego ciągu słów u_0, u_1, \dots, u_n układ $u = \langle u_1, u_2, \dots, u_n \rangle$ nazywamy procedurą wywodu, jeśli $n = 1$ lub

$$u_1 \rightarrow^* u_2, u_2 \rightarrow^* u_3, \dots, u_{n-1} \rightarrow^* u_n \quad (5)$$

Przez język $L(G)$ generowany przez gramatykę G [kow79b, s. 16] rozumie się zbiór takich słów $y \in T^*$, że $S \rightarrow^* y$. Słowo $x \in (V \cup T)^*$ takie, że $x \in L(G)$, nazywamy akceptowanym przez gramatykę G . Natomiast język $L \subseteq L(G)$ nazywamy językiem akceptowanym przez gramatykę G . Językiem bezkontekstowym nazywamy język generowany przez gramatykę bezkontekstową.

Gramatykę bezkontekstową G nazywamy *regularną* [kow79b, s. 41], jeżeli każda jej produkcja jest postaci $A \rightarrow v$ lub $A \rightarrow vB$, gdzie A, B są symbolami nieterminalnymi, a v jest symbolem terminalnym.

Relację dwuargumentową \leq na słowach ze zbioru $(V \cup T)^*$ nazywamy porządkiem *prefiksowym*, jeśli: $x \leq y$ wtedy i tylko wtedy, gdy istnieje takie słowo z , że $y = xz$.

Relację tę nazywamy porządkiem *infiksowym*, jeśli: $x \leq y$ wtedy i tylko wtedy, gdy istnieją takie słowa z_1, z_2 , że $y = z_1xz_2$.

Porządek prefiksowy jest porządkiem infiksowym, ale nie odwrotnie.

Dla dowolnej gramatyki regularnej G oraz słów $x, y \in (V \cup T)^*$ oraz porządku prefiksowego \leq , jeśli $x \rightarrow^* y$, to $x \leq y$.

Przykładowymi gramatykami są gramatyki struktur frazowych Chomskiego. Ich opis teoretyczny to gramatyki języków abstrakcyjnych w lingwistyce matematycznej.

Zmiennymi są kategorie gramatyczne, zapisywane np. w nawiasach: <zdanie>, <podmiot>, <orzeczenie>, <czasownik>, <przymiotnik> itp.

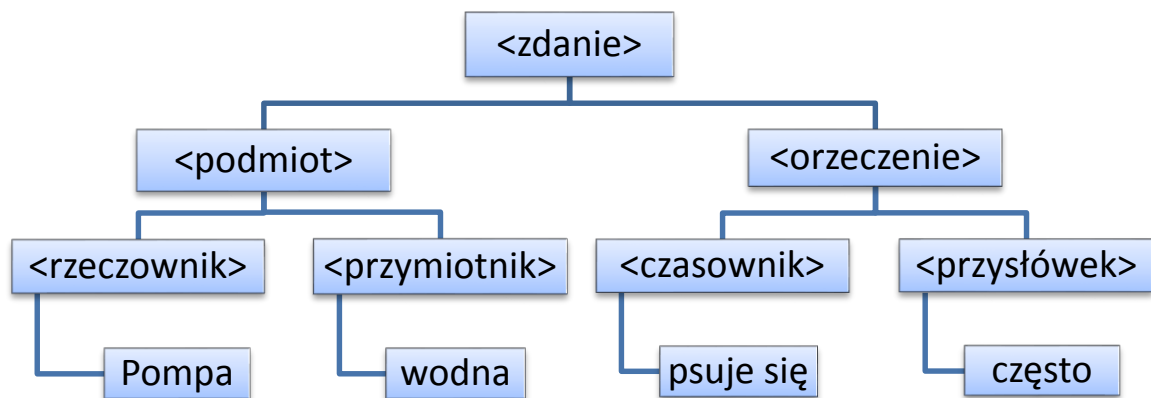
Wyrażenia zapisujemy następująco. Dla wyrażenia „Pompa wodna psuje się często” mamy:

- <zdanie> → <podmiot> <orzeczenie>
- <podmiot> → <rzeczownik> <przymiotnik>
- <orzeczenie> → <czasownik> <przysłówek>

W tym przypadku wyrażenia określają:

- <rzeczownik> → Pompa
- <przymiotnik> → wodna
- <czasownik> → psuje się
- <przysłówek> → często

Wywód został przedstawiony na rysunku 2.7.



Rysunek 2.7. Diagram przedstawiający gramatykę struktur zdania „Pompa wodna psuje się często”.
Opracowanie własne.

Bardzo dogodną notacją używaną do definiowania języków programowania jest notacja Backusa–Naura. Pozwala ona długie ciągi reguł postaci: $M \rightarrow u_1, M \rightarrow u_2, \dots, M \rightarrow u_n$, zapisać bardziej zwięźle i czytelnie:

$$M ::= u_1 | u_2 | \dots | u_n \quad (6)$$

Zapis ten rozumie się także w ten sposób, że wyrażeniami tej samej kategorii syntaktycznej co M są wyrażenia u_1, u_2, \dots, u_n . Zastępując M przez słowa u_1, u_2, \dots, u_n , otrzymujemy wyrażenia tej samej kategorii syntaktycznej.

Przykładem jest *gramatyka języka wyrażeń algebraicznych* [kow79b, s. 20], w której:

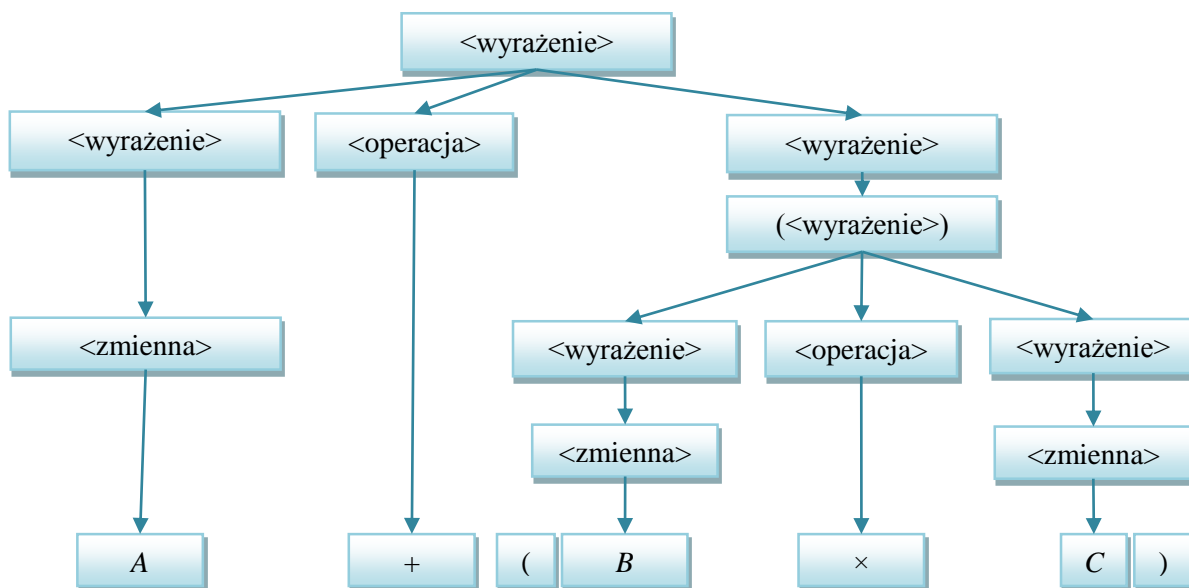
- 1) Zmiennymi są: <zmienna>, <operacja>, <wyrażenie>.
- 2) Symbolami terminalnymi są: litery A, B, \dots, Z oraz symbole działań: $+, -, \times, \div$.
- 3) Symbolem wyjściowym jest symbol <wyrażenie>.
- 4) Reguły produkcji tego języka wyglądają następująco:
 - <zmienna> ::= $A | B | \dots | Z$,

- $\langle \text{operacja} \rangle ::= + | - | \times | \div$,
- $\langle \text{wyrażenie} \rangle ::= \langle \text{zmienna} \rangle | \langle \text{wyrażenie} \rangle \langle \text{operacja} \rangle \langle \text{wyrażenie} \rangle | (\langle \text{wyrażenie} \rangle)$.

Poniżej podano przykłady wywodu dla gramatyki języka wyrażeń algebraicznych:

- 1) $\langle \text{wyrażenie} \rangle \rightarrow \langle \text{wyrażenie} \rangle \langle \text{operacja} \rangle \langle \text{wyrażenie} \rangle \rightarrow \langle \text{zmienna} \rangle \langle \text{operacja} \rangle (\langle \text{wyrażenie} \rangle) \rightarrow A + (\langle \text{wyrażenie} \rangle \langle \text{operacja} \rangle \langle \text{wyrażenie} \rangle) \rightarrow A + (B \times C)$.
- 2) $\langle \text{wyrażenie} \rangle \rightarrow \langle \text{wyrażenie} \rangle \langle \text{operacja} \rangle \langle \text{wyrażenie} \rangle \rightarrow \langle \text{wyrażenie} \rangle \langle \text{operacja} \rangle \langle \text{wyrażenie} \rangle \langle \text{operacja} \rangle \langle \text{wyrażenie} \rangle \rightarrow \langle \text{zmienna} \rangle \langle \text{operacja} \rangle \langle \text{zmienna} \rangle \langle \text{operacja} \rangle \langle \text{zmienna} \rangle \rightarrow A + B \times C$.

Diagram przedstawiający pierwszy wywód przedstawiono na rysunku 2.8.



Rysunek 2.8. Diagram przedstawiający przykład gramatyki języka wyrażeń algebraicznych. Opracowanie własne.

Ważną klasą języków, które pojawiają się podczas wyszukiwania informacji w Internecie, są **języki znakowania** (zwane też językami znacznikowymi) [hop05, s. 182-184]. W tych językach kategorie syntaktyczne budowy dokumentów internetowych określone są przez umieszczenie elementów tych dokumentów pomiędzy znakami zwanymi *znacznikami*. Przykładem języka znacznikowego jest *gramatyka języka HTML* (skr. ang. *Hyper Text Markup Language* – język znakowania hipertekstu) [hop 2005, s. 183]. W gramatyce języka HTML określone są następujące kategorie syntaktyczne:

- 1) **Tekst** – dowolny łańcuch znaków (słowo języka), które można interpretować w sensie syntaktycznym dosłownie, tzn. niezawierający znaczników, np. „niesprawna pompa wodna”.
- 2) **Znak** – dowolny łańcuch złożony z pojedynczego znaku dozwolonego w HTML wraz ze spacjami.
- 3) **Dok** – ciąg jednego lub więcej elementów dokumentu.
- 4) **Element** – łańcuch tekstów lub dokument występujący po znacznikach.

5) **Pozycja** – dokument poprzedzony znacznikiem listy (jest to element listy).

6) **Lista** – ciąg jednej lub więcej pozycji listy.

Nazwy kategorii są nazwami zmiennych gramatyki. Symbolami terminalnymi są znaki tekstów oraz znaczniki, np. A, B, \dots, a, b, \dots , ' ' (spacja), <P>, </P>, , , , ... Zmienne zaznaczone są kursywą, np. *Znak, Tekst, ...*, natomiast ε oznacza słowo puste.

Reguły produkcji określone są następująco:

1) $Znak ::= a | A | \dots$,

2) $Tekst ::= \varepsilon | Znak | ZnakTekst$,

3) $Dok ::= Tekst | Element Dok$,

4) $Element ::= TekstTekst | Dok | <P> Dok </P> | Lista | \dots$,

5) $Pozycja ::= Dok$,

6) $Lista ::= Pozycja | Pozycja Lista$.

Słowa mogą być sklejane, np. *TekstTekst, ZnakTekst*, lub łączone spacją, np. *Element Dok, Pozycja Lista*.

Językiem znakowania służącym do definiowania kategorii semantycznych, w tym kategorii diagnostycznych, jest język **OWL** (skr. ang. *Web Ontology Language*) [owl14], [ant04]. Charakterystyka gramatyki języka OWL zostanie przedstawiona w rozdziale 3.2.

2.2.2. Agenci w systemach komunikowania się

Pod koniec XX wieku wraz z popularyzacją komputerów klasy PC coraz większego tempa nabrał rozwój Internetu. Internet stał się źródłem wiedzy. Jednakże ze względu na nieuporządkowany charakter przechowywania informacji w sieci brakowało, i dalej brakuje, kontroli nad jakością i kompletnością dostępnej w niej wiedzy. Coraz trudniej znaleźć w Internecie poszukiwaną przez nas informację. Dużo czasu traci się na przeglądanie i filtrowanie linków zwróconych przez wyszukiwarkę internetową. Jednym z celów informatyki jest maksymalne wyręczenie człowieka w czynnościach, do których nie zawsze potrzebna jest inteligencja człowieka. Pojawiła się idea stworzenia internetowego, spersonalizowanego „szperacza”, zwanego potem **agentem** lub **multiagentem** [fla11, s. 206-214], który na podstawie wiedzy ułatwia korzystanie z Internetu. Innym powodem powstania idei stworzenia agenta programowego stało się spopularyzowanie urządzeń mobilnych, takich jak laptopy i palmtopy. Mała moc obliczeniowa, słabe łącza internetowe oraz konieczność zminimalizowania użycia procesora w celu umożliwienia jak najdłuższego korzystania z baterii spowodowały, iż pojawił się pomysł, aby odciążyć urządzenia mobilne poprzez zdalne wykonywanie zadań. Przykładowo użytkownik laptopa zlecałby wykonanie zadania, następnie zadanie przesyłane byłoby do jednej lub wielu innych stacjonarnych maszyn w celu wykonania obliczeń zdalnie. Kolejną z przyczyn powstania idei opracowania i prac nad systemami agentowymi było stałe dążenie programistów do jak najbardziej uniwersalnego podejścia do dekompozycji problemów świata rzeczywistego. W literaturze idea tworzenia oprogramowania agentowego po raz pierwszy pojawiła się około 1994 roku [rus95].

Podsumowując, należy stwierdzić, że głównymi przyczynami rozwoju technologii agentowej były:

- wzrost potrzeb użytkowników,
- rosnąca liczba informacji w Internecie,
- rozwój technologii mobilnych,
- dążenie do stworzenia jak najbardziej naturalnego sposobu projektowania oprogramowania.

Nie istnieje jedna ogólnie przyjęta ani jednoznaczna definicja agenta. Można zaobserwować, że każdy ośrodek pracujący nad technologią agentową stworzył własną definicję agenta. Często definicje te są wzajemnie sprzeczne oraz niekompletne. W pracy [pap09] autor przedstawia cztery popularne definicje agenta:

- Cokolwiek, co może być uznane za obserwujące otoczenie poprzez sensory i działające w ramach tego otoczenia [rus95].
- Autonomiczne jednostki zdolne do interakcji w swoim środowisku [ode00]. Jednostki programowe podejmujące działania w imieniu użytkownika lub innych programów, w pewnym stopniu niezależnie lub autonomicznie, które działając stosują pewną wiedzę lub reprezentację celów, potrzeb użytkownika określonych w protokole SNMP (ang. *Simple Network Management Protocol*) przez IBM w 1997 r. [ibm97].
- Zamknięty system komputerowy znajdujący się w pewnym otoczeniu, mający umiejętność elastycznego działania w tymże otoczeniu. Działania systemu polegają na realizowaniu celów, dla jakich został stworzony, nazwany przez Wooldridge'a w 1997 roku multiagentem [fis97], [woo02].
- Autonomiczny system znajdujący się w dynamicznym otoczeniu, działający niezależnie od narzucanych przezeń ograniczeń i realizujący w jego ramach zbiór celów lub poleceń, dla których został stworzony [mae98].

Wszystkie zasygnalizowane wyżej definicje agenta obejmuje ujęcie H. Van Dyke'a Parunaka z 1997 roku [par97].

System multiagentowy opisuje się następująco $\langle \text{Agenci}, \text{Środowisko}, \text{Sprzęganie} \rangle$, gdzie:

- **Środowisko** – posiada własne procesy zmieniające jego stan, ale jest niezależne od akcji wbudowanych w to środowisko agentów.
- **Sprzęganie** – odwzorowanie wejść agenta ze stanu środowiska i wyjść na stan środowiska.
- $\text{Agenci} = \{ \text{Agent}_i, \dots, \text{Agent}_n \}$.

Agent_i opisywany jest jako czwórka $\langle \text{Stan}_i, \text{Wejście}_i, \text{Wyjście}_i, \text{Proces}_i \rangle$, dla $i = 1, 2, \dots, n$, gdzie:

- **Stan_i** – zbiór własności (np. wartości liczbowe, twierdzenia logiczne), które całkowicie opisują agenta.
- **Wejście_i, Wyjście_i** – podzbiory zbioru **Stan_i**, których zmienne sprzęgają agenta ze środowiskiem.

- **Proces_i** – niezależny proces wykonawczy, który zmienia **Stan_i** agenta.

Ujęcie Parunaka korzysta z pojęć teorii automatów. Pozwala to rozważać agentów jako automaty działające w świecie realnym (w środowisku) w taki sposób, że kodują one lub interpretują wzajemnie wyróżnione języki akceptowane przez określone w tym środowisku gramatyki regularne. Agenci są więc translatorami i interpretatorami tych języków. Wykorzystując standardową terminologię konstrukcji kompilatorów, teorii automatów, języków i obliczeń [wai89], [hop05], można system multiagentów opisać następująco:

Niech dane będą dwie gramatyki regularne $G_1 = \langle V_1, T_1, P_1, S_1 \rangle$, $G_2 = \langle V_2, T_2, P_2, S_2 \rangle$, a języki L_1, L_2 są akceptowane odpowiednio przez gramatyki G_1, G_2 , tj. $L_1 \subseteq L(G_1)$ i $L_2 \subseteq L(G_2)$. Ponadto niech zbiory K i I będą zbiorami pewnych symboli zmiennych, zwanych odpowiednio **kodującymi** i **interpretacyjnymi**. Dodane są reguły: **kodowania** $k ::= x$, gdzie $\langle x, k \rangle \in K_d \subseteq (T_1 \cup T_2 \cup K)^* \times K$, oraz **interpretowania** $i ::= x$, gdzie $\langle x, i \rangle \in I_n \subseteq (T_1 \cup T_2 \cup I)^* \times I$, takie, że poniższe gramatyki są regularne:

$$G_K = \langle V_1 \cup V_2 \cup K, T_1 \cup T_2, P_1 \cup P_2 \cup K_d, S_1 \rangle \quad (7)$$

$$G_I = \langle V_1 \cup V_2 \cup I, T_1 \cup T_2, P_1 \cup P_2 \cup I_d, S_2 \rangle \quad (8)$$

Ponadto zachodzi **kodowanie**: każde słowo $u_1 \in L_1$, które można wywieść w gramatyce G_K , można również wywieść w G_K z pewnego słowa $u_2 S_1$, gdzie $u_2 \in L_2$, które można wywieść ze słowa S_1 w gramatyce G_K , a słowo u_2 można wywieść w gramatyce G_I ze słowa $u_1 S_2$, które można wywieść ze słowa S_2 w gramatyce G_I .

Także odwrotnie zachodzi **interpretacja**: każde słowo $u_2 \in L_2$, które można wywieść w gramatyce G_I , można również wywieść w G_I z pewnego słowa $u_1 S_2$, gdzie $u_1 \in L_1$, które można wywieść ze słowa S_2 w gramatyce G_I , a słowo u_1 można wywieść w gramatyce G_K ze słowa $u_2 S_1$, które można wywieść ze słowa S_1 w gramatyce G_K .

Gramatyki regularne G_K, G_I nazywane są odpowiednio **translatorem** i **interpreterem** języków L_1, L_2 akceptowanych przez gramatyki G_1, G_2 .

Rozważmy gramatykę $G_A(L_1, L_2, G_1, G_2)$, określoną wzorem:

$$G_A(L_1, L_2, G_1, G_2) = \langle V_A, T_A, P_A, S_A \rangle \quad (9)$$

gdzie:

- 1) $V_A = V_1 \cup V_2 \cup K \cup I \cup \{S_A\}$,
- 2) $T_A = T_1 \cup T_2$,
- 3) $P_A = P_1 \cup P_2 \cup K_d \cup I_n \cup \{S_A ::= S_1, S_A ::= S_2\}$.

Definicja (protokół) [fra97] 2.2.

Gramatykę regularną $G_A(L_1, L_2, G_1, G_2)$ utożsamiać można z **agentem** obsługującym języki L_1, L_2 , **wejściowy, wyjściowy**, akceptowane przez gramatyki G_1, G_2 . Agent obsługującego języki akceptowane przez innych agentów nazywa się **protokołem**.

Dowolna przeliczalna rodzina języków L_i , dla $i = 1, 2, \dots, n, \dots$, akceptowanych przez gramatyki regularne G_i , dla $i \in \mathbb{N}$, tworzy **sieć komunikowania się**, czyli **system**

multiagentowy, o strukturze połączeń określonej przez pary indeksów $\langle k, l \rangle \in I \subseteq \{\langle n_1, n_2 \rangle : n_1, n_2 \in \aleph\}$. Dzieje się tak, jeśli istnieje taki zbiór agentów $a_{kl} = G_A(L_k, L_l, G_k, G_l)$, dla $\langle k, l \rangle \in I$, że dowolny agent a_{kl} z tego zbioru obsługuje języki L_k, L_l , akceptowane przez gramatyki G_k, G_l . Ponadto agenci z tego zbioru agentów obsługują odpowiednio wszystkie języki ze zbioru $\{L_i\}$. W ujęciu Parunaka zbiór $\{L_i\}$ języków akceptowanych przez gramatyki regularne G_i jest środowiskiem systemu multiagentowego, a zbiór par $\langle L_k, L_l \rangle$ tych języków (dla agentów a_{kl} , języków wejściowych i wyjściowych), zgodnie z parami indeksów $\langle k, l \rangle$, jest relacją sprzęgania wejść z wyjściami tego systemu.

Obserwacja 2.5.

Zgodnie z teorią automatów, języków i obliczeń oraz zasadami konstruowania kompilatorów, kompilatory języków programowania można rozważać jako sieci komunikowania się [wai05].

Obserwacja 2.6.

Sieć internetowa Web jest siecią komunikowania się. W tej sieci kompilowane mogą być tzw. *języki znakowania*, takie jak HTML, XML, OWL.

Obserwacja 2.7.

Wszelkie urządzenia pomiarowe, eksperymentatorów, procesory sieci komputerowych, działające w środowisku pomiarowym, można rozważać jako agentów systemu komunikowania się / multiagentów, systemu zwanego wtedy systemem pomiarowym. Rozważając system diagnostyczny jako system pomiarowy, będzie się go też traktować jako system multiagentów [zół12].

Postulat (modelu świata rzeczywistego) 2.1.

Istnieje model świata rzeczywistego, zwany **modelem źródłowym**, obejmujący wszystkie możliwe do rozpoznania przez umysł człowieka obiekty i encje wskazujące te obiekty.

Obserwacja 2.8.

W modelu źródłowym w pełni określona jest, przez stosowne encje, struktura dowolnych sieci komunikowania się: języków, gramatyk, agentów.

Postulat (istnienia uniwersalnej sieci komunikowania się) 2.2.

W modelu źródłowym można określić **uniwersalną sieć komunikowania się**, zawierającą dowolną sieć komunikowania się.

Postulat (istnienia silnej inteligencji) 2.3.

Uniwersalna sieć komunikowania się zawiera agenta posługującego się wszystkimi językami tej sieci. Agent ten zwany jest **agentem silnej sztucznej inteligencji** (ang. *Artificial Intelligence* – AI) [fla11, s. 206-214, 242].

Obserwacja 2.9.

Nie wiadomo, czy istnieje test Turinga mogący sprawdzić, czy umysł człowieka jest równoważny agentowi silnej inteligencji. Wiadomo, że wszyscy agenci działający w sieciach komputerowych mogą być określani przez umysł człowieka. Ci agenci zwani są **agentami słabej AI** [fla11, s. 206-214, 242].

Postulat (człowieka jako agenta) 2.4.

O człowieku wcielającym w swoim umyśle strukturę agenta słabej AI będziemy mówić, że jest **człowiekiem agentem** [fla11, s. 206-214].

2.2.3. Modele wiedzy diagnostycznej w systemach danych diagnostycznych

Systemy diagnostyczne dokonują diagnozy, czyli przetwarzają informacje o stanach uszkodzeń, wykorzystując informacje o symptomach tych uszkodzeń. Stany i symptomy są zbiorami oznak/encji interpretowanymi przez agenta (w tym człowieka) jako oznaki tych stanów lub symptomów. Tak zinterpretowane zbiory encji są pewnymi danymi diagnostycznymi lub typami danych diagnostycznych. Na przykład daną o stanie uszkodzenia roweru, wyrażoną w języku polskim, jest „uszkodzenie łańcucha”, a daną o symptomie tego uszkodzenia jest „rozerwany łańcuch roweru”. Wymienione nazwy danych oznaczają zbiór wszelkich możliwych egzemplarzy tych nazw dla „uszkodzonych łańcuchów rowerów” i „rozerwanych łańcuchów rowerów”.

W świecie rzeczywistym każdy konkretny agent przetwarza pewne encje, wskazujące na zbiory zwane **danymi**. Zgodnie ze standardowymi modelami danych: atrybutowymi, czy relacyjnymi, dana jest wartością abstrakcyjną atrybutu [wai89, s. 33, 34] lub elementem dziedziny relacji (związków pomiędzy danymi) [ull88, s. 25, 32]. W tym kontekście, dla określonego wcześniej systemu multiagentów, dana może być rozważana jako zbiór słów-encji, które jakiś agent jednakowo interpretuje, tj. z których wyprowadza w interpreterze te same słowa akceptowane przez interpreter, a ponadto jednakowo koduje w translatorze te słowa za pomocą słów należących do tego zbioru (słów interpretowanych). Słowa interpretowane i ich interpretacje zwane są **egzemplarzami danej** (niekiedy *obiektami danej* [wai89, s. 33]). Zbiór wszystkich wyników interpretacji akceptowanych w interpreterze jest utożsamiany z pewnym **obiektem**, a jego elementy – z egzemplarzami danych o tym obiekcie, o ile każdy z egzemplarzy jest oznaką tego samego rzeczywistego obiektu. Ponieważ każdy egzemplarz obiektu jest wynikiem interpretacji, więc obiekt jest daną. Na przykład egzemplarzami danego obiektu – konkretnej maszyny – są jego konkretne nazwy, fotografie, obrazy, rysunki techniczne itp. Egzemplarze danych mogą być konkretnymi napisami, wypowiedziami wskazującymi w świecie na jakieś obiekty, ich cechy lub właściwości i jako takie mogą być oznakami/encjami tworzenia odpowiedniej struktury obiektów rzeczywistych, oznakami/encjami przedmiotów materialnych czy sygnałów konkretnego środka informacyjnego, np. sygnałów w komputerze. W tym sensie mówimy, że dana dotyczy pewnych obiektów. Jednakowe użycie przez agentów danych o obiektach, w dowolnej sieci

komunikowania się, prowadzi do wyznaczenia **wiedzy** agentów o tych obiektach. Informatycy mówią wtedy, że jeśli dana reprezentuje wiedzę o konkretnym obiekcie, to agent reprezentuje tę wiedzę za pomocą pewnego egzemplarza tej danej, określając ten obiekt za pomocą egzemplarza danej wskazującej na ten konkretny obiekt. Ponadto samego konkretnego agenta opisuje dana wskazująca na tego agenta. Zatem dla agentów danymi są „dany agent” i „dany obiekt”, a dany agent jest również pewnym obiektem. Można również zauważyć, że agenci i obiekty utożsamiane są z pewnymi danymi. Zbiory agentów i obiektów rozważane są jako zbiory danych [bry12c].

System danych (ang. *data system* – DS) można określić jako strukturę relacyjną składającą się z:

- zbioru egzemplarzy danych, krótko – egzemplarzy,
- zbioru danych,
- zbioru obiektów – zbioru danych o byciu obiektem,
- zbioru agentów – zbioru danych o byciu agentem,
- zbioru typów danych,
- relacji *reprezentowania wiedzy przez egzemplarze danych*: agent „*a*” reprezentuje wiedzę o obiekcie „*o*” za pomocą egzemplarza „*e*” zbioru obiektów wskazujących konkretne obiekty,
- relacji *przetwarzania danych danego typu*: tj. agent *u* przetwarza dane typu *e*.

W systemie danych *d* jest daną wtedy i tylko wtedy, gdy istnieje egzemplarz danej *e* taki, że *d* jest zbiorem egzemplarzy danych *c*, dla których istnieją agent *a*, obiekt *o* takie, że jednocześnie:

- agent *a* za pomocą *e* reprezentuje wiedzę o obiekcie *o*,
- agent *a* za pomocą *c* reprezentuje wiedzę o obiekcie *o*.

Wtedy daną tę nazywamy **daną o reprezentancie** *e* i oznaczamy przez [*e*] (dane oznaczone przez [*e*], gdy $e \in d$, nie muszą być identyczne).

Wszelkie **typy danych** składające się z danych są mnogościowymi sumami danych (stosownych zbiorów egzemplarzy danych). Elementy typów danych zwane są *reprezentantami tych typów*. Typy danych, poza agentami i obiektami, w ich opisach i opisach reprezentowania wiedzy przez egzemplarze danych, oznaczane są dalej małymi, pogrubionymi literami: ***a, d, e, o, t, ...***

O agencie *a* mówimy, że **przetwarza dane typu *e***, gdy istnieją egzemplarz danej $e \in e$ oraz obiekt *o* takie, że *a* za pomocą *e* reprezentuje wiedzę o obiekcie *o*. Tę relację nazywa się *relacją przetwarzania danych danego typu*.

Obiekty są danymi, a agenci są pewnymi obiektami. Ponadto obiekty i agenci są pewnymi typami danych.

Bardzo istotne jest jednak to, że **żadne przedmioty matematyczne nie są typami danych** (w tym danymi), np. liczby 0, 1, 2, ..., ich zbiory, operacje na liczbach itp. [abe02, s. 5]. Ogólnie żadne struktury materialne i abstrakcyjne nie są danymi, jeśli nie są zbiorami, jednakowo

interpretowanymi w procesie komunikowania się, oznakami/encjami pewnych struktur. Na przykład napis „5” może oznaczać strukturę równoliczności pięcioelementowych zbiorów. Daną [5] jest zbiór wszystkich słów używanych w komunikowaniu się w języku, będących oznaką/encją struktury oznaczanej przez napis „5”.

Obserwacja 2.10.

We wszelkich konkretnych interpretacjach systemu danych egzemplarze danych można interpretować jako *konkretne opisy* obiektów w wyróżnionym do tego języku naturalnym lub sztucznym. Ponadto typy danych interpretuje się za pomocą wybranych, wzorcowych egzemplarzy należących do tych typów.

W dalszym ciągu rozdziału będziemy rozważać interpretacje systemu danych zgodne z Obserwacją 2.10. Dowolny agent może używać w ten sam sposób (stosować te same procedury przetwarzania danych) różnych typów danych, np. wymienionych zgodnie z Obserwacją 2.10. typów danych:

- 1) $e = [\text{Anna} - \text{córka Izabeli}]$, $e' = [\text{Anna, której matką jest Izabela}]$,
- 2) $e = [\text{trójkąt, którego kąty są przystające}]$, $e' = [\text{trójkąt równoboczny}]$,
- 3) $e = \cup \{[1], [2], [3]\}$, $e' = \cup \{[\text{jeden}], [\text{dwa}], [\text{trzy}]\}$.

Ten sposób używania dwóch typów danych dla każdego agenta precyzuje definicja:

Dwa dowolne typy danych e, e' są **równoznaczne**, pisze się $e \equiv e'$, gdy dla dowolnego agenta a zachodzi warunek: a przetwarza dane typu e wtedy i tylko wtedy, gdy a przetwarza dane typu e' . Egzemplarze przetwarzanych typów danych reprezentują wiedzę o tych samych obiektach.

Typy danych są równoznaczne, gdy są przez dowolnego agenta równoważnie przetwarzane oraz ich egzemplarze reprezentują wiedzę o tych samych obiektach.

Obserwacja 2.11.

Warunek równoważności przetwarzania danych typów e, e' zachodzi dla grupy agentów nieprzetwarzających tych typów danych. W praktyce często taka sytuacja występuje, gdyż pewne zbiory typów danych wyrażonych w języku naturalnym są jednakowo przetwarzane tylko przez grupy ludzi znających ten język i komunikujących się w tym języku. Wtedy słuszne jest stwierdzenie, że **typy danych są równoznaczne dla rozważanej grupy agentów**.

Innymi słowy, posiadamy tę samą wiedzę o obiektach, gdy tak samo przetwarzamy typy danych reprezentujących wiedzę o tych samych obiektach. W tym sensie klasę wszystkich równoznacznych typów danych możemy utożsamiać z elementem wiedzy i tak w praktyce komunikacji językowej zazwyczaj się robi. Słuszne zatem będzie wprowadzenie następującej definicji:

Element wiedzy i wiedza

Klasę abstrakcji $[e]_{\equiv}$, relacji \equiv nazywa się **elementem wiedzy** reprezentowanym przez typ e danych, a dowolny podzbiór K zbioru $K = \{[e]_{\equiv} : e \text{ jest typem danych}\}$, wszystkich elementów wiedzy, nazywamy **wiedzą** w systemie danych.

O typie danych e takim, że element wiedzy $[e]_{\equiv}$ należy do wiedzy K , mówi się, że: e *reprezentuje wiedzę* K oraz że jest to *wiedza o obiektach*, którą reprezentują egzemplarze danych typu e (krótko: *wiedza o obiektach*).

Obserwacja 2.12.

W systemach komunikowania się informacja, rozumiana jako pewna wiadomość, jest kodowana za pomocą egzemplarza e znaku $[e]$. Zbiór k wszystkich równoznacznych znaków ze znakiem $[e]$ jest elementem wiedzy, którą reprezentuje ten znak. W tym kontekście wiedzę $K = \{k\}$ można utożsamiać z kodowaną przez e informacją (wiadomością). Co więcej, każde przetwarzanie informacji jest przetwarzaniem egzemplarzy danych, określonym w systemie komunikowania się (systemie multiagentów).

Obserwacja 2.13.

Gdy uwzględnia się Obserwację 2.11., w praktyce często występuje sytuacja, w której wiedza, w sensie jednakowego użycia typów danych, jest wyznaczana tylko w tej grupie agentów, którzy równoznacznie używają typów danych reprezentujących tę wiedzę. Dlatego słusznie jest mówić o wiedzy rozważanej przez grupy agentów. Różne grupy agentów mogą posiadać różną wiedzę o tych samych obiektach. Co więcej, może się zdarzyć, że jakiś agent będzie należał do dwóch grup agentów posiadających różną wiedzę o tym samym obiekcie, a więc komunikując się z tymi agentami, będzie się posługiwał egzemplarzami danych, raz reprezentujących wiedzę jednej grupy, a drugi raz egzemplarzami danych reprezentujących wiedzę drugiej grupy.

Obserwacja 2.14.

Systemem komunikowania się reprezentującym wiedzę w sensie opisanym w Obserwacji 2.11. i zarazem zintegrowanym komputerowym systemem pomiarowym jest *DiaSter – inteligentny system diagnostyki i wspomaganie sterowania procesami przemysłowymi*, zaprojektowany przez zespół specjalistów Politechniki Warszawskiej, Politechnik Śląskiej, Politechniki Rzeszowskiej oraz Uniwersytetu Zielonogórskiego [kor09]. W bazach danych tego systemu zawartych jest bardzo dużo danych o procesach, będących wartościami atrybutów (zmiennych i parametrów), używanych we wnioskowaniu diagnostycznym. Według twórców [kor09, s. 24]: „System wyposażony jest w mechanizmy odkrywania wiedzy mogącej znaleźć zastosowanie w rozumowaniu diagnostycznym. Przedmiotem odkrywania wiedzy przydatnej do budowy modeli dynamicznych mogą być szeroko rozumiane regularności w danych. Przez regularności rozumie się jakiś wzorzec oraz zakres jego »ważności«, określony poprzez podzbiór zbioru atrybutów i/lub rekordów”.

Wiedza diagnostyczna utożsamiana jest tu z pewnymi regularnościami w danych, co można opisać za pomocą elementów wiedzy składających się z danych równoznacznych ze względu na występujące w nich regularności.

W następnych podrozdziałach 2.2.4. – 2.2.6. bardziej szczegółowo zostaną opisane atrybutowe modele reprezentacji wiedzy: bazy danych i sieci semantyczne. W rozdziale 3 pracy podane zostaną pewne przykłady zastosowania tych modeli do opisu reprezentacji wiedzy diagnostycznej.

2.2.4. Logika atrybutowa

Kluczowym pojęciem w atrybutowym modelu reprezentacji wiedzy jest klasyfikacja obiektów. Przyjmujemy intuicyjnie, że klasyfikacja jest wiedzą, której poszczególne elementy wyznaczają rozłączne ze sobą zbiory obiektów [wai89], [lig10], [moc02].

Klasyfikacja

Klasyfikacją obiektów w systemie danych nazywamy wiedzę K taką, że zbiory $\{O(k): k \in K\}$ są rozłączne, a dla dowolnego $k \in K$ istnieje typ danych e taki, że $k = [e]_{\equiv}$ oraz istnieje agent u i egzemplarz danej e typu e takie, że u za pomocą e reprezentuje wiedzę o obiekcie o .

Niech wiedza K będzie klasyfikacją w systemie danych. Wtedy dla dowolnego obiektu $o \in \cup \{O(k): k \in K\}$ istnieje dokładnie jeden element wiedzy $k \in K$ taki, że $o \in O(k)$.

Stąd otrzymujemy, że dla klasyfikacji K zbiór danych D o obiektach ze zbioru $O(k)$ jest zbiorem poszczególnych reprezentantów elementów wiedzy K . Wtedy dla dowolnego obiektu $o \in \cup \{O(k): k \in K\}$ istnieje dokładnie jeden element wiedzy $d \in D$ taki, że $o \in O(k)$, dla $k = [d]_{\equiv}$.

Atrybuty

Powyższe stwierdzenie, zgodnie ze standardową definicją atrybutu obiektów, podawaną w literaturze informatycznej [wai89, s. 34], [lig10, s. 263], określa funkcję będącą atrybutem. Z tego powodu w systemie danych przyjmujemy następującą definicję atrybutu:

Atrybutem w systemie danych jest dowolna funkcja częściowa $A: \text{Obiekty} \rightarrow D \subseteq \text{Dane}$, taka, że każda dana $t \in D$ wyznacza inny element wiedzy $k = [t]_{\equiv}$, a przeciwobraz $A^{-1}(\{t\}) = O(k)$.

Teraz można podać istotne definicje wprowadzające logikę atrybutową:

Niech A_1, A_2, \dots, A_n oznaczają pewne atrybuty w systemie danych DS , a D_1, D_2, \dots, D_n dziedziny tych atrybutów, tj. zbiory ich wartości, o oznacza obiekt, a d daną. Wtedy **językiem atrybutów** nazywamy zbiór formuł określonych następująco:

- Wyrażenie postaci „ $A_i(o) = d$ ” jest **formułą atomową**, dla $i = 1, 2, \dots, n$, a gdy obiekt o jest wyspecyfikowany (zadany) dla wartości d , to zapis można skrócić do: $A_i = d$; mówimy, że obiekt o jest wyspecyfikowany dla danej d , jeśli dana d reprezentuje wiedzę o tym obiekcie, tj. istnieje agent a i egzemplarz $e \in d$ takie, że a za pomocą e reprezentuje wiedzę o .

- Jeśli Φ i Ψ są formułami, to formułami są koniunkcja $\Phi \wedge \Psi$ oraz alternatywa $\Phi \vee \Psi$ tych formuł.
- **Selektorem** [lig10, s. 264] $[A_i = d]$ formuły atomowej $A_i = d$ nazywamy zbiór wszystkich obiektów o wyspecyfikowanych dla danej d takich, że $A_i(o) = d$.
- **Konstruktorem** [abe02, s. 83] formuły atomowej $A_i = d$ nazywamy klasyfikację wyznaczającą atrybut A_i .

Przykład 2.1.

Atrybut „bycie członkiem rodziny” ma następujące przykładowe wartości: rodzic i potomek. Opisywanymi obiektami są: ojciec Roman, matka Paulina, syn Jan, córka Anna. W tej interpretacji formuła atomowa „ $A = d$ ” jest schematem wyrażen: „ojciec Roman jako członek rodziny jest rodzicem”, „matka Paulina jako członek rodziny jest rodzicem”, „syn Jan jako członek rodziny jest potomkiem”, „córka Anna jako członek rodziny jest potomkiem”. Specyfikacją danej „rodzic” są osoby: ojciec Roman, matka Paulina, a specyfikacją danej „potomek” są osoby: syn Jan, córka Anna. Konstruktor formuły jest klasyfikacja rodziny na rodziców, synów i córki. Selektor „selekcjonuje” członków rodziny na tych, na których wskazuje dana w formule atomowej, np. wskazanie jest specyfikacją danej „rodzic”. Natomiast formuła atomowa postaci „ $A = d$ ” jest schematem wyrażen: „rodzic jako członek rodziny”, „potomek jako członek rodziny” [lig10].

Formuła atomowa $A_i = d (i = 1, 2, \dots, n)$ języka atrybutowego AL jest spełniona w systemie danych DS, jeśli dla każdego o oznaczającego obiekt wyspecyfikowany dla d wartość atrybutu $A_i(o)$ jest daną oznaczaną przez d . Jeśli Φ i Ψ są formułami, to koniunkcja $\Phi \wedge \Psi$ jest spełniona w DS, gdy obie formuły są spełnione, oraz alternatywa $\Phi \vee \Psi$ jest spełniona, gdy jedna z tych formuł jest spełniona.

Reguły wnioskowania i tablice decyzyjne [lig10, s. 267]

Niech $A_1, A_2, \dots, A_n, H_1, H_2, \dots, H_m$ oznaczają pewne atrybuty w systemie danych DS, a $d_1, d_2, \dots, d_n, h_1, h_2, \dots, h_m$ oznaczają dane. Schemat **reguły wnioskowania** w logice atrybutowej ma postać:

$$r: (A_1 = d_1) \wedge (A_2 = d_2) \wedge \dots \wedge (A_n = d_n) / \\ / (H_1 = h_1) \wedge (H_2 = h_2) \wedge \dots \wedge (H_m = h_m) \quad (10)$$

Tablicą decyzyjną nazywamy dowolny ciąg reguł wnioskowania $r_i, i = 1, 2, \dots, k$ o schematach:

$$r_i: (A_1 = d_{i1}) \wedge (A_2 = d_{i2}) \wedge \dots \wedge (A_n = d_{in}) / \\ / (H_1 = h_{i1}) \wedge (H_2 = h_{i2}) \wedge \dots \wedge (H_m = h_{im}) \quad (11)$$

Taki schemat tablicy decyzyjnej został przedstawiony w tabeli 2.1. Atrybuty A_i nazywamy *warunkowymi*, a atrybuty H_i – *decyzyjnymi*.

Tabela 2.1. Schemat tablicy decyzyjnej [lig10].

Reguła	$A_1, A_2, \dots, A_i, \dots, A_n$	$H_1, H_2, \dots, H_i, \dots, H_m$
r_1	$d_{11}, d_{12}, \dots, d_{1i}, \dots, d_{1n}$	$h_{11}, h_{12}, \dots, h_{1i}, \dots, h_{1m}$
r_2	$d_{21}, d_{22}, \dots, d_{2i}, \dots, d_{2n}$	$h_{21}, h_{22}, \dots, h_{2i}, \dots, h_{2m}$
\vdots	\vdots	\vdots
r_l	$d_{l1}, d_{l2}, \dots, d_{li}, \dots, d_{ln}$	$h_{l1}, h_{l2}, \dots, h_{li}, \dots, h_{lm}$
\vdots	\vdots	\vdots
r_k	$d_{k1}, d_{k2}, \dots, d_{ki}, \dots, d_{kn}$	$h_{k1}, h_{k2}, \dots, h_{ki}, \dots, h_{km}$

Można teraz wprowadzić atrybuty wieloargumentowe oraz tablice decyzyjne dla tych atrybutów.

2.2.5. Wieloargumentowe atrybuty i tablice decyzyjne dla tych atrybutów

O konkretnych obiektach bardzo często wiemy, że ustalają związki (zależności) pomiędzy innymi obiektami, np. więzi społeczne, więzi i połączenia produktów technicznych, związki chemiczne, zwroty tworzące złożone wyrażenia języka (np. przymiotniki, zaimki), w tym zwroty tworzące zdania złożone (np. czasowniki i spójniki zadaniowe), ogólnie związki (zależności) umożliwiające ustalenie wszelkich relacji. Dane wskazujące na wystąpienia tych związków są **krotkami** związanych obiektów: parami, trójkami, czwórkami, ogólnie n -tkami. Teoriomnogościowo są one opisywane jako uporządkowane układy elementów: $\langle x_1, x_2, \dots, x_n \rangle$. W językach programowania krotki obiektów zapisywane są jako kwerendy będące wieloargumentowymi termami, jako tablice czy listy, czy jako wieloargumentowe predykaty. Z tego powodu w formalnym opisie systemu danych DS słusznie jest wyróżnić zbiór obiektów O niebędących krotkami, natomiast zbiór $Object$, do którego należą obiekty będące krotkami obiektów o krotności nieprzekraczającej liczby n , określić jako odpowiednią sumę zbiorów [lig10]:

$$Object = O \cup O^2 \cup \dots \cup O^n \quad (12)$$

gdzie $O^k, k = 2, 3, \dots, n$, jest zbiorem wszystkich k -elementowych krotek $\langle o_1, o_2, \dots, o_k \rangle, o_i \in O, i = 1, 2, \dots, k$. Przyjęte założenie pozwala rozważać wieloargumentowe atrybuty i odpowiednie dla tych atrybutów tablice decyzyjne.

Atrybut k -argumentowy

Atrybutem k -argumentowym w systemie danych DS jest dowolny argument $A: Object \rightarrow D \subseteq Data$ określony dla zbioru $O^k \subseteq Object$ krotek obiektów.

Definicja (tablice decyzyjne wieloargumentowych atrybutów) 2.3.

Tablicę decyzyjną, w której występują wieloargumentowe atrybuty warunkowe lub decyzyjne, nazywamy **tablicą decyzyjną wieloargumentowych atrybutów** [lig10].

Przykład 2.2.

Nazwa „więzi rodzinne pomiędzy osobami” jest nazwą atrybutu dwuargumentowego, który parze osób przyporządkowuje daną: matka, ojciec, córka, syn, brat, siostra itp., tak że pierwsza osoba jest matką drugiej osoby, pierwsza osoba jest ojcem drugiej osoby, pierwsza osoba jest córką drugiej osoby, pierwsza osoba jest synem drugiej osoby, pierwsza osoba jest bratem drugiej osoby, pierwsza osoba jest siostrą drugiej osoby itp. Przy takiej interpretacji formuły atomowej postaci „ $A = d$ ” ta formuła jest schematem zdania „wszystkie pary osób, w których pierwsza osoba jest w więzi rodzinnej z drugą osobą”. Selektorem tej formuły jest zbiór par osób, w których pierwsza osoba jest w więzi rodzinnej reprezentowanej przez tę daną z drugą osobą, np. zbiór wszystkich par osób, w których pierwsza osoba jest w więzi rodzinnej bycia matką drugiej osoby.

Przykład 2.3.

Nazwa „pomiędzy” jest nazwą atrybutu trójargumentowego, który trójce osób przyporządkowuje przykładowo dane: siedzi, leży, stoi itp., tak że pierwsza osoba siedzi pomiędzy drugą i trzecią osobą, pierwsza osoba leży pomiędzy drugą i trzecią osobą, pierwsza osoba stoi pomiędzy drugą i trzecią osobą itp.

Zauważmy, że z teoriomnogościowej definicji n -tki, więcej niż dwuelementowej, wynika, że można tę n -tkę wyrazić za pomocą kolejno występujących w niej par elementów.

Obserwacja 2.15.

Każdą formułę atomową zawierającą atrybut o więcej niż dwóch argumentach można wyrazić za pomocą koniunkcji formuł atomowych z atrybutami dwuargumentowymi.

Przykład 2.4.

Niech formuła „ $A = d$ ” będzie schematem wyrażenia „pierwsza osoba siedzi pomiędzy drugą i trzecią osobą”. „ A ” oznacza atrybut trójargumentowy. Podane wyrażenie można równoznacznie opisać przez koniunkcję „pierwsza osoba siedzi za drugą osobą i pierwsza osoba siedzi przed trzecią osobą”. Wyrażenie to jest zgodne ze schematem: $A_1 = d_1 \wedge A_2 = d_2$ gdzie A_1 i A_2 oznaczają dwuargumentowe atrybuty „za”, „przed” o wartości d_1 i d_2 interpretowane są jako słowo „siedzi”.

2.2.6. Sieć semantyczna wyznaczona przez język atrybutów wieloargumentowych

Poniżej przedstawiono definicję sieci semantycznej w systemie danych oraz podstawowych elementów tej sieci [bry12c].

Przyjmijmy, zgodnie z Obserwacją 2.15., że będziemy rozważać tylko atrybuty jedno- i dwuargumentowe. Ponadto zauważmy, że:

Obserwacja 2.16.

Każdy **typ danych** w systemie danych, reprezentujący wiedzę o budowie napisów, którego wszystkie egzemplarze danych są równokształtnymi napisami (identyfikowanymi jako napisy według tych samych procedur), zawiera się zarazem w każdej danej reprezentującej wiedzę o obiektach niebędących napisami. Może się zdarzyć, że ten sam napis zawiera się w dwóch różnych danych.

Określone w Obserwacji 2.16. typy danych będą dalej nazywane **opisami** i oznaczane: dużymi literami R, R_1, R_2, \dots , gdy ich egzemplarze są reprezentantami danych o parach obiektów, a przez C, C_1, C_2, \dots , gdy ich egzemplarze są reprezentantami danych obiektów niebędących krotkami obiektów.

Obserwacja 2.17.

W praktyce do zapisywania danych stosuje się tylko ich egzemplarze będące egzemplarzami opisów używanych przez większość agentów. Na przykład nie używa się egzemplarza znaku, za pomocą którego wiedzę o pewnym obiekcie reprezentuje tylko jeden agent. Zgodnie z Obserwacją 2.16. egzemplarz opisu może być egzemplarzem dwóch różnych danych. Taki stan rzeczy prowadzi nieraz do **wieloznaczności** w reprezentowaniu wiedzy o obiektach przez egzemplarze danych. Dlatego różne grupy agentów mogą coś innego wiedzieć o tych samych obiektach.

Koncept, rola, ich wystąpienia oraz asercje

Daną d , dla pewnego języka atrybutów, nazywamy **konceptem** o opisie C lub krótko konceptem C , jeśli egzemplarze konceptu C są egzemplarzami danej d , a ta dana jest wartością pewnego jednoargumentowego atrybutu A .

Daną d będącą wartością pewnego dwuargumentowego atrybutu A nazywamy **rolą** o opisie R , jeśli egzemplarze opisu R są egzemplarzami danej d będącej wartością atrybutu dwuargumentowego.

Wystąpieniem konceptu C nazywamy każdy obiekt $o \in [A = d]$ (obiekt należący do selektora formuły atomowej $A = d$, tj. spełniający $A(o) = d$). **Wystąpieniem roli** R nazywamy każdą parę $\langle o_1, o_2 \rangle \in [A = d]$. Wyrażenia postaci $o: C$ i $\langle o_1, o_2 \rangle: R$ stwierdzające, że o jest wystąpieniem konceptu C , a para $\langle o_1, o_2 \rangle$ jest wystąpieniem roli R , nazywamy **asercjami**.

Przykład 2.5.

Niech opis „ojciec” będzie równoznaczny „ojcu jako członkowi rodziny”. Dana „ojciec” będąca wartością atrybutu „członek rodziny” jest konceptem „ojca”. Wystąpieniami tego konceptu są dane wyrażające osoby będące ojcami jako członkami rodziny. Asercją wyrażającą to, że Jan jest ojcem jako członek rodziny, piszemy „Jan: ojciec”.

Przykład 2.6.

Niech opis „ojciec” będzie równoznaczny danej „ojciec” atrybutu „więzi rodzinne”, tj. jest równoznaczny temu, że ktoś jest ojcem kogoś. Wtedy dana „ojciec” jest rolą. Wystąpieniami tej roli są wszystkie pary osób, w których pierwsza osoba jest ojcem drugiej. Asercję wyrażającą to, że Jan jest ojcem Anny, piszemy: „(Jan, Anna): ojciec”.

Obserwacja 2.18.

Asercje postaci $o : C$ wyrażają posiadanie przez obiekty cech opisanych jako koncept C .

Obserwacja 2.19.

Asercje postaci $(o_1, o_2) : R$ wyrażają posiadanie przez pary obiektów własności, opisane jako rola R .

Do sieci semantycznych możemy zaliczyć:

- **Sieci sensorowe** mierzące dane dotyczące miejsca (np. sieci przyrządów pomiarowych składających się na system pomiarowy, w tym system diagnostyczny), w którym znajdują się sensory, i przekazujące zebrane dane do jednostek sterujących.
- **Sieci autopilotów maszyn** (np. robotów lub pojazdów) wyszukujące dane o środowisku (np. dane o linii technologicznej lub komunikacyjnej), częściowo sterujące pracą tych maszyn (przekazywanie informacji lub kierowanie ruchem pojazdów).
- **Sieci semantyczne Web z poszerzoną rzeczywistością** (np. o dane meteorologiczne i mapy dróg), korzystające z danych zebranych w wymienionych sieciach.

Nawiązując do koncepcji sieci semantycznych zaprezentowanej w pracach [kow79b], [mag78], jako grafu indeksowanego skierowanego, reprezentującego cechy i własności obiektów, możemy wprowadzić następującą definicję sieci semantycznej w systemie DS:

Sieć semantyczna w systemie danych [kow79b]

Siecią semantyczną w systemie danych [kow79b] nazywamy dowolny skierowany, indeksowany graf, w którym:

- Jeżeli C jest opisem krawędzi pętli, a obiekt o opisem o jest wierzchołkiem (węzłem), to C jest opisem konceptu, a $o : C$ jest asercją; opis asercji o postaci $o : C$ (lub inny równoznaczny mu opis) jest zarazem opisem roli „obiekt o jest elementem konceptu C ”.
- Jeżeli R jest opisem krawędzi (strzałki), a kolejnymi wierzchołkami krawędzi są obiekty o_1, o_2 , to R jest opisem roli, a $(o_1, o_2) : R$ jest asercją; opis asercji o postaci $(o_1, o_2) : R$ (lub inny równoznaczny jemu opis) jest zarazem opisem roli „para (o_1, o_2) jest w roli R ”.

O zbiorze klasyfikacji wyznaczających język atrybutów, dla którego określone są koncepty i role występujące w sieci semantycznej, mówimy, że **wyznacza tę sieć semantyczną**.

Powyższa definicja oraz definicja wiedzy prowadzą do następującego ogólnego spostrzeżenia. W dowolnej sieci semantycznej koncepty i role odpowiadają danym o obiektach

opisanych w wierzchołkach sieci. Te dane wyznaczają elementy wiedzy, które składają się na wiedzę reprezentowaną przez sieć semantyczną.

2.2.7. Reprezentacja wiedzy w sieci semantycznej Web

W 1999 roku w ramach konsorcjum W3C opracowano standard RDF (ang. *Resource Description Framework*), opisany na stronach W3C [rdf04], [man04], a jego model w pracy [las03].

RDF służy do opisywania danych – zasobów internetowych. **Zasobem** jest każda encja (jako oznaka, opis obiektów) będąca składnikiem systemu danych, posiadającym dobrze określoną strukturę reprezentacji wiedzy: strona internetowa i jej elementy, dokument, obraz, film, dokument dźwiękowy itp., udostępniany w Internecie za pomocą adresu. W tym sensie zasób jest agregacją encji w sieci internetowej (sieci semantycznej Web), zbudowaną z innych encji zgodnie ze strukturą reprezentacji wiedzy określoną przez gramatykę języka RDF. **Adres** jest opisem zasobu w języku URI (ang. *Uniform Resource Identifier*). Język URI posiada uogólnioną składnię przeznaczoną do opisywania zasobów w sieci. Opis URI jest zbudowany z nazwy schematu i części specyficznej dla schematu. Opis URI stanowi ogólny sposób adresowania. Jednym z rodzajów opisów URI są adresy w języku URL. Skrót URL (ang. *Uniform Resource Locator*) – **adres internetowy** (lokalizacja) **strony**, pliku lub innego zasobu w Internecie. Ogólna składnia URL jest następująca: **protokół://adres serwera/adres zasobu**

Oto kilka przykładów adresów URL:

- <http://www.prezydent.pl/>
- <http://www.radio.com.pl/trojka/index.html#test>
- <https://ssl.onet.pl/login>
- <ftp://po.opole.pl/programy/ssh.exe>
- <ftp://ania.haslo@po.opole.pl/programy/ssh.exe>
- <news://news.vogel.pl/pl.comp.www>
- <http://www.po.opole.pl/index2.php?mod=start>

Przykładowe bardziej uogólnione formy adresu:

protokół // komputer.domena / procedura ? p1=w1#p2=w2 ...

protokół://uzytkownik:haslo@host:nr_portu/ścieżka:parametr_obiektu?zapytanie#etykieta

Standard RDF oferuje zestaw środków do modelowania danych (zasobów) oraz język znakowania (znaczników) do wyrażania modeli. Model danych w rozważanym standardzie składa się ze zbioru stwierdzeń (ang. *statements*), które opisują wartości (ang. *object*) przypisywane cechom, właściwościom (atrybutom, związkom – ang. *properties*) opisywanych zasobów (ang. *resources*). Wartościami mogą być literały znakowe (ang. *literals*) lub zasoby. Dopuszcza się także możliwość tworzenia stwierdzeń „wyższych rzędów” (stwierdzenia o stwierdzeniach) za pośrednictwem mechanizmu reifikacji (ang. *reification*) – procesu, w którym abstrakcyjne wyobrażenie o programie komputerowym, napisane np. w języku RDF, XML czy OWL, jest włączone do wyraźnego **modelu danych** lub innego obiektu stworzonego w **języku programowania** sieci Web (np. HTML). Wtedy poddane reifikacji stwierdzenie staje

się zasobem tej sieci. Standard RDF poszerzono o system schematów RDFS (skr. ang. *Resource Description Framework Schema*) [bri04] zawierających środki do definiowania pojęć, relacji i typów danych.

Reprezentację wiedzy w sieciach semantycznych Web można sformułować w języku znakowania RDF – jest to model opisu *zasobów*. Wtedy zasób jest wszystkim, co można zidentyfikować poprzez napisy w języku adresów URI. Otrzymuje się dokumenty RDF – opisy zasobów i związków pomiędzy nimi, także schematy RDF – klasy zasobów, typy związków, rodzaje ograniczeń. Model RDF jest zbiorem stwierdzeń, utożsamianym z siecią semantyczną (ang. *Semantic Web*). Stwierdzenia te mają schemat trójek zawierających: wierzchołek, węzeł sieci, krawędź (strzałka), wierzchołek, węzeł sieci (rysunek 2.9.).



Rysunek 2.9. Powiązanie podmiotu z przedmiotem w stwierdzeniu z sieci semantycznej.
Opracowanie własne.

Stwierdzenia zapisane za pomocą trójek można interpretować w trzech znaczeniach:
podmiot – predykat – przedmiot lub
podmiot – orzeczenie – dopełnienie lub
obiekt – właściwość – wartość.

Na przykład:

Jan | jest_ojcem | Anny

Jan | mówi_że | (Jerzy | jest_ojcem | Anny)

Każdy element trójki jest zasobem identyfikowanym przez URI. W celu uniknięcia konfliktów nazw wprowadza się pojęcie **przestrzeni nazw** (ang. *namespace*). Istnieją przestrzenie nazw standardowych dla RDF: `rdfs`, `rdf`, definiujące podstawowe klasy RDF:

- `rdfs:Resource` – klasa zawierająca wszystkie zasoby,
- `rdfs:Class` – klasa zawierająca wszystkie klasy i ich instancje,
- `rdf:Property` – klasa zawierająca wszystkie właściwości,
- `rdf:Restriction` – klasa zawierająca wszystkie ograniczenia.

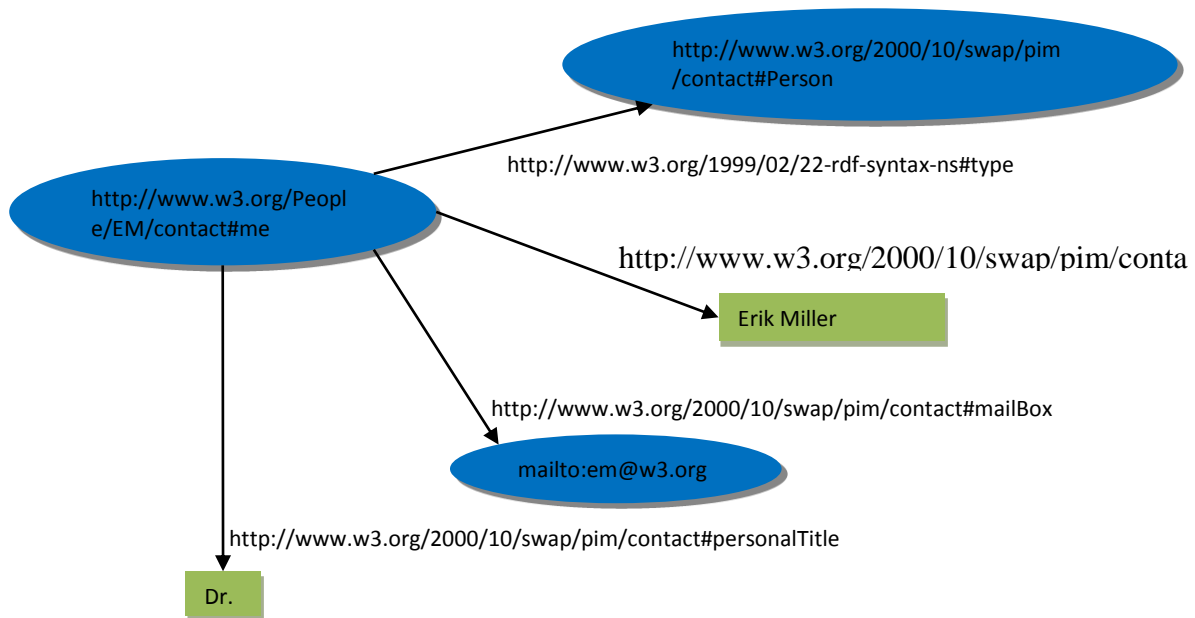
Inne przestrzenie nazw to na przykład Dublin Core (www.dublincore.org).

Przykład 2.7

Rysunek 2.10. pokazuje, jak język RDF, zastosowany do identyfikacji (reprezentacji) zasobów, wykorzystuje graf składający się z trójek opisanych przez URI. Na rysunku:

- indywidua, tj. Erik Miller, identyfikowane są przez nazwę/adres <http://www.w3.org/People/EM/contact#me>,
- rodzaje rzeczy, np. osoba (Person), identyfikowane są przez nazwę/adres <http://www.w3.org/2000/10/swap/pim/contact#Person>,

- własności tych obiektów, np. poczta (mail), identyfikowane są przez `http://www.w3.org/2000/10/swap/pim/contact#mailbox`,
- wartości tych właściwości, np. `mailto:em@w3.org` jako stałe wartości ustalające własności skrzynki pocztowej (RDF wykorzystuje również ciągi znaków, takich jak „Erik Miller”, a wartości z innych typów danych, takich jak liczby i daty, jak wartości ustalające własności).



Rysunek 2.10. Język RDF zastosowany do identyfikacji zasobów. Źródło [man04].

Identyfikacja zasobów reprezentowanych w sieci semantycznej na rysunku 2.10. może być zapisana w języku RDF za pomocą znaczników postaci:

<nazwa znacznika oraz atrybuty> opis identyfikatorów zasobu </nazwa znacznika>.

Przykład takiego zapisu dla danych osoby Erik Miller:

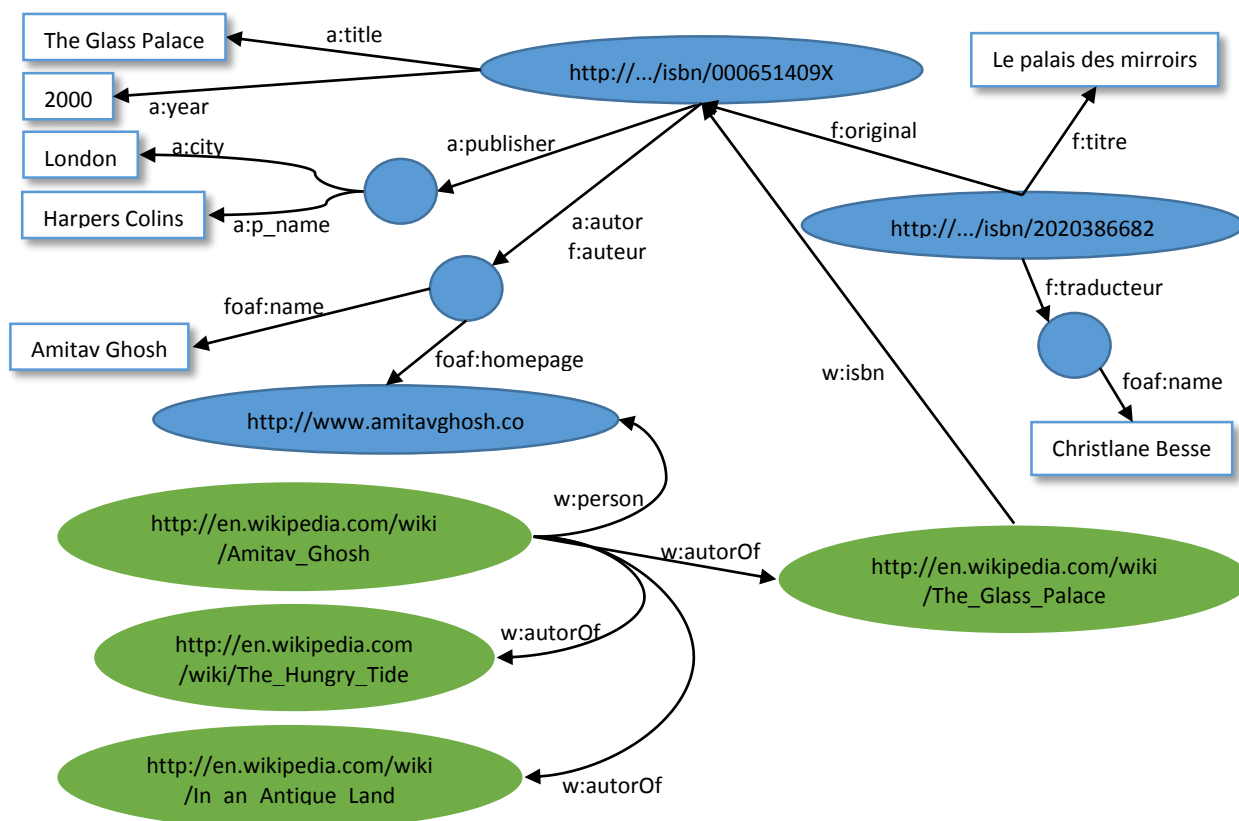
```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:contact="http://www.w3.org/2000/10/swap/pim/contact#">
<contact:Person rdf:about="http://www.w3.org/People/EM/contact#me">
<contact:fullName>Erik Miller</contact:fullName>
<contact:mailbox rdf:resource="mailto:em@w3.org"/>
<contact:personalTitle>Dr.</contact:personalTitle>
</contact:Person>
</rdf:RDF>
```

Przykład 2.8.

Rysunek 2.11. przedstawia dostęp do zasobów w sieci semantycznej Web. Zasoby są reprezentowane przez dane bibliograficzne. Klasyfikacja zasobów odbywa się na trzech poziomach:

- 1) stron internetowych (w wierzchołkach grafu są owale z adresami stron),

- 2) zbieranych danych (w wierzchołkach grafu są puste owale, tzw. wierzchołki *anonimowe*) oraz dodatkowo szukanych typów danych lub adresów stron (w wierzchołkach grafu są puste owale, tzw. wierzchołki *kontenery*),
- 3) konkretnych danych (w wierzchołkach grafu są prostokąty z opisami danych).



Rysunek 2.11. Dostęp do zasobów w sieci semantycznej Web. Źródło [w3c14].

Reprezentowane w wierzchołkach anonimowych (kontenerach) zasoby wiedzy są określone przez role (związki z innymi zasobami), w które wchodzi zbierane dane, np. pewien nieznan zasób dobrze określony jest przez poszukiwanie wydawcy książki (a:publisher), który określony jest przez jego nazwę (a:p_name) i miejsce wydania książki (a:city). Nazwy ról są postaci $\langle \text{wystąpienie atrybutu} \rangle : \langle \text{atrybut} \rangle$. Wystąpieniem atrybutu *publisher* jest para nazw atrybutów *p_name*, *city*, a wystąpieniami tych atrybutów są odpowiednio: *HarpersCollins*, *London*.

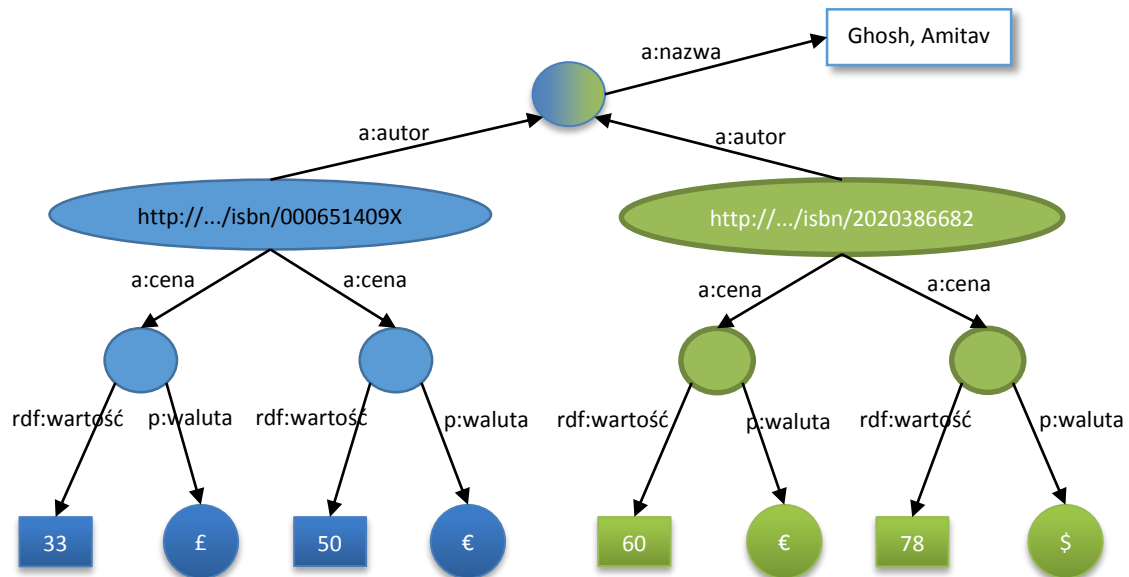
Przykład 2.9.

Do wyszukiwania informacji w sieci semantycznej określonej przez RDF stosuje się język **SPARQL** (ang. *Protocol and RDF Query Language*) zapytań dla RDF (rysunek 2.12.).

Zapytania odnoszące się do sieci semantycznej na rysunku 2.12. dotyczą (SELECT) książek o podanych numerach *isbn*, ich *cen* oraz *waluty*, gdzie (WHERE) dla tych książek szukana (?x)

jest dla roli *a:cena*, w wierzchołkach anonimowych, cena określona rolą *p:wartość* i waluta określona rolą *p:waluta*.

```
SELECT ?isbn ?cena ?waluta # note: not ?x!
WHERE {?isbn a:cena ?x. ?x rdf:wartość ?cena. ?x p:waluta ?waluta.}
```



Rysunek 2.12. Język zapytań SPARQL dla RDF. Źródło [w3c14].

Poniżej podane są inne przykładowe kody zapytań:

- Kod wyszukujący w bazie (SELECT) tytuł, który jest (WHERE) tytułem książki autora o nazwisku „Kurose”:

```
SELECT ?givenTitle
WHERE {?y, <http://www.helion.pl/books-rdf/3.0#Autor>, "Kurose"},
{?y, <http://www.helion.pl/books-rdf/3.0#Title>, ?givenTitle}
```

- Kod szukania (SELECT) tytułu książki zgodnie z bibliografią, do której jest na stronie (PREFIX) hiperłącze (WHERE):

```
PREFIX dc: <http://purl.org/dc/elements/1.1/>
SELECT ?title
WHERE { <http://example.org/book/book1> dc:title?title}
```

Jak widać, język SPARQL jest językiem nie tylko zapytań, ale i protokołów (patrz Definicja 2.2.) dla zasobów opisanych w RDF. Pozwala wyciągać z nich dane zawężone według kryteriów określonych poprzez predykaty RDF. Jest opisany przez kilka specyfikacji W3C [w3c14]:

- SPARQL protokół dla RDF,
- SPARQL język zapytań dla RDF,
- SPARQL wyniki zapytania w formacie XML.

Wiedza reprezentowana w sieciach semantycznych Web składa się na ontologię i tezaurus.

2.2.8. Ontologia i tezaurus

Zgodnie z ujęciem W. J. Rozenberga [roz82, s. 45, 46] dane pomiarowe o badanych obiektach w dziedzinach nauki i techniki reprezentują wiedzę o procesach lub systemach, np. urządzeniach technicznych, wytworzonych dla osiągnięcia określonych celów. Jej opis składa się z opisu łańcucha $X - M - E$, gdzie X jest zbiorem badanych obiektów, M jest tezauresem – zbiorem obiektów abstrakcyjnych, zbiorem modeli matematycznych lub formalnologicznych poszczególnych badanych obiektów, a E – zbiorem etalonów, zbiorem modeli fizycznych zachowujących się jak modele abstrakcyjne.

Tezaurus tworzy słownik wzorów, formuł, stałych i reguł ich przekształcania, za pomocą którego można wyprowadzić ilościowe sądy o realnych obiektach. Kompletność tezausa jest określona stanem zaawansowania teorii i potrzebami praktycznymi. Przyjmuje się, że tezaurus posiada organizację hierarchii klasyfikacji zbioru obiektów X równoważną klasyfikacji zbioru M abstrakcyjnych modeli tych obiektów, a opis klas tej hierarchii w języku zastosowanej w pomiarach dziedziny nauki lub techniki może mieć strukturę o następującej postaci [roz82, s. 55-60]:

1. Opis klasy M_1 ;
 - 1.1. Opis klasy M_1^1 ;
 - 1.2. Opis klasy M_2^1 ;
 - 1.2.1. Opis klasy $M_1^{1,2}$;
 - 1.2.2. Opis klasy $M_2^{1,2}$;
 - 1.2.3. Opis klasy $M_3^{1,2}$;
 - 1.2.3.1. Opis klasy $M_1^{1,2,3}$;
 - 1.2.3.2. Opis klasy $M_2^{1,2,3}$;
2. Opis klasy M_2 ;
 - 2.1. Opis klasy M_1^2 ;
- ...
3. Opis klasy M_3 ;
 - 3.1. Opis klasy M_1^3 ;
 - 3.2. Opis klasy M_2^3 ;
 - 3.2.1. Opis klasy $M_1^{3,2}$;
 - 3.2.2. Opis klasy $M_2^{3,2}$;
 - 3.2.3. Opis klasy $M_3^{3,2}$;
 - 3.2.3.1. Opis klasy $M_1^{3,2,3}$;
 - 3.2.3.2. Opis klasy $M_2^{3,2,3}$;
- ...
- 3.3. ...

Relację hierarchii można wtedy opisać w sposób analogiczny:

1. Opis tego, że klasa M_1 jest podklasą klasy M ;
 - 1.1. Opis tego, że klasa M_1^1 jest podklasą klasy M_1 ;

- 1.2. Opis tego, że klasa M_2^1 jest podklasą klasy M_1 ;
 - 1.2.1. Opis tego, że klasa $M_1^{1.2}$ jest podklasą klasy M_2^1 ;
 - 1.2.2. Opis tego, że klasa $M_2^{1.2}$ jest podklasą klasy M_2^1 ;
 - 1.2.3. Opis tego, że klasa $M_3^{1.2}$ jest podklasą klasy M_2^1 ;
 - 1.2.3.1. Opis tego, że klasa $M_1^{1.2.3}$ jest podklasą klasy $M_3^{1.2}$;
 - 1.2.3.2. Opis tego, że klasa $M_2^{1.2.3}$ jest podklasą klasy $M_3^{1.2}$;
- 2. Opis tego, że klasa M_2 jest podklasą klasy M ;
 - 2.1. Opis tego, że klasa M_1^2 jest podklasą klasy M_2 ;
- ...
- 3. Opis tego, że klasa M_3 jest podklasą klasy M ;
 - 3.1. Opis tego, że klasa M_1^3 jest podklasą klasy M_3 ;
 - 3.1.1. Opis tego, że klasa $M_1^{3.2}$ jest podklasą klasy M_1^3 ;
 - 3.1.2. Opis tego, że klasa $M_2^{3.2}$ jest podklasą klasy M_1^3 ;
 - 3.1.3. Opis tego, że klasa $M_3^{3.2}$ jest podklasą klasy M_1^3 ;
 - 3.1.3.1. Opis tego, że klasa $M_1^{3.2.3}$ jest podklasą klasy $M_3^{3.2}$;
 - 3.1.3.2. Opis tego, że klasa $M_2^{3.2.3}$ jest podklasą klasy $M_3^{3.2}$;
 - ...
 - 3.2. ...

Przykład opisu klas hierarchii dla obiektu diagnostycznego, na podstawie [moc97]:

- 1. Obiekt diagnozowania
 - 1.1. Urządzenie
 - 1.2. Maszyna
 - 1.2.1. Maszyna wirnikowa
 - 1.2.1.1. Turbina
 - 1.2.1.1.1. Turbina wodna
 - 1.2.1.1.2. Turbina parowa
 - 1.2.1.1.2.1. Turbina parowa 360 MW
 - 1.2.1.1.2.2. Turbina parowa 200 MW
 - 1.2.1.1.2.2.1. Turbina parowa 200 MW w elektrowni X
 - 1.2.1.1.2.2.2. Turbina parowa 200 MW w elektrowni Y
 - 1.2.1.1.2.2.3. ...
 - 1.2.1.1.2.3. ...
 - 1.2.1.1.3. ...
 - 1.2.1.2. Pompa
 - 1.2.1.3. ...
 - 1.2.2. ...
 - 1.3. ...

Wszystkie klasy opisane w dowolnym podpunkcie jakiegoś punktu są rozłączne i sumują się do klasy opisanej w tym punkcie. Zatem zachodzi wzór:

$$M \supseteq M_{i_1} \supseteq M_{i_2}^{i_1} \supseteq M_{i_3}^{i_1, i_2} \supseteq M_{i_4}^{i_1, i_2, i_3} \supseteq \dots \supseteq M_{i_n}^{i_1, i_2, i_3, \dots, i_{n-1}} \quad (13)$$

Indeksy postaci i_1, i_2, \dots, i_k są adresami klas $M_{i_k}^{i_1, i_2, i_3, \dots, i_{k-1}}$ – i_k -tej podklasy klasy o adresie i_1, i_2, \dots, i_{k-1} . Adresy te są zarazem adresami relacji $M_{i_{k-1}}^{i_1, i_2, i_3, \dots, i_{k-2}} \supseteq M_{i_k}^{i_1, i_2, i_3, \dots, i_{k-1}}$ bycia klasy o adresie i_1, i_2, \dots, i_k podklasą klasy o adresie i_1, i_2, \dots, i_{k-1} (zgodnie z powyższym wzorem). Jak widać, zbiór adresów klas jest uporządkowany przez relację bycia podklasą i . Wraz z tą relacją tworzy graf zwany niekiedy **drzewem klasyfikacyjnym** lub **klasyfikatorem** [por. moc02, s. 120]. Adresy klas są wierzchołkami tego grafu, a uporządkowane pary tych adresów są jego krawędziami. Jeśli wierzchołki i krawędzie drzewa klasyfikacyjnego opiszemy w języku teoretycznym zastosowanej w pomiarach dziedziny nauki lub techniki, zarówno adresy klas, jak i relacje zawierania podklas, to otrzymamy sieć semantyczną tezaurya reprezentującą pewną wiedzę o realnych obiektach ze zbioru X . Taka sieć utożsamiana będzie w dalszym ciągu z **tezaurem** [roz82, s. 55-60].

Podobnie jak wykorzystanie tezaurya, tak i proces pomiaru prowadzi do wyznaczenia drzewa klasyfikacyjnego odpowiadającego klasom wskazanym w wyniku pomiaru [roz82, s. 195-208]. Klasy te mogą być dopasowane do klas tezaurya z dokładnością do błędu pomiaru, czyli odpowiadają wiedzy adekwatnej lub dokładnej o obiektach. Często jednak są niedopasowane, tj. odpowiadają wiedzy niedokładnej (co omówione zostanie w następnym podrozdziale 2.2.9.). Opisując w języku pragmatyki i praktyki, stosowanym w procesach pomiarów w danej dziedzinie eksperymentalnej nauki lub techniki, adresy klas i relacje zawierania podklas, otrzymamy sieć semantyczną procesu pomiaru reprezentującą wiedzę o pomiarze realnych obiektów ze zbioru X . Ta sieć zawiera sieć semantyczną tezaurya, reprezentującą wiedzę o cechach i własnościach badanych obiektów. Taka sieć utożsamiana będzie w dalszym ciągu z **ontologią**.

Ontologia według ujęcia konsorcjum W3C [w3c14], od 13 lat badającego sieci semantyczne Web, definiuje pojęcia używane do opisanie i reprezentacji gałęzi wiedzy. Ontologie są używane przez agentów (spełniających podaną wcześniej definicję): ludzi, baz danych i aplikacji, które potrzebują informacji pochodzących z danej dziedziny. Dziedzina jest rozumiana jako specyficzna gałąź wiedzy, taka jak np. medycyna, diagnostyka, wytwarzanie narzędzi, handel nieruchomościami, naprawa samochodów, zarządzanie finansami itp. Ontologie zawierają definicje pojęć z danej dziedziny i relacji zachodzących pomiędzy tymi pojęciami, które są czytelne dla komputera. Definicje te nie muszą być ścisłe w sensie rozumianym przez logikę, ale raczej powinny być zrozumiałe dla aplikacji. Ontologie klasyfikują wiedzę w postaci gałęzi wiedzy, a także wiedzę pochodzącą z różnych dziedzin. W ten sposób czynią tę wiedzę dostępną dla człowieka i aplikacji. Tezaurus pewnej dziedziny wiedzy jest w tym sensie ontologią. Jest to słownik, wynik konceptualizacji wiedzy, która łączy zbiór pojęć pochodzących z pewnej dziedziny wiedzy oraz relacje zachodzące

między nimi z opisami tekstowymi czytelnymi dla człowieka, a także regułami poprawnego użycia tych pojęć czytelnymi dla inteligentnej aplikacji. Konceptualizacja z kolei jest uproszczonym obrazem świata, jaki ma inteligentna aplikacja. Może ona być utożsamiana z wiedzą teoretyczną o zbiorze pojęć dotyczących pewnej dziedziny wiedzy i określona przez wspólne dla agentów procedury użycia typów danych reprezentujących wiedzę z tej dziedziny. Te procedury mają swoją implementację w RDF (opisaną w podrozdziale 2.2.7.) oraz w schematach RDFS (ang. *Resource Description Framework Schema*). Dysponując znaczną „siłą wyrazu”, nie mają jednak formalnie zdefiniowanej semantyki. Ograniczenia tego nie ma, rozwijany w celu określenia semantyki, język OIL (ang. *Ontology Inference Language*) [hor00], który definiuje zestaw mechanizmów do modelowania ontologii wraz z ich składnią i formalną semantyką. Projektanci języka OIL przyjęli ważne założenie o jego warstwowej budowie i rozszerzalności. Warstwy *Core OIL* i *Standard OIL* dostarczają pojęć do modelowania ontologii, a warstwa *Instance OIL* umożliwia posługiwanie się przykładami pojęć („obiettami klas”). Zaproponowano także warstwę *Heavy OIL*, która może zawierać dalsze rozszerzenia. Projektując warstwowy model języka i godząc się na ograniczoną „siłę wyrazu” jego trzech warstw niższych, chciano zapewnić efektywność działania bazujących na OIL narzędzi programistycznych (m.in. systemów wnioskujących), a tym samym możliwość powszechnego ich stosowania w Internecie. Dlatego pojawiają się **ontologie szczytowe**, stanowiące klasyfikację najogólniejszych kategorii semantycznych. Ich uszczegółowieniami mogą być np. ontologie: dziedzinowe (ang. *domain ontologies*), zorientowane na zadania (ang. *task ontologies*) lub zorientowane na szczególne zastosowania (ang. *application ontologies*). Na podstawie języka OIL rozwijano język DAML (ang. *DARPA Agent Markup Language*), a projektanci obu języków postanowili połączyć swe siły w projekcie DAML+OIL, który następnie przekształcono w projekt OWL (ang. *Web Ontology Language*) [owl14]. Obecnie stosowany język OWL daje znacznie większe możliwości w zakresie opisu zależności między klasami, a co za tym idzie – w zakresie wnioskowania.

Obserwacja 2.20.

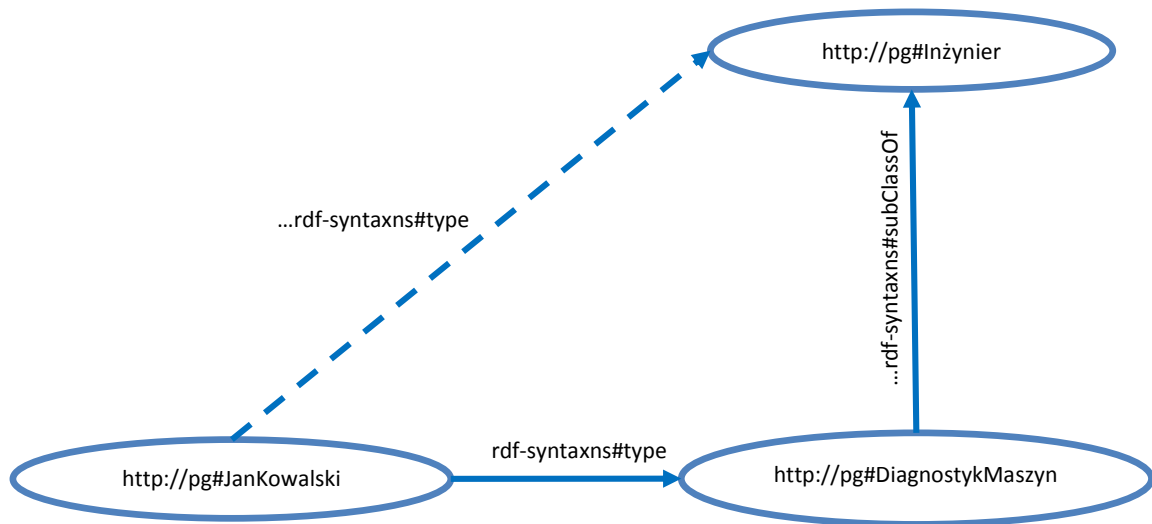
W ujęciu W3C sieci semantycznych ontologię można określić za pomocą opisów wierzchołków i krawędzi sieci. Jeśli w ramach tej sieci będzie się zadawać pytania w języku SPARQL, dysponując wybranymi opisami sieci, to można będzie uzyskać dostęp do innych opisów sieci. Jest tak dlatego, że ten język jest językiem nie tylko zapytań, ale i protokołów (patrz Definicja protokołu 2.2.) dla zasobów opisanych w RDF. Pozwala on wyciągać z opisów dane zawężone według kryteriów określonych przez predykaty RDF za pomocą operacji wnioskowania w RDF. Wnioskowanie to polega na odkrywaniu nowych prawdziwych trójek (stwierżeń, asercji) określających krawędzie sieci semantycznej na podstawie tych, które już znamy.

Wnioskowanie postaci: „Jeżeli *A* jest podklasą klasy *B* i *a* jest obiektem typu *A*, to *a* jest obiektem typu *B*” odbywa się według schematu wnioskowania w RDFS:

(A, rdfs:subClassOf, B)
 (a, rdf:type, A)

 (a, rdf:type B)

Na przykład na rysunku 2.13. przedstawione jest wnioskowanie: *Jeśli Jan Kowalski jest diagnostykiem maszyn, a diagnostycy maszyn są inżynierami, to Jan Kowalski jest inżynierem.* Wynikiem wnioskowania jest trójka opisów wierzchołków i krawędzi narysowana linią przerywaną.



Rysunek 2.13. Wnioskowanie według schematu wnioskowania w RDFS. Opracowanie własne.

Należy również uwzględnić niepewność wiedzy, która jest przetwarzana w procesie wnioskowania. W tym celu omówimy wiedzę nieadekwatną i niedoskonałą wykorzystywaną w systemach danych.

2.2.9. Modele wiedzy nieadekwatnej i niedoskonałej w systemach danych

Na podstawie dotychczasowych rozważań można postawić pytania, które pomogą w ustaleniu odniesień do obiektów. Są to pytania rozstrzygające adekwatność reprezentacji wiedzy o rzeczywistości (obiektach tej rzeczywistości):

- Czy wiedza reprezentowana przez egzemplarze danych odnosi się do tych samych obiektów, które wskazują te egzemplarze?
- Czy procedury poznawcze, w których agenci wykorzystują tę wiedzę, są tymi samymi procedurami poznawczymi, w których agenci na podstawie egzemplarzy typów danych reprezentujących tę wiedzę dowiadują się tego samego?

Intuicyjnie wiedza K jest reprezentowana adekwatnie, jeśli pewien typ e reprezentujący tę wiedzę odnosi się do wszystkich obiektów, do których odnoszą się wszystkie elementy wiedzy $k \in K$, a ponadto $[e]_{\equiv} \in K$. Innymi słowy, adekwatność jest tu taką zgodnością w reprezentowaniu wiedzy, że typ e reprezentuje wiedzę K o tych samych obiektach, które opisują egzemplarze danych typu e . Na przykład adekwatna, w tym sensie jest wiedza o ilości

przedmiotów w dwu-, pięcio- lub dziesięcioelementowych kolekcjach przedmiotów. Tę problematykę precyzują poniższe definicje oraz twierdzenia [fla11].

Referencje – odniesienia do obiektów

Operację $refer: Type \rightarrow 2^{Object}$, określoną wzorem $refer(e) = \{o: o \text{ jest obiektem, istnieje agent } u, \text{ egzemplarz danej } e \text{ typu } e \text{ takie, że } u \text{ za pomocą } e \text{ reprezentuje wiedzę o obiekcie } o\}$, nazywamy **operacją referencji (lub odniesienia) typów danych**, a wartości tej operacji zwane są **referencjami (lub odniesieniami) typów danych**.

Operację $refer_K: K \rightarrow 2^{Object}$, określoną wzorem $refer_K(k) = \{o: o \text{ jest obiektem, istnieje typ } e \text{ taki, że } k = [e]_{\equiv} \text{ oraz istnieje agent } u \text{ i egzemplarz danej } e \text{ typu } e \text{ takie, że agent } u \text{ za pomocą } e \text{ reprezentuje wiedzę o obiekcie } o\}$, nazywa się **operacją referencji (lub odniesienia) elementów wiedzy**. Dla dowolnego typu danych e zachodzi:

$$refer(e) = refer_K([e]_{\equiv}) \quad (14)$$

Adekwatność semantyczna pozwala dla dowolnego typu e danej oraz wiedzy K sprawdzić, czy istnieje rozwiązanie x równania $refer(e) = refer_K(x)$, należące do K i będące znaczeniem e . Szerzej, czy istnieje jednostka wiedzy $k \in K$, spełniająca wyrażenie $refer_K(k) \subseteq refer(e)$, tj. czy wiedza K jest wiedzą o obiektach opisywanych przez egzemplarze danych typu e . Intuicyjnie pytamy o adekwatność reprezentowanej przez e wiedzy K oraz o to, czy wiedza ta dokładnie wyznacza odniesienia typu e danych do obiektów opisanych przez egzemplarze e . W tym sensie pytamy, czy wiedza K jest dokładna. Oznacza to, że istnieje typ zawierający podtypy wyznaczające wszystkie elementy wiedzy K i będący sumą tych podtypów, a odniesienia do obiektów także są sumą odniesień tych podtypów. Można zauważyć, że wskazane podtypy wystarczają do *dokładnego* wyznaczenia wiedzy K i obiektów przez nie opisanych.

Wiedzę K nazywamy **adekwatną**, jeśli istnieje typ e danych taki, że dla dowolnej jednostki wiedzy $k \in K$ zachodzą warunki:

- 1) $refer_K(k) \subseteq refer(e)$,
- 2) istnieje typ danych $e' \subseteq e$, dla którego $k = [e']_{\equiv}$,
- 3) istnieje taka jednostka wiedzy $k' \in K$, że $k' = [e]_{\equiv}$, tj. zachodzi warunek adekwatności określony wzorem (14).

W przeciwnym wypadku tę wiedzę nazywamy **wiedzą nieadekwatną**, niedokładną, a do opisu tej wiedzy można wykorzystać **logikę rozmytą** [fla11].

Wiedzę K nazywamy **dokładną** lub **doskonałą**, jeśli: istnieje typ e danych spełniający równość $refer(e) = \cup refer_K(K)$; dla dowolnej jednostki wiedzy $k \in K$ istnieje typ danych $e' \subseteq e$, dla którego $k = [e']_{\equiv}$ oraz $e = \cup \{e' \in Type: e' \subseteq e \text{ i istnieje } k' \in K, \text{ dla którego } k' = [e']_{\equiv}\}$.

Wiedzę, która nie jest dokładna, nazywamy wiedzą **nieostrą** lub **niedoskonałą**. Wiedza K jest dokładna dla pewnego typu danych e , w tym sensie, że elementy tej wiedzy o obiektach dotyczą tych samych obiektów, o których reprezentowana jest wiedza przez egzemplarze tego

typu danych. Ponadto ten typ danych wyznaczony jest przez egzemplarze typów danych wyznaczających elementy wiedzy K . Innymi słowy, wiedza dla danego typu danych jest dokładna, gdy wystarcza do jednoznacznego ustalenia tego typu danych (w tym zbioru obiektów, których te dane dotyczą). Na podstawie powyższej definicji przyjmujemy, że [fla11, rozdz. 12 i 13]:

- Wiedza składająca się z jednej jednostki wiedzy jest adekwatna.
- Wiedza reprezentowana przez każdą z danych zawartych w pewnym typie danych jest wiedzą dokładną.
- Wiedza adekwatna jest wiedzą dokładną.
- Wiedza nieostra jest wiedzą nieadekwatną.

Przykład 2.10.

Wiedza $K = \{k_1, k_2, k\}$, gdzie elementy wiedzy k_1, k_2, k dotyczą odpowiednio liczby przedmiotów, w dwu-, pięcio- lub dziesięcioelementowych kolekcjach przedmiotów jest wiedzą adekwatną. Wiedza odnosi się do dziesięcioelementowych kolekcji przedmiotów. Rzeczywiście, jeśli danymi o liczbie przedmiotów są $\{[1], [2], [3], \dots, [10]\}$, to:

- 1) $e_1 = \text{dwuelementowe_kolekcje} = \{[1], [2]\}, k_1 = [e_1]_{\equiv}$,
- 2) $e_2 = \text{pęcioelementowe_kolekcje} = \{[1], [2], [3], [4], [5]\}, k_2 = [e_2]_{\equiv}$,
- 3) $e = \text{dziesięcioelementowe_kolekcje} = \{[1], [2], [3], \dots, [10]\}, k = [e]_{\equiv}$.

Ponieważ wszystkie wyróżnione typy danych odnoszą się do tych samych kolekcji, co element wiedzy k , to warunki definicji adekwatności wiedzy są spełnione.

Przykład 2.11.

Rozważmy dane wyznaczone przez egzemplarze z napisami „zimno”, „ciepło”, „w odczuciu 12°C”, „odczuwana temperatura”. Dla każdego z podanych napisów używanych przez różnych agentów mogą istnieć różne dane, odnoszące się do temperatur od -5°C do 20°C . W takiej sytuacji wiedza reprezentowana przez te dane nie może być adekwatna, gdyż nie można znaleźć używanego przez wszystkich agentów typu danych, który by wyznaczał jednakową dla wszystkich napisów wiedzę.

Powyższy przykład w typowy sposób prezentuje wiedzę, do której opisu można zastosować logikę rozmytą. Wiedzę o wartościach danych z pewnego zakresu X , wyrażonych przez opisy $A_1, A_2, \dots, A_k, \dots$, uważa się za reprezentowaną w sposób rozmyty, gdy każdy opis A_k odnosi się do funkcji przynależności $\mu_{A_k}: X \rightarrow [0, 1]$. Funkcja przynależności ustala stopień z przedziału $[0, 1]$ posiadania przez x z zakresu X własności opisanej przez A_k . Stopień ten określa, na ile zastosowanie tego opisu jest słuszne, adekwatne do rzeczywistości [fla11]. Takie opisy wartości zmiennej są nazwami zmiennej lingwistycznej w sensie Zadeha (por. Definicja 2.4. zmiennej lingwistycznej). Dokładniejszy opis podstaw logiki rozmytej przedstawiono w rozdziale 2.3.

W teoriomnogościowym ujęciu, gdy A jest zbiorem rozmytym, to $A = \{ \langle x, \mu_A(x) \rangle : x \in X, \mu_A: X \rightarrow [0, 1] \}$. Potrzeba stosowania zbiorów rozmytych w systemach danych określona jest zróżnicowaniem stopni, z jakimi agenci opisują dane w zakresie X . Określają one słusność zastosowania danego opisu, czyli tego, że „ x jest A_k ”. Ta niejednoznaczność wiedzy agentów powoduje, że **wiedza reprezentowana w sposób rozmyty jest wiedzą nieadekwatną** [fla11].

Przykład 2.12.

Podczas pomiarów, w których wszystkie wartości danych mają dobrze określony rozkład prawdopodobieństwa, wiedza o badanych obiektach jest ustalona dokładnie, tj. z dokładnością pomiaru, a więc typy danych określone przez wartości mierzonych wielkości wyznaczone są jednoznacznie przez elementy wiedzy reprezentowane przez dane pomiaru. Jednak, jak zauważa Zadeh [zad02, s. 3-34], sytuacja dokładności wiedzy może ulec zmianie, gdy rozkłady prawdopodobieństwa są niepewne. Taką sytuację reprezentują następujące wypowiedzi podane przez Zadeha:

„Zazwyczaj Robert wraca z pracy około godz. 6 po południu” lub

„Jest bardzo mało prawdopodobne, że nastąpi znaczny wzrost cen ropy w najbliższej przyszłości”.

Wiedza reprezentowana przez te wypowiedzi nie jest wiedzą dokładną, ale wiedzą nieprecyzyjną, do której opisu można wykorzystać logikę rozmytą.

Przykład 2.11. i sposób określania sieci semantycznej prowadzi także do następującej obserwacji.

Obserwacja 2.21.

W systemie danych DS, różne klasyfikacje mogą wyznaczać jednakowe sieci semantyczne, tj. o tej samej budowie oraz opisach wierzchołków i krawędzi. Suma takich klasyfikacji też będzie wyznaczała sieć o tej samej budowie co poprzednie sieci. Sieci semantyczne Web zazwyczaj można traktować jako tak określoną sieć semantyczną. Z tego powodu wiedza reprezentowana przez opisy wierzchołków i krawędzi może być wiedzą nieadekwatną (por. Obserwacja 2.17.). Jeśli szukamy wiedzy o obiektach reprezentowanych przez wierzchołki i krawędzie tej sieci, to o takiej wiedzy mówimy, że jest **wyszukiwana w sieci semantycznej**.

W ujęciu zbiorów rozmytych traktuje się zbiory rozmyte jako interpretacje pojęć nieostrych [fla11] (jest to tzw. operacja rozmywania) oraz wprowadza się pewną interpretację (standardową) operacji wyostrzania tego pojęcia. W pracach [bry11a], [bry11b] autorka niniejszej pracy pozwala sobie szerzej spojrzeć na interpretacje operacji rozmywania i wyostrzania reprezentacji wiedzy wyszukiwanej w sieciach semantycznych. W pracach tych wprowadza się interpretacje rozmywające i wyostrzające dla opisów wiedzy w języku logiki opisowej (ang. *description logic*), co zostanie szerzej omówione w rozdziałach 2.4. i 2.5.

2.2.10. Modele wiedzy diagnostycznej w informatycznych systemach danych

Komputerowe systemy pomiarowe zaczęły pełnić rolę *systemów ekspertowych – agentów*, pozwalających człowiekowi dokonywać pomiaru lub jego symulacji, sterować tym systemem oraz monitorować i automatycznie zbierać oraz archiwizować dane z pomiaru. Te przemiany dotyczyły także komputerowych systemów diagnostycznych.

Michael Schroeder w swojej monografii wydanej w 1998 roku [sch98], zainspirowany kosmiczną technologią NASA, prezentuje szeroką problematykę teoretyczną odnoszącą się do modelowej bazy wiedzy *agenta diagnozy* i algorytmów wnioskowania diagnostycznego. Zarysowuje się pewna koncepcja projektowania robotów do prowadzenia diagnozy, sterowanych przez komputerowego inteligentnego agenta diagnozy. Taki agent może korzystać z zaimplementowanych modeli diagnozy, opisywanych przez *tezaurusy*, a formalnie zaprezentowanych w standardowych monografiach diagnostyki procesów [kor02], [uli96], [zół96]. Może on również korzystać z wiedzy reprezentowanej w sieci komputerowej, w tym sieci semantycznej Web, opisanej przez tzw. *ontologie (ontologie diagnozy)*. Teoretyczne koncepcje tworzenia ontologii diagnozy przedstawione są w dysertacjach doktorskich Urlicha HellerA [hel01] i Joachima BaumeisterA [bau04].

Współcześnie wiedza techniczna coraz częściej pozyskiwana jest z globalnych sieci komputerowych. Do takich sieci należą m. in. sieć semantyczna Web, określona przez projekt W3C [w3c14]. Ważna jest też sieć, określona przez projekt europejski MIMOSA (ang. *Microsystems Platform for Mobile Services and Applications*), którego celem jest stworzenie inteligencji otoczenia (ang. *Ambient Intelligence*) poprzez opracowanie otwartej platformy dla telefonii komórkowej [jan12]. Inny projekt o nazwie MIMOSA, stworzony do zarządzania przedsiębiorstwem (ang. *Machinery Information Management Open Systems Alliance*) [mim13], umożliwia definiowanie obiektów technicznych za pomocą biblioteki DTD języka XML. Istotnym aspektem projektu MIMOSA jest wyróżnienie zespołów funkcyjno-usługowych (ang. *segment*) oraz egzemplarzy, konkretności (ang. *asset*) [moc02]. W ten sposób w wymienionych sieciach można reprezentować wiedzę techniczną. Te metody można stosować do wiedzy technicznej reprezentowanej w sieciach semantycznych [kor02].

Do programu WEB W3C, logiki deskrypcyjnej oraz do projektu MIMOSA nawiązuje koncepcja ontologii prezentowana w programie projektu SEKT (ang. *Semantically Enabled Knowledge Technologies*, 2003-2005), przedstawionym przez Stefana Schlobacha i Zhishenga Huanga w pracy [sch05].

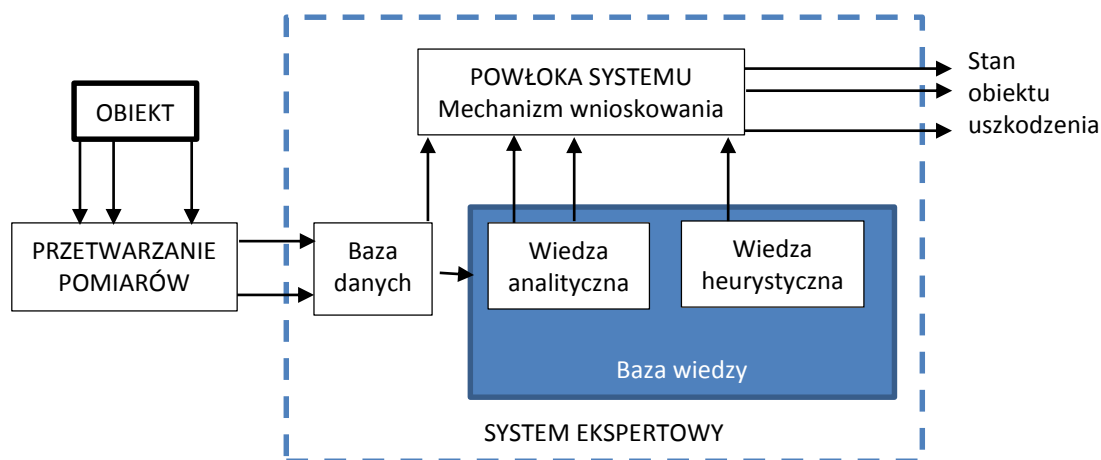
Modele wiedzy diagnostycznej

W diagnostyce technicznej, jak i w innych dziedzinach badań naukowych, dane są wyrażane w języku zgodnym z metodologią tej dziedziny. Badacze, teoretycy i eksperymetatorzy są agentami systemu danych, którzy używają nie tylko języka naturalnego, ale również języka zgodnego z terminami stosowanymi w danej dziedzinie. Jeżeli egzemplarze tych terminów są poprawnie używane, zgodnie z terminologią danej dziedziny, to tworzy się równoważne typy danych. Dane te tworzą **model wiedzy diagnostycznej**.

Dzięki technologii informacyjnej można rozszerzyć komunikację w językach naturalnych o języki sztuczne, używane i przetwarzane przez komputery. Wtedy egzemplarze danych są reprezentowane w obu tych językach (naturalnym i sztucznym). W ten sposób można stworzyć **komputerowe modele wiedzy** implementowane w językach programowania komputerów, w tym w językach znacznikowych tworzących zasoby internetowe. Daje to możliwość rozważania modeli wiedzy diagnostycznej reprezentowanych za pomocą sieci semantycznych Web [zół12]. Przy określaniu tych modeli bardzo istotne jest utożsamianie systemów diagnostycznych nie tylko z systemami pomiarowymi, ale także z systemami ekspertowymi [zół12, s. 46], [kor02, s. 543-580].

Systemy ekspertowe w diagnostyce [zół12, s. 45] można rozważać jako pewne bazy danych, w których znajduje się system reguł: „JEŻELI-TO” (ang. „IF-THEN”), umożliwiający wnioskowanie diagnostyczne (rysunek 2.14.). Wnioskowanie dotyczy badanego stanu na podstawie modeli danych ilościowych oraz jakościowych. Bazę wiedzy w takim systemie umownie można podzielić na [zół04], [zół96] (rysunek 2.14.):

- **wiedzę analityczną** – są to modele matematyczne zadane odpowiednimi równaniami, algorytmy estymacji stanu zwane obserwatorami, algorytmy identyfikacji parametrów oraz algorytmy lub testy przetwarzania symptomów,
- **wiedzę heurystyczną** – jest to wiedza jakościowa, umownie wiedza operatora (eksperta z tej dziedziny).



Rysunek 2.14. Schemat blokowy diagnostycznego systemu ekspertowego. Źródło [zół12, s. 46].

Przy budowie modeli jakościowych diagnozowanych obiektów oraz podczas przetwarzania symptomów można zastosować **logikę rozmytą**. W przypadku rozmytego modelowania (jakościowego) dynamika procesu jest opisywana przez stosunkowo małą liczbę symboli i wielkości liczbowych dzięki zastosowaniu zbiorów rozmytych z charakterystycznymi funkcjami przynależności. Funkcje przynależności mogą określać różne typy modeli obiektów: jakościowe, rozmyto-jakościowe oraz ilościowe.

Określenie tablicy decyzyjnej Symptom-Stan

W procesie użytkowania maszyny bardzo istotne są takie aspekty, jak niezawodność maszyny, efektywność jej użytkowania, proces jej technicznego obsługiwanie. Jeżeli w tym procesie dojdzie do zaburzenia, które wpływa na wymienione aspekty, to następuje *uszkodzenie maszyny* (zespołu, podzespołu, elementu). Najogólniej uszkodzenie maszyny to zdarzenie polegające na przejściu maszyny ze stanu zdatności do stanu niezdatności [moc02]. Przez **stan zdatności** rozumie się taki stan, w którym maszyna spełnia wyznaczone funkcje i zachowuje parametry określone w dokumentacji technicznej. **Stan niezdatności** określa natomiast stan maszyny, w którym nie spełnia ona przynajmniej jednego z wymagań określonych w dokumentacji technicznej [niz00], [zól96].

Główne przyczyny powstania uszkodzeń to:

1. **konstrukcyjne** – powstałe wskutek błędów projektowania i konstruowania obiektu,
2. **produkcyjne (technologiczne)** – powstałe wskutek błędów i niedokładności procesów technologicznych,
3. **eksploatacyjne** – powstałe w wyniku nieprzestrzegania obowiązujących zasad eksploatacji lub na skutek oddziaływań czynników zewnętrznych nieprzewidzianych dla warunków użytkowania danego obiektu,
4. **starzeniowe** – wynikające ze zmiany parametrów obiektu w czasie eksploatacji.

Głównym celem diagnostyki maszyn jest wskazanie stanu maszyny X_n na podstawie generowanych przez tę maszynę sygnałów diagnostycznych S_m . Oczywiście mogą jeszcze wystąpić inne oddziaływania zewnętrzne, jednak na razie zostaną one pominięte. W tabeli 2.2. przedstawiono tablicę obserwacji symptomów dla wybranych cech stanu obiektu (zdadności i niezdatności).

Tabela 2.2. Tablica obserwacji symptomów S_m dla wybranych cech stanu obiektu X_n . Opracowanie na podstawie [zól12].

Cechy stanu obiektu X_n	Symptomy S_m	Wartości mierzonych symptomów									
	C	K	I	P_{sk}	...	F_0	...	H_v	...	m	
1. Bicie											
2. Luz											
3. Zacisk											
...	Symptomy diagnostyczne w dziedzinie czasu, amplitud, częstotliwości (wymiarowe, bezwymiarowe)										
...											
...											
n. Ilość pracy											

Możliwe stany obiektu można podzielić na dwie klasy:

- klasę stanów zdatności,
- klasę stanów niezdatności – przy wystąpieniu uszkodzeń.

Klasa stanów zdatności reprezentuje zdatność funkcjonalną i zadaniową. Zdatość funkcjonalna to możliwość realizacji przez maszynę zadanej funkcji zgodnie z jej przeznaczeniem, w określonym przedziale czasu i zgodnie z założeniami dokumentacji. Natomiast zdatość zadaniowa również określa możliwość realizacji wyznaczonych zadań, jednak w warunkach eksploatacyjnych przy określonych wymuszeniach sterujących i obciążeniach w losowo zmiennym przedziale czasu [zól12].

Stany niezdatności determinują wystąpienie pewnego uszkodzenia w obiekcie. Wyróżnia się:

- **niezdatność częściową** – charakteryzującą się niezdolnością obiektu do spełniania niektórych, lecz nie wszystkich wymaganych funkcji,
- **niezdatność krytyczną** – stwarzającą zagrożenie dla ludzi, pociągającą za sobą znaczne straty materialne lub inne niedopuszczalne skutki,
- **niezdatność trwałą** – trwającą do czasu wykonania obsługi naprawczej.

Modelowy opis stanu maszyny jest ograniczony dostępnymi wskaźnikami i jest budowany na podstawie przyjętych kryteriów. Z tego powodu powinien być na tyle dokładny, aby umożliwiał różnicowanie zmian zachodzących w rzeczywistej maszynie. Utrudniają to losowe właściwości procesów wytwarzania, kontroli i eksploatacji oraz złożoność maszyn. Bezpośrednie przewidywanie relacji między cechami sygnałów diagnostycznych a cechami stanu maszyny jest przez to utrudnione. Właściwości obiektu, za pomocą których tworzy się jego model, nazywa się często cechami. **Cecha** obiektu jest to wielkość fizyczna, charakteryzująca go ze względu na działanie zgodne z przeznaczeniem. Wśród cech obiektu wyróżnić można:

- **cechy funkcjonalne** – określające zadania (funkcje) obiektu lub sprawdzające jakość, stopień wykonywania tych zadań,
- **cechy konstrukcyjne** – opisujące obiekt lub jego elementy ze względu na zasadę pracy, sposób współdziałania elementów lub ich wykonanie,
- **cechy obsługowe** – charakteryzujące się tym, że ich wartość może być zmieniana w czasie obsługiwanego,
- **cechy diagnostyczne** – kontrolowane w czasie działań diagnostycznych; są to **cechy stanu**, gdyż za pomocą tych cech opisuje się stany obiektu.

Cecha stanu to związana z właściwością maszyny wielkość fizyczna posiadająca miarę, wzorzec i poziom odniesienia, jednoznacznie opisująca wartość składowej wektora chwilowego stanu maszyny. Cechy charakteryzujące obiekt i jego stan, które występują tylko w czasie, gdy obiekt jest uszkodzony lub nie w pełni zdalny, nazywamy **symptomami**. Także *symptom diagnostyczny* określa miarę sygnału diagnostycznego, odwzorowującego określony typ uszkodzenia (składowa wektora sygnału). Natomiast cechy, które wyznaczają stan

zdatności obiektu, nazywane są **parametrami**. *Parametr diagnostyczny* jest zawsze związany z obserwowalnym opisem obiektu diagnozowanego za pomocą sygnałów (procesów) diagnostycznych. Ponadto określa pośrednio wartości cech stanu obiektu [zół12].

Głównym etapem wypracowania decyzji w procesie diagnostycznym jest wnioskowanie diagnostyczne. Decyzja, odpowiedź systemu diagnostycznego, odwzorowuje relację symptom – stan.

Model rozmyty wiedzy diagnostycznej

Prezentowana wyżej tablica 2.2. obserwacji symptomów dla wybranych cech stanu obiektu jest przykładem tablicy decyzyjnej dla danych diagnostycznych, tj. danych z obserwacji diagnostycznych. Ogólnie taką tablicę można przedstawić następująco (tabela 2.3.):

Tabela 2.3. Schemat tablicy decyzyjnej dla danych diagnostycznych. Opracowanie własne.

Reguły	Symptomy	Stan f
	$s_1, s_2, \dots, s_i, \dots, s_n$	
r_1	$r_{11}, r_{12}, \dots, r_{1i}, \dots, r_{1n}$	f_1
r_2	$r_{21}, r_{22}, \dots, r_{2i}, \dots, r_{2n}$	f_2
.	.	.
.	.	.
.	.	.
r_l	$r_{l1}, r_{l2}, \dots, r_{li}, \dots, r_{ln}$	f_l
.	.	.
.	.	.
.	.	.
r_k	$r_{k1}, r_{k2}, \dots, r_{ki}, \dots, r_{kn}$	f_k

Symptomy i stan diagnostyczny (usterka, wada, uszkodzenie) są atrybutami odpowiednio warunkowymi i decyzyjnymi tej tablicy. Niech zbiór $S = \{s_1, s_2, \dots, s_i, \dots, s_n\}$ będzie zbiorem symptomów, a f będzie atrybutem stanu diagnostycznego. Zbiór $F = \{f_1, f_2, \dots, f_l, \dots, f_k\}$ jest zbiorem danych będących wartościami atrybutu stanu diagnostycznego, a elementy macierzy $r_{ij}, i = 1, 2, \dots, k, j = 1, 2, \dots, n$ są danymi będącymi wartościami symptomów. Wymienione dane są danymi o obserwacjach diagnostycznych, odpowiednio obserwacjach symptomów i stanu diagnostycznego.

Reguły wnioskowania w tablicy decyzyjnej dla danych diagnostycznych określają wzory:

$$r_l: \text{JEŻELI } (s_1 = r_{l1}) \text{ i } \dots \text{ i } (s_n = r_{ln}) \text{ TO } f = f_l, l = 1, 2, \dots, k \quad (15)$$

Ważne jest, że wyrażenie postaci „ $s_i(o) = d$ ” jest formułą atomową, dla $i = 1, 2, \dots, n$. Gdy obiekty-obszawacje diagnostyczne o są wyspecyfikowane (zadane) dla wartości d , to zapis można skrócić do: $s_i = d$. Mówimy, że obiekt (obszawacja) o jest wyspecyfikowany dla danej d , jeśli dana d reprezentuje wiedzę o tej obszawacji, tj. istnieje agent a i egzemplarz $e \in d$ takie, że a za pomocą e reprezentuje wiedzę o .

Jeśli dla danych wartości symptomów mogą występować różne wartości stanów, to tablica decyzyjna przybiera postać jak w tabeli 2.4.

Tabela 2.4. Schemat tablicy decyzyjnej dla wieloznacznych wartości stanu. Opracowanie własne.

Reguły	Symptomy	Stan f
	$s_1, s_2, \dots, s_i, \dots, s_n$	
r_1	$r_{11}, r_{12}, \dots, r_{1i}, \dots, r_{1n}$	F_1
r_2	$r_{21}, r_{22}, \dots, r_{2i}, \dots, r_{2n}$	F_2
.	.	.
.	.	.
.	.	.
r_l	$r_{l1}, r_{l2}, \dots, r_{li}, \dots, r_{ln}$	F_l
.	.	.
.	.	.
.	.	.
r_k	$r_{k1}, r_{k2}, \dots, r_{ki}, \dots, r_{kn}$	F_k

W tabeli 2.4. stany $F_i \subseteq F$, dla $i = 1, 2, \dots, k$. Natomiast reguły wnioskowania, dla tej tablicy decyzyjnej, określają wzory:

$$r_l: \text{JEŻELI } (s_1 = r_{l1}) \text{ i } \dots \text{ i } (s_n = r_{ln}) \text{ TO } f = f_{l1} \text{ lub } f = f_{l2} \text{ lub } \dots \text{ lub } f = f_{lm},$$

$$F_l = \{f_{l1}, f_{l2}, \dots, f_{lm}\}, l = 1, 2, \dots, k \quad (16)$$

Tablicy obserwacji 2.2. odpowiada tablica decyzyjna 2.5.

Tabela 2.5. Schemat tablicy decyzyjnej dla danych diagnostycznych odpowiadający tabeli 2.2. Opracowanie własne.

Reguły	Symptomy	Stan f
	$C, K, I, P_{sk}, \dots, F_0, \dots, H_v, \dots, m$	
r_1	$r_{11}, r_{12}, r_{13}, r_{14}, \dots, r_{1i}, \dots, r_{1j}, \dots, r_{1n}$	Bicie
r_2	$r_{21}, r_{22}, r_{23}, r_{24}, \dots, r_{2i}, \dots, r_{2j}, \dots, r_{2n}$	Luz
.	.	.
.	.	.
.	.	.
r_l	$r_{l1}, r_{l2}, r_{l3}, r_{l4}, \dots, r_{li}, \dots, r_{lj}, \dots, r_{ln}$	Zacisk
.	.	.
.	.	.
.	.	.
r_k	$r_{k1}, r_{k2}, r_{k3}, r_{k4}, \dots, r_{ki}, \dots, r_{kj}, \dots, r_{kn}$	Ilość pracy

Reguły wnioskowania w tej tablicy określają wzory takie jak np.:

$$r_l: \text{JEŻELI } (C = r_{l1}) \text{ i } (K = r_{l2}) \text{ i } (P_{sk} = r_{l3}) \text{ i } \dots \text{ i } (F_0 = r_{li}) \text{ i } \dots \text{ i } (m = r_{ln})$$

$$\text{TO } (f = \text{Zacisk}) \quad (17)$$

Tablica decyzyjna jest szczególnym przypadkiem macierzy diagnostycznej. W monografiach [kor02], [kor09] i [koś08] przedstawia się metodę reprezentowania wiedzy diagnostycznej w macierzach diagnostycznych. **Macierz diagnostyczna** r_{ij} dla wartości symptomów (tabela 2.6.) może być określona funkcją $r: F \times S \rightarrow V$, gdzie V jest zbiorem danych, które są wartościami symptomów, przy występowaniu danego stanu diagnostycznego: $r_{ij} = r(f_i, s_j)$.

Tabela 2.6. Schemat macierzy diagnostycznej. Opracowanie własne.

S\F	$f_1, f_2, \dots, f_i, \dots, f_k$
s_1	$r_{11}, r_{12}, \dots, r_{1i}, \dots, r_{1k}$
s_2	$r_{21}, r_{22}, \dots, r_{2i}, \dots, r_{2k}$
.	.
.	.
.	.
s_l	$r_{l1}, r_{l2}, \dots, r_{li}, \dots, r_{lk}$
.	.
.	.
.	.
s_n	$r_{n1}, r_{n2}, \dots, r_{ni}, \dots, r_{nk}$

Występowanie pewnej wartości symptomu podczas obserwacji nie oznacza występowania związku pomiędzy tym symptomem a stanem diagnostycznym. Jeśli taki związek jest znany, to jest relacją $R \subseteq F \times S$, która określa **binarną macierz diagnostyczną**:

$$r_{ij} = r(f_i, s_j) = \begin{cases} 0, & (f_i, s_j) \notin R \\ 1, & (f_i, s_j) \in R \end{cases} \quad (18)$$

Niech zbiory X_i , dla $i = 1, 2, \dots, n$, będą przestrzeniami (zakresami) obserwacji symptomów s_i , a niektóre dane o tych obserwacjach są rozmyte, wtedy symptomy są zmiennymi lingwistycznymi o wartościach rozmytych, a macierz diagnostyczna jest macierzą zbiorów rozmytych:

$$r_{ij} = r(f_i, s_j) = \{ \langle o_{ij}, \mu_{ij}(o_{ij}) \rangle : o_{ij} \in X_i, \mu_{ij}: X_j \rightarrow [0,1] \} \quad (19)$$

Podobnie niektóre rozmyte dane o obserwacjach i -tych stanów diagnostycznych mogą reprezentować informację niepełną i nieprecyzyjną o stanie diagnostycznym f_i . Niech zbiory F_i , dla $i = 1, 2, \dots, k$, będą przestrzeniami (zakresami) obserwacji stanów f_i , wtedy atrybut stanu diagnostycznego jest zmienną lingwistyczną o i -tej wartości:

$$f'_i = \{ \langle o_i, \mu_i(o_i) \rangle : o_i \in F_i, \mu_i: F_i \rightarrow [0,1] \} \quad (20)$$

Wszelkie dane reprezentujące wiedzę dokładną lub adekwatną można interpretować jako zbiory rozmyte o funkcji przynależności będącej funkcją charakterystyczną. Wtedy tablica decyzyjna o danych rozmytych nazywa się **rozmytą tablicą decyzyjną** [kor09], [koś08].

Dla skończonych, m -elementowych zbiorów obserwacji jako liczbowych wyników pomiarów symptomów, o wartościach w zbiorze liczb rzeczywistych oraz zastosowaniu wyostrzenia zbiorów rozmytych metodą środka ciężkości, otrzymuje się **rozmytą macierz diagnostyczną** [kor09], [koś08]:

$$r_{ij} = r(f_i, s_j) = \frac{\sum_{l=1}^m o_{jl} \mu_{ij}(o_{jl})}{\sum_{l=1}^m \mu_{ij}(o_{jl})} \quad (21)$$

gdzie $o_{jl} \in X_j$, a r_{ij} jest stopniem bycia w związku stanu f_i z symptomem s_j .

Zbiór danych przedstawiony za pomocą macierzy lub tablicy decyzyjnej może być wykorzystany w procesie odkrywania wiedzy [moc02], co zostanie pokazane na przykładzie w rozdz. 3.4.

2.3. Wybrane zagadnienia z teorii zbiorów rozmytych

Niniejszy podrozdział prezentuje podstawowe definicje i zagadnienia teorii zbiorów rozmytych wykorzystywane w niniejszej pracy. Przedstawiono operacje mnogościowe wykonywane na zbiorach rozmytych, struktury t-norm i s-norm oraz wnioskowanie w logice rozmytej. Tematyka ta jest szeroko opisywana w literaturze polskiej [kac86], [kac01], [łęs08], [czo85], [rut97], [rut05], [pie99], [ryk06], [wie09] i zagranicznej [zad65], [zad99], [yag95], [kli95], [haj98]. Wprowadzony aparat pojęciowy będzie służył do określenia struktur relacyjnych opisujących zbiory rozmyte, w których będzie interpretowana logika wyszukiwania informacji (rozdz. 2.4. i 2.5.).

2.3.1. Teoria zbiorów rozmytych

Klasyczna teoria zbiorów, w której stosuje się logikę dwuwartościową, nie jest wystarczająca w przypadku, gdy analizowane są procesy lub zjawiska, w których występuje niepewność i niejednoznaczność. W 1965 roku Lotfi Zadeh [zad65] przedstawił słynną pracę wprowadzającą teorię *zbiorów rozmytych* (ang. *fuzzy sets*). Teoria ta uwzględnia wieloznaczny charakter procesów, przez co pozwala na ich opis oraz analizę w przypadkach, gdy klasyczna logika dwuwartościowa się nie sprawdza.

Zbiory rozmyte pozwalają opisać takie wieloznaczne pojęcia, jak na przykład „wysoka temperatura”, „średnie zużycie materiału”, „duży hałas”. Przy definiowaniu zbioru rozmytego należy określić obszar rozważań (ang. *the universe of discourse*) [zad65]. Jest to istotne, ponieważ pojęcie „wysoka temperatura” będzie miało zupełnie inne znaczenie, gdy będziemy rozważać czujnik temperatury powietrza np. w zakresie $[-40^{\circ}\text{C}, 80^{\circ}\text{C}]$, a inne, gdy będziemy brać pod uwagę temperaturę gazów spalinowych w samochodzie w przykładowym zakresie $[-40^{\circ}\text{C}, 900^{\circ}\text{C}]$. Obszar rozważań poniżej będzie nazywany *przestrzenią* oraz będzie oznaczony jako X .

Definicja (zbiór rozmyty) 2.3.

Zbiór rozmyty A w pewnej (niepustej) przestrzeni X , oznaczany jako $A \subseteq X$, jest zbiorem par:

$$A = \{(x, \mu_A(x)); x \in X; \mu_A: X \rightarrow [0,1]\} \quad (22)$$

gdzie μ_A jest *funkcją przynależności* zbioru rozmytego A . Funkcja ta dla każdego elementu $x \in X$ przypisuje jego stopień przynależności do zbioru rozmytego A . Wartości stopnia przynależności do zbioru można podzielić na trzy przypadki:

- 1) $\mu_A = 1$, pełna przynależność elementu do zbioru rozmytego,
- 2) $\mu_A = 0$, brak przynależności elementu do zbioru rozmytego,
- 3) $0 < \mu_A < 1$, częściowa przynależność elementu do zbioru rozmytego.

Gdy dla danego zbioru rozmytego występuje tylko przypadek 1) i 2), to ten zbiór może być interpretowany jako zbiór nierozmyty.

Bardzo istotna jest kwestia interpretacji wartości stopnia rozmycia zbioru rozmytego, w języku naturalnym, zrozumiałym dla człowieka. Służy do tego zmienna lingwistyczna. Zmienne lingwistyczne określa się za pomocą następującej definicji.

Definicja (zmienna lingwistyczna) [zad75] 2.4.

Zmienna lingwistyczna (ang. *linguistic variable*) definiowana jest jako piątka:

$$\{x_{nazwa}, L(x), X, G, M_x\} \quad (23)$$

gdzie:

x_{nazwa} – nazwa zmiennej lingwistycznej (np. temperatura powietrza, poziom hałasu),

$L(x)$ – zbiór wartości (terminów, termów) lingwistycznych (ang. *linguistic value*), jakie przyjmuje x , czyli słowa lub zdania w języku naturalnym (np. duży, średni, mały),

X – obszar rozważań,

G – zbiór reguł syntaktycznych, które umożliwiają utworzenie wszystkich terminów w zbiorze $L(x)$,

M_x – funkcja semantyczna, przyporządkowująca każdej wartości lingwistycznej ze zbioru $L(x)$ zbiór rozmyty zdefiniowany nad X .

Na przykład niech zmienna lingwistyczna o nazwie $x =$ „poziom natężenia dźwięku”, zdefiniowana w obszarze rozważań $X = [10,350]$ dB, posiada zbiór wartości lingwistycznych: $L(x) = \{„niski”, „średni”, „wysoki”, „cichy”, „głośny”, „ogłuszający”\}$. Wówczas definicję wartości lingwistycznej „wysoki” można zapisać przykładowo jako:

$$\mu_{wysoki}(x) = \begin{cases} 0, & \text{dla } x < 50 \\ 1 - \left(\frac{100-x}{50}\right)^2, & \text{dla } x \in [50,100] \\ 1, & \text{dla } x \geq 100 \end{cases} \quad (24)$$

2.3.2. Podstawowe definicje i operacje na zbiorach rozmytych

Poniżej zdefiniowano podstawowe pojęcia z teorii zbiorów rozmytych, takie jak zbiór rozmyty pusty, zawieranie się zbiorów rozmytych i równość tych zbiorów na podstawie [zad65], [zad99], [kac86], [łes08], [yag95].

Definicja (zbiór pusty) 2.5.

Zbiór rozmyty $A \subseteq X$ nazywamy zbiorem pustym, co zapisujemy $A = \emptyset$, wtedy i tylko wtedy, gdy:

$$\mu_A(x) = 0, \text{ dla każdego } x \in X \quad (25)$$

Definicja (zawieranie się zbiorów) 2.6.

Niech zbiory $A, B \subseteq X$ będą zbiorami rozmytymi. Wtedy zbiór A zawiera się w zbiorze B (A jest podzbiorem B), co zapisujemy $A \subset B$, jeśli dla każdego $x \in X$:

$$\mu_A(x) \leq \mu_B(x) \quad (26)$$

Ta klasyczna definicja zawierania się dwóch zbiorów jest bardzo „twarda” (trudno ją dostosować do opisu rzeczywistości). Na przykład zbiory rozmyte nie zawsze spełniają warunek (26) dla wszystkich wartości x , innymi słowy „nie zawsze zawierają się całkowicie”, dlatego wprowadzono pojęcie *stopnia zawierania się zbiorów rozmytych*. Dla zbiorów rozmytych A, B , zawierających się częściowo, stopień $I(A \subset B)$ zawierania się zbioru A w zbiorze B proponuje się zdefiniować następująco:

$$I(A \subset B) = \min_{x \in S} \mu_B(x), \text{ gdzie } S = \{x \in X; \mu_A(x) \leq \mu_B(x), \mu_A(x) > 0\} \quad (27)$$

Definicja (równość zbiorów) 2.7.

Zbiory rozmyte $A, B \subseteq X$ są równe, co zapisujemy $A = B$, jeżeli dla każdego $x \in X$:

$$\mu_A(x) = \mu_B(x) \quad (28)$$

Wtedy $A \subset B$ i $B \subset A$. Podobnie jak w przypadku zawierania się zbiorów, wprowadzono pojęcie *stopnia równości zbiorów rozmytych*, gdy zbiory są prawie równe. Stopień równości proponuje się określić następująco:

$$E(A = B) = 1 - \max_{x \in S} |\mu_A(x) - \mu_B(x)|, \text{ gdzie } S = \{x \in X; \mu_A(x) \neq \mu_B(x)\} \quad (29)$$

Do podstawowych operacji mnogościowych na zbiorach rozmytych zaliczamy sumę, przecięcie oraz dopełnienie zbioru rozmytego. Operacje te są również wykorzystywane przy interpretacji zbiorów w rozmytej logice opisowej prezentowanej w podrozdziale 2.4.2. Poniżej zostaną zdefiniowane operacje mnogościowe na podstawie [zad65], [kac86], [łes08], [yag95].

Definicja (przecięcie zbiorów rozmytych) 2.8.

Przecięciem (iloczynem) zbiorów rozmytych $A, B \subseteq X$ jest zbiór rozmyty $A \cap B$ o funkcji przynależności, dla każdego $x \subseteq X$:

$$\mu_{A \cap B}(x) = \mu_A(x) \wedge \mu_B(x) = \min(\mu_A(x), \mu_B(x)) \quad (30)$$

Definicja (suma zbiorów rozmytych) 2.9.

Sumą zbiorów rozmytych $A, B \subseteq X$ jest zbiór rozmyty $A \cup B$ o funkcji przynależności, dla każdego $x \subseteq X$:

$$\mu_{A \cup B}(x) = \mu_A(x) \vee \mu_B(x) = \max(\mu_A(x), \mu_B(x)) \quad (31)$$

Suma i przecięcie zbiorów rozmytych są uogólnieniem odpowiadających im operacji na zbiorach nierozmytych. Jednak operacje maksimum i minimum nie są jedynymi sposobami rozszerzenia tych operacji w dziedzinie rozmytej. Ogólnie, do określenia funkcji przynależności sumy i przecięcia zbiorów rozmytych można stosować funkcje zwane *normami trójkątnymi*: t-normą i s-normą [łę08], [yag95].

$$\mu_{A \cap B}(x) = \mathbf{t}(\mu_A(x), \mu_B(x)) = \mathbf{t}(A, B) \quad (32)$$

gdzie funkcja \mathbf{t} jest dowolną t-normą, $\mathbf{t}: [0,1] \times [0,1] \rightarrow [0,1]$.

$$\mu_{A \cup B}(x) = \mathbf{s}(\mu_A(x), \mu_B(x)) = \mathbf{s}(A, B) \quad (33)$$

gdzie funkcja \mathbf{s} jest dowolną s-normą, $\mathbf{s}: [0,1] \times [0,1] \rightarrow [0,1]$.

Normy trójkątne zostały szerzej opisane w kolejnym podrozdziale 2.3.3.

Definicja (dopełnienie zbioru rozmytego) 2.10.

Dopełnieniem zbioru rozmytego $A \subseteq X$ jest zbiór rozmyty A' o funkcji przynależności, dla każdego $x \subseteq X$:

$$\mu_{A'}(x) = 1 - \mu_A(x) \quad (34)$$

W przypadku, gdy funkcje przynależności μ_A, μ_B zbiorów rozmytych A, B są funkcjami charakterystycznymi, powyższe definicje zgadzają się z definicjami w klasycznym ujęciu boolowskim.

2.3.3. Normy trójkątne w logice rozmytej

Normy trójkątne zostały wprowadzone przez Karla Mengersa w 1942 roku [men42] w problemie uogólnienia nierówności trójkąta z klasycznych przestrzeni metrycznych na probabilistyczne przestrzenie metryczne, w których odległość między obiektami dana jest w postaci rozkładu prawdopodobieństwa, a nie liczbowo. Dziś zastosowania norm trójkątnych są bardzo rozległe.

Z obszarów, w których normy trójkątne odgrywają kluczową rolę, można wymienić teorię zbiorów rozmytych oraz logikę wielowartościową. Teorię norm trójkątnych rozwijali następnie

Schweizer i Sklar, podając zbiór obowiązujących do dzisiaj aksjomatów, a w monografii [sch83] z 1983 roku podali szereg istotnych wyników dotyczących tej teorii. Obecny stan badań nad teorią norm trójkątnych można znaleźć w pracach Gotwalda [got99], [got01], Klementa, Mesiara i Papa [kle00], [kle04], Lowena [low96], Nguyena [ngu97] oraz Webera [web83]. W niniejszym rozdziale przedstawione zostaną podstawowe pojęcia i fakty dotyczące teorii norm trójkątnych szczególnie istotne z punktu widzenia dalszych rozważań.

Poniżej przedstawiono definicje dla t-normy i s-normy na podstawie [yag95], [łes08], [kac86].

Definicja 2.11.

Dwuargumentową operację $\mathbf{t}: [0,1] \times [0,1] \rightarrow [0,1]$ nazywamy normą trójkątną (w skrócie t-normą), gdy dla każdego $a, b, c, d \in [0,1]$ spełnia ona następujące warunki:

- 1) $a \mathbf{t} b = b \mathbf{t} a$ (przemienność),
- 2) $(a \mathbf{t} b) \mathbf{t} c = a \mathbf{t} (b \mathbf{t} c)$ (łączność),
- 3) jeśli $(a \leq b \text{ i } c \leq d)$, to $a \mathbf{t} c \leq b \mathbf{t} d$ (monotoniczność),
- 4) $a \mathbf{t} 1 = a$ (1 – element neutralny; warunek brzegowy).

Definicja 2.12.

Dwuargumentową operację $\mathbf{s}: [0,1] \times [0,1] \rightarrow [0,1]$ nazywamy konormą trójkątną (w skrócie s-normą), gdy s spełnia warunki 1)–3) oraz 0 jest jej elementem neutralnym, tzn. dla każdego $a \in [0,1]$:

- 5) $a \mathbf{s} 0 = a$ (0 – element neutralny; warunek brzegowy).

W tabelach 2.7. i 2.8. przedstawiono wybrane operacje s-norm i t-norm.

Tabela 2.7. Wybrane operatory s-normy [yag95].

Nazwa operatora	Wzór
Maksimum (max)	$a \mathbf{s} b = \max(a, b)$
Suma algebraiczna	$a \mathbf{s}_a b = a + b - a \cdot b$
Suma Hamachera	$a \mathbf{s}_H b = \frac{a + b - 2a \cdot b}{1 - a \cdot b}$
Suma Einsteina	$a \mathbf{s}_E b = \frac{a + b}{1 + a \cdot b}$
Suma drastyczna	$a \mathbf{s}_d b = \begin{cases} \max(a, b), & \text{gdy } \min(a, b) = 0 \\ 1, & \text{gdy } \min(a, b) \neq 0 \end{cases}$
Suma ograniczona Łukasiewicza	$a \mathbf{s}_L b = \min(1, a + b)$

Tabela 2.8. Wybrane operatory t-normy [yag95].

Nazwa operatora	Wzór
Minimum (min)	$a \mathbf{t} b = \min(a, b)$
Iloczyn algebraiczny	$a \mathbf{t}_a b = a \cdot b$
Iloczyn Hamachera	$a \mathbf{t}_H b = \frac{a \cdot b}{a + b - 2a \cdot b}$
Iloczyn Einsteina	$a \mathbf{t}_E b = \frac{a \cdot b}{2 - (a + b - 2a \cdot b)}$
Iloczyn drastyczny	$a \mathbf{t}_d b = \begin{cases} \min(a, b), & \text{gdy } \max(a, b) = 1 \\ 0, & \text{gdy } \max(a, b) \neq 1 \end{cases}$
Iloczyn ograniczony Łukasiewicza	$a \mathbf{t}_L b = \max(0, a + b - 1)$

Jeśli t-normy i s-normy są używane do obliczeń sumy i przecięcia funkcji przynależności zbiorów rozmytych, to powyższe oznaczenia przyjmują postać, dla zbiorów rozmytych $A, B \subseteq X$, i funkcji przynależności $a = \mu_A(x)$, $b = \mu_B(x)$:

$$\mu_{A \cup B}(x) = a \mathbf{s} b \quad (35)$$

$$\mu_{A \cap B}(x) = a \mathbf{t} b \quad (36)$$

Dla wygody normy trójkątne (t-normy i s-normy) nazwiemy wspólnie operacjami trójkątnymi (t-operacjami). Z definicji 2.11. i 2.12. wynikają następujące własności norm:

- 1) $a \mathbf{t} 0 = 0, a \mathbf{s} 1 = 1$,
- 2) $\mathbf{t}_d \leq \mathbf{t} \leq \wedge \leq \vee \leq \mathbf{s} \leq \mathbf{s}_d$, gdzie relacja częściowego porządku zdefiniowana jest punktowo,
- 3) $a \mathbf{t} a \leq a \leq a \mathbf{s} a$,
- 4) $a \mathbf{t} b = 1$ wtw $a = b = 1$,
- 5) $a \mathbf{s} b = 0$ wtw $a = b = 0$,
- 6) $(\forall a \in [0,1]: a \mathbf{t} a = a)$ wtw $\mathbf{t} = \wedge$,
- 7) $(\forall a \in [0,1]: a \mathbf{s} a = a)$ wtw $\mathbf{s} = \vee$,
- 8) $(\forall a, b, c \in [0,1]: a \mathbf{t} (b \mathbf{s} c) = (a \mathbf{t} b) \mathbf{s} (a \mathbf{t} c))$ wtw $\mathbf{s} = \vee$,
- 9) $(\forall a, b, c \in [0,1]: a \mathbf{s} (b \mathbf{t} c) = (a \mathbf{s} b) \mathbf{t} (a \mathbf{s} c))$ wtw $\mathbf{t} = \wedge$,

gdzie wtw oznacza wtedy, i tylko wtedy.

Jeżeli \mathbf{t} jest t-normą, to operację \mathbf{t}^* nazywać będziemy s-normą sprzężoną, jeżeli:

$$\forall a, b \in [0,1]: a \mathbf{t}^* b = 1 - (1 - a) \mathbf{t} (1 - b) \quad (37)$$

Jeżeli s jest s-normą, to t-norma sprzężona $s *$ określona jest wzorem:

$$\forall a, b \in [0,1]: a s * b = 1 - (1 - a) s (1 - b) \quad (38)$$

Podstawowymi przykładami operacji sprzężonych są pary t-operacji wymienionych w tabelach 2.7. i 2.8. Inne ważne pary operacji sprzężonych zostały przedstawione poniżej (tabela 2.9.).

Tabela 2.9. Wybrane t-operacje z parametrem [yag95].

Nazwa operatora	t-operacje
t-operacje Schweizera z parametrem $\lambda > 0$	$a t_S b = (0 \vee (a^\lambda + b^\lambda - 1))^{1/\lambda}$ $a s_S b = 1 - (0 \vee ((1 - a^\lambda) + (1 - b^\lambda) - 1))^{1/\lambda}$
t-operacje Yagera z parametrem $\lambda > 0$	$a t_Y b = 0 \vee (1 - ((1 - a)^\lambda + (1 - b)^\lambda)^{1/\lambda})$ $a s_Y b = 1 \wedge (a^\lambda + b^\lambda)^{1/\lambda}$
t-operacje Franka z parametrem $\lambda > 0, \lambda \neq 1$	$a t_F b = \log_\lambda \left(1 + \frac{(\lambda^a - 1)(\lambda^b - 1)}{\lambda - 1} \right)$ $a s_F b = 1 - \log_\lambda \left(1 + \frac{(\lambda^{1-a} - 1)(\lambda^{1-b} - 1)}{\lambda - 1} \right)$
t-operacje Webera z parametrem $\lambda > -1$	$a t_W b = 0 \vee \left(\frac{a + b - 1 + \lambda ab}{1 + \lambda} \right)$ $a s_W b = 1 \wedge \left(\frac{(1 + \lambda)(a + b) - \lambda ab}{1 + \lambda} \right)$

Wiele innych przykładów t-operacji można znaleźć w pracach [low96] i [web83]. Z teoretycznego i praktycznego punktu widzenia interesujące jest zachowanie się wyżej wymienionych sparametryzowanych rodzin t-operacji w zależności od zmieniającego się parametru λ . Szczególnie interesujące są własności graniczne t-operacji Franka:

$$a t_{F,\lambda} b_\lambda \rightarrow_0 a \wedge b, a s_{F,\lambda} b_\lambda \rightarrow_0 a \vee b \quad (39)$$

$$a t_{F,\lambda} b_\lambda \rightarrow_1 a t_a b, a s_{F,\lambda} b_\lambda \rightarrow_1 a s_a b \quad (40)$$

$$a t_{F,\lambda} b_\lambda \rightarrow_\infty a t_\perp b, a s_{F,\lambda} b_\lambda \rightarrow_\infty a s_\perp b \quad (41)$$

Szeroki zestaw własności granicznych dla innych t-operacji można znaleźć w pracy [low96].

2.3.4. Wnioskowanie przybliżone w systemach wyszukiwania wiedzy

W logice klasycznej operujemy na zdaniach logicznych, które mogą być prawdziwe lub fałszywe i nie przyjmują stanów pośrednich. Przy wykorzystaniu logiki rozmytej mamy do czynienia z odmienną sytuacją.

Zbiory rozmyte pozwalają w sposób usystematyzowany modelować pojęcia nieprecyzyjne, jakimi ludzie posługują się na co dzień. Przykładami mogą być wyrażenia: „wysoka

temperatura”, „duża prędkość” czy „młody człowiek”. Również podczas wnioskowania i podejmowania decyzji ludzie wykorzystują takie rozmyte pojęcia. Rozumujemy np.: „Jeśli droga jest bardzo kręta i widoczność jest bardzo słaba, to należy jechać bardzo powoli”. Lotfi Zadeh stworzył aparat matematyczny umożliwiający formalizację tego typu rozumowania [zad75], [zad99].

Jedną z reguł wnioskowania logiki klasycznej jest reguła *modus ponens* określona przez następujący schemat wnioskowania [łes08], [yag95]:

Przesłanka: A
 Implikacja: $A \rightarrow B$

 Wniosek: B

Z prawdziwości implikacji wiadomo, że zawsze ilekroć zachodzi zdanie A (A przyjmuje wartość „prawda”), tylekroć zachodzi również zdanie B . Jeżeli w konkretnej sytuacji zachodzi A , to można być pewnym, że zachodzi również B .

Podstawową regułą wnioskowania *modus ponens* w klasycznej logice dwuwartościowej rozszerzono na logikę rozmytą. L. A. Zadeh sformułował wnioskowanie przybliżone (ang. *approximate reasoning*) oraz „rozmytą” regułą wnioskowania zwaną **uogólnioną regułą modus ponens** (ang. *generalized modus ponens*) [zad75]. Schemat wnioskowania dla tej reguły wygląda następująco:

Przesłanka: x jest A°
 Implikacja: **Jeśli** x jest A , **to** y jest B

 Wniosek: y jest B°

gdzie:

x, y – zmienne lingwistyczne,

A, A°, B, B° – zbiory rozmyte określające wartości zmiennych lingwistycznych.

Zmienne lingwistyczne przyjmują jako swoje wartości słowa lub zdania wypowiedziane w języku naturalnym, takie jak „mała prędkość”, „umiarkowana temperatura”, „duże ciśnienie” lub wartości liczbowe. Do stwierdzeń wyrażonych przez zmienne lingwistyczne słownie można przyporządkować pewne zbiory rozmyte w celu formalizacji tych wyrażen [rut97], [rut05].

Wynik wnioskowania przybliżonego jest zależny od implikacji zapisanej w postaci reguły *jeżeli-to*. Rozmyta reguła warunkowa (ang. *fuzzy conditional rules*) „**Jeśli** x jest A , **to** y jest B ”, rozumiana jest jako relacja zdefiniowana w przestrzeni $X \times Y$. Relacja ta nazywana jest *rozmytą implikacją*, określoną jako:

$$R = (A \rightarrow B) \tag{42}$$

Takie podejście nazywane jest *logiczną interpretacją* reguł *jeżeli-to* [łes08].

Istnieje również drugie stosowane podejście, zwane *koniunkcyjną interpretacją* reguł *jeżeli-to*. Wtedy reguła jest interpretowana jako relacja rozmyta R , gdzie:

$$R = (A \cap B) \quad (43)$$

W powyższym wzorze przecięcie zbiorów jest realizowane przez t-normę, przy czym popularne reguły wyznaczania funkcji przynależności rozmytej implikacji to reguła typu minimum (Mamdaniego) i reguła typu iloczyn (Larsena).

Koniunkcyjna interpretacja prowadzi do prostszych algorytmów wnioskowania, a co się z tym wiąże, prowadzi do mniejszych nakładów obliczeniowych, jednak jest mniej intuicyjna z punktu widzenia logiki [łes08].

Rozmyta implikacja została bardziej ogólnie zdefiniowana przez J. Fodora jako łącznik o pewnych własnościach ogólnych (koniecznych) i szczegółowych (zalecanych) [łes08]. Według niego, *implikacja rozmyta* jest funkcją $I: [0,1] \times [0,1] \rightarrow [0,1]$ spełniającą następujące warunki (dla $x, y, z \in [0,1]$):

- 1) jeżeli $x \leq z$, to $I(x, y) \geq I(z, y)$,
- 2) jeżeli $y \leq z$, to $I(x, y) \leq I(x, z)$,
- 3) $I(0, y) = 1$,
- 4) $I(x, 1) = 1$,
- 5) $I(1, 0) = 0$.

Poniżej w tabeli 2.10. podano wybrane implikacje rozmyte na podstawie [łes08], [kac01].

Tabela 2.10. Wybrane implikacje rozmyte [łes08].

Nazwa	Wzór
Łukasiewicza	$\min(1 - \mu_A(x) + \mu_B(x), 1)$
Fodora	$\begin{cases} 1, \mu_A(x) \leq \mu_B(x) \\ \max(1 - \mu_A(x), \mu_B(x)), \mu_A(x) > \mu_B(x) \end{cases}$
Reichenbacha	$1 - \mu_A(x) + \mu_A(x)\mu_B(x)$
Kleene'a-Dienes	$\max(1 - \mu_A(x), \mu_B(x))$
Zadeha	$\mu_{A \rightarrow B}(x) = \max\{1 - \mu_A(x), \min(\mu_A(x), \mu_B(x))\}$
Goguena	$\min\left(\frac{\mu_B(x)}{\mu_A(x)}, 1\right)$
Gödela	$\begin{cases} 1, \mu_A(x) \leq \mu_B(x) \\ \mu_B(x), \mu_A(x) > \mu_B(x) \end{cases}$
Reschera	$\begin{cases} 1, \mu_A(x) \leq \mu_B(x) \\ 0, \mu_A(x) > \mu_B(x) \end{cases}$

W logice rozmytej wnioskowanie realizuje się przez złożenie rozmytego zbioru A° oraz rozmytej implikacji $(A \rightarrow B)$. W wyniku tego złożenia otrzymamy zbiór rozmyty B° , który stanowi wniosek: $B^\circ = A^\circ \circ (A \rightarrow B)$. Złożenie to realizowane jest następująco [łes08], [zad99]:

$$\mu_{B^\circ}(y) = \sup_{x \in X} \{\mu_{A^\circ}(x) \bullet \mu_{A \rightarrow B}(x, y)\} \quad (44)$$

gdzie \bullet oznacza pewną t-normę. Jeśli przykładowo t-norma jest typu min, otrzymuje się regułę:

$$\mu_{B^\circ}(y) = \sup_{x \in X} \{\min[\mu_{A^\circ}(x), \mu_{A \rightarrow B}(x, y)]\} \quad (45)$$

W procesie wyszukiwania informacji o wystarczającym dla ekspertów stopniu pewności, na podstawie informacji niepewnej, ważne jest wnioskowanie przybliżone wykorzystujące następująco określoną implikację rozmytą [tsu09]. Niech operacja \bullet jest pewną t-normą. Każda t-norma określa swoją unikalną operację implikacji \rightarrow , zwaną *residuum* lub *R-implikacją* ([bac13], [tsu09]) wyznaczającą miarę podobieństwa implikacji formuł (dla wszystkich $x, y, z \in [0, 1]$):

$$z \leq (x \rightarrow y) \text{ wtw } x \bullet z \leq y \quad (46)$$

lub

$$(x \rightarrow y) = \sup\{t \in [0, 1] : x \bullet t \leq y\} \quad (47)$$

Przykładową implikacją typu residuum jest implikacja Łukasiewicza. Pokazano w pracach [tsu09], [bac08], że każda rozmyta implikacja określona za pomocą residuum spełnia uogólnioną regułę modus ponens. Ponadto zachodzi twierdzenie:

Twierdzenie (o prawdziwości residuum) [bac13] 2.1.

Dla dowolnych $x, y \in [0, 1]$:

$$x \leq y \text{ wtw } (x \rightarrow y) = 1 \quad (48)$$

Dowód.

Jeżeli $x \leq y$, to ze wzoru (46) i definicji t-normy, dla $z = 1$ mamy $1 \leq (x \rightarrow y) \text{ wtw } x \leq y$, a więc $1 \leq (x \rightarrow y)$, co oznacza że $(x \rightarrow y) = 1$. Natomiast, jeśli $(x \rightarrow y) = 1$, to wzór (46), wobec definicji t-normy, przekształca równoważność $z \leq 1 \text{ wtw } x \bullet z \leq y$ zachodzącą także dla $z = 1$, co oznacza, że $1 \leq 1 \text{ wtw } x \leq y$, a więc $x \leq y$.

We wnioskowaniu przybliżonym, wykorzystującym residuum, jeśli implikacja jest pewna, a do poprzednika implikacji, posiadającego stopień rozmycia x , występuje zaufanie ekspertów, to do każdej wywnioskowanej informacji, o większym od x stopniu rozmycia, będzie występowało zaufanie ekspertów. Taki stan rzeczy, opisany w następnym rozdziale jako paradoks rozmytej dokładności, oczekiwany jest zwykle podczas wyszukiwania informacji. Dlatego słuszne będzie stosowanie implikacji residuum we wnioskowaniu występującym podczas wyszukiwania informacji.

2.4. Postulaty znaczeniowe logiki wyszukiwania informacji

W literaturze zostały szeroko opisane systemy wyszukiwania informacji (ang. *Information Retrieval Systems*) [bae99], [man04], [ceg04], [ceg10]. W niniejszym podrozdziale przedstawiono logikę wyszukiwania informacji IRL (ang. *Information Retrieval Logic*)

[bry13a], [bry13b]. Logika *IRL* może zostać wykorzystana w systemie wyszukiwania informacji, dla sieci semantycznych Web. Logika wyszukiwania informacji pozwala na wyszukiwanie wiedzy w sieci semantycznej, opisanej za pomocą rozmytej logiki opisowej (ang. *Fuzzy Description Logic – fuzzyDL*) [str98], [str01], [bob08], [pan08], [sim08].

W podrozdziale 2.4.1. zostało omówione uzyskiwanie dokładnych wyników wyszukiwania informacji. Należy zauważyć, że w rezultacie operacji wyostrażania można uzyskać kilka wyników, które będą należały do zakresu ufności. Jednak nawet w takiej sytuacji można uzyskać dokładną odpowiedź w wyniku wyszukiwania. Na tej podstawie sformułowano dokładność wyniku rozmytego wyszukiwania informacji (rozdz. 2.4.1.) oraz postulaty rozmytości logiki IRL (rozdz. 2.4.3.).

2.4.1. Dokładność wyniku rozmytego wyszukiwania informacji

Współcześnie wyszukiwanie informacji (ang. *Information Retrieval – IR*) w sieci semantycznej Web zazwyczaj oznacza szukanie wiarygodnego źródła tej informacji. Jak dotąd systemy wyszukiwania informacji i systemy semantycznych relacji wskazywały jedynie na semantycznie najbliższe źródła szukanej informacji. Nie zawsze jednak to ma miejsce. Często szukając informacji o jakimś obiekcie, można uzyskać wiedzę niepewną opisaną w odpowiednim języku (języku ontologii), a pomimo tego ta niepewność może prowadzić do jednoznacznego ustalenia źródeł wiedzy o tym obiekcie. W ten sposób otrzymuje się zgodność z opisem modelu obiektu (zgodność z tezauresem). Opisana sytuacja nazywana jest tu **paradoksem rozmytej dokładności** (ang. *fuzzy disambiguation*) wyszukiwania informacji. Wymienione poniżej metody oraz ich implementacje (np. w OWL) pomijają występowanie tego paradoksu. Są to metody:

- eksploracji danych (ang. *data mining*) [han01], [lar05], [mar07],
- ujednoznacznienia słów w wyszukiwaniu informacji (ang. *word sense disambiguation in information retrieval*) [bae99], [ceg06], [ceg10], [jay11], [sto03], [san94],
- relacji semantycznych (ang. *semantic relation*) [ceg06], [ceg10],
- teorii zbiorów rozmytych i logiki rozmytej (ang. *fuzzy sets theory and fuzzy logic*) [haj98],
- rozmytej logiki opisowej (ang. *fuzzy description logic*) [bob07], [bob09], [str98], [sim10].

Poniżej podane są dwa przykłady ilustrujące ten paradoks:

Przykład 2.13.

Dane użytkownika żelazka: tkanina Y po wyprasowaniu skurczyła się i pofalowała.

Dane eksperta: niektóre nici tkaniny Y kurczą się tylko w parze wodnej w temperaturze wrzenia wody, 100⁰C, ale nie w wodzie w stanie wrzenia.

Dane użytkowników żelazka są niepewne, gdyż nie wiadomo, jaki program prasowania był stosowany. Jednak z tych niepewnych danych wynika dokładna informacja: przy prasowaniu

został włączony program wytwarzania przez żelazko pary wodnej. Wnioskowanie to wykorzystuje daną eksperta, wyklucza bowiem sytuację, że przy prasowaniu tkanina była tylko nawilżona i podgrzewana przez żelazko.

Przykład 2.14.

Czy „Ralf Möller”, kulturysta i aktor telewizyjny, oraz „Ralf Möller”, profesor na Politechnice w Hamburgu, to ta sama osoba?

Dane w zasobach sieci Web: Ralf Möller, aktor, urodził się w 1959 roku; Ralf Möller, profesor, urodził się, gdy kanclerzem Niemiec był Ludwig Erhard (z historii można się dowiedzieć, że były to lata 1963-1966).

Dane eksperta: dowolny atrybut dla danej osoby posiada tylko jedną wartość.

Dane w zasobach Web są wieloznaczne i niepewne. Nazwami osób są: $imie_nazwisko(osoba1) = „Ralf Möller”$ dla aktora, $imie_nazwisko(osoba2) = „Ralf Möller”$ dla profesora. Czy $osoba1 = osoba2$? Uzupełniając niepewną wiedzę o wiedzę eksperta: $data_urodzenia(osoba1) = 1959$, $data_urodzenia(osoba2) > 1963$, można wywnioskować, że $osoba1 \neq osoba2$. Wynik wnioskowania jest informacją dokładną.

Wyszukiwanie informacji w sieciach semantycznych Web sprowadza się do znajdowania egzemplarzy danych będących:

- wartościami jednoargumentowych atrybutów, czyli danych reprezentujących wiedzę o pewnych cechach lub rodzajach obiektów,
- wartościami dwuargumentowych atrybutów, czyli danych reprezentujących wiedzę o własnościach obiektów lub relacjach pomiędzy dwoma obiektami.

W pierwszym przypadku dane nazywane są **konceptami**, a w drugim – **rolami**. Do opisu konceptów i ról wykorzystywany jest język **logiki opisowej** (ang. *Description Logic – DL*) [baa03]. Język logiki DL opisujący koncepty i role może zostać rozszerzony o pewne formuły logiki pierwszego rzędu. W tak rozszerzonym języku tworzy się **tezaurus** opisujący wzorcowe koncepty i role. Natomiast **ontologią** jest język opisujący obserwowane koncepty i role, czyli wyszukiwane dane. Jeśli dla szukanych danych z ontologii oraz z zaleceń (kryteriów i wiedzy) ekspertów będzie wynikać stopień zgodności tych danych z pewnymi danymi z tezaury, to ten związek nazywany będzie **residuum**.

Wyszukiwanie danych w zasobach internetowych (w zbiorze adresów internetowych) jest więc pewną ich interpretacją określającą stopień podobieństwa tych danych do danych tezaury. Stopień ten jest liczbą z przedziału $[0,1]$ i jest miarą przynależności szukanych danych do zbioru danych dostępnych w Internecie. Uwzględnia on także semantyczną strukturę tych zasobów, określoną za pomocą sieci semantycznych. Koncepty i role z ontologii interpretowane są jako zbiory rozmyte, określone na przestrzeni adresów zasobów wiedzy w sieci Web oraz par tych adresów. W ten sposób dokonywane jest **rozmywanie** reprezentacji wiedzy [bry12a], a ustalenie residuum pozwala dokonać jej **wyostrzenia** [bry12a]. Ustalenie residuum polega na tym, że dla danego zapytania o wiedzę wskazuje się wiarygodny dla

ekspertów zbiór adresów sieci Web reprezentujących tę wiedzę. Zbiory interpretacji tworzące residuum będą dalej traktowane jako **wyszukiwanie informacji**.

W trakcie wyszukiwania informacji stosowana jest następująca reguła wyszukiwania:

- 1) Pytanie o informację jest wyrażeniem ontologii zbudowanym ze słów i formuł tezauryusa.
- 2) Niech jedynym zasobem, z którego pozyskiwane są dane, będzie strona internetowa tezauryusa, co oznacza, że zbiór szukanych adresów sieci semantycznej Web jest pusty; niech ma to miejsce dla wszystkich przyjętych interpretacji (metod) ustalających stopień podobieństwa pozyskanych egzemplarzy danych do egzemplarzy danych tezauryusa; w takim przypadku stopień podobieństwa pozyskanych egzemplarzy danych w innych zasobach wiedzy jest zerowy.
- 3) Jeśli nie ma miejsca pkt 2), to akceptowany przez ekspertów zbiór adresów zasobów wiedzy diagnostycznej reprezentuje wiedzę najbliższą znaczeniowo szukanej informacji.

Tę regułę wyszukiwania informacji nazywa się **regułą residuum**. Na tej podstawie można wprowadzić **logikę wyszukiwania informacji IRL**, która stosuje tę regułę.

Można zauważyć, że proces wyszukiwania informacji jest procesem pomiarowym, mierzącym dostępność danych w przestrzeni adresów internetowych. Stosowanie reguły residuum jest zarazem dokonywaniem diagnozy dostępności danych w Internecie. W tym sensie proces wyszukiwania informacji diagnostycznej jest także procesem diagnostycznym. Naturalne jest rozszerzenie procesu diagnostycznego o dane pozyskiwane w sieci semantycznej Web.

2.4.2. Język logiki IRL

Sieć semantyczna może być rozpatrywana jako indeksowany graf skierowany [kow79a]. Struktura sieci semantycznej została przedstawiona na rysunku 2.5. Węzły (wierzchołki) tego grafu utożsamiane są ze stanami wyszukiwania wiedzy o jakimś obiekcie, czyli z wynikami pozyskiwania danych diagnostycznych w zasobach internetowych (por. [moc02]). Natomiast krawędzie (ramiona) utożsamiane są ze stanami identyfikującymi wiedzę o relacjach pomiędzy wskazanymi przez węzły grafu obiektami. Krawędzie określają wyniki wyszukiwania danych o związkach pomiędzy diagnozowanymi wielkościami. Zarówno węzły, jak i krawędzie grafu opisane są jakimiś terminami oznaczającymi obiekty i relacje. Opisy węzłów są nazwami indywiduów, opisy krawędzi jednowęzłowych są nazwami konceptów (pojęć), a opisy krawędzi dwuwęzłowych są nazwami ról spełnianych przez opisywane obiekty.

Opisy indywiduów (t_1, t_2, \dots) , konceptów (C_1, C_2, \dots) i ról (R_1, R_2, \dots) składają się na terminologię. Gdy związek pomiędzy obiektami reprezentowany jest przez sieć semantyczną, to nazywamy go asercją. Asercję, że „ t_1 jest C_1 ”, zapisujemy w postaci „ $t_1: C_1$ ”, natomiast asercję o relacji nazywanej R_1 pomiędzy obiektami nazywanymi t_1 i t_2 , zapisujemy w postaci „ $(t_1, t_2): R_1$ ” (rysunek 2.5.).

W kontekście badań sieci semantycznej [bob08], [gal04], [pan08], [sim08], [sim10] reprezentacja wiedzy w sieci semantycznej może być określona w języku atrybutowym (ang. *attributive language* – AL) logiki opisowej [baa03], [cer10]. Wtedy wiedza jest reprezentowana przez: koncepty **TBox**, role **RBox** oraz asercje **ABox**. Sieć semantyczna może zostać rozszerzona o krawędzie ustalające zależności pomiędzy konceptami lub rolami. Opisy tych zależności nazywane są **aksjomatami**.

Sieć semantyczna może również zostać rozszerzona o opis rozmytości elementów w niej występujących, czyli: konceptów, ról, asercji i aksjomatów. Rozmyta logika opisowa (ang. *Fuzzy Description Logic* – *fuzzyDL*) [str98], [str01] pozwala na opisanie elementów sieci semantycznej oraz przypisanie im pewnej funkcji przynależności, określającej niepewny charakter opisywanej wiedzy. Poniżej określono syntaktykę języka AL dla logiki wyszukiwania informacji IRL (ang. *Information Retrieval Logic*), analogicznie do języka *fuzzyDL*.

Syntaktyka TBox

Do zbioru nazw konceptów i ról należą następujące nazwy:

- \top (*Top*) – koncept uniwersalny oraz rola uniwersalna,
- \perp (*Bottom*) – koncept pusty oraz rola pusta.

Niech C, D będą nazwami konceptów, R będzie nazwą roli, a m – symbolem **modyfikatora**, wtedy **złożonymi** konceptami są:

- $\neg C$ – **negacja konceptu** – wyrażenie oznacza wszystkie wystąpienia konceptów niebędące wystąpieniami konceptu C ,
- $C \sqcap D$ – **przecięcie, koniunkcja konceptów** C i D – wyrażenie oznacza wszystkie wystąpienia konceptów C i D ,
- $C \sqcup D$ – **suma, alternatywa konceptów** C i D – wyrażenie oznacza wszystkie wystąpienia konceptu C lub konceptu D ,
- $\exists R.C$ – **kwantyfikacja egzystencjalna** – wyrażenie oznacza wszystkie wystąpienia konceptów pozostające w roli R co najmniej raz z wystąpieniem konceptu C ,
- $\forall R.C$ – **kwantyfikacja ogólna** – oznaczająca wszystkie wystąpienia konceptów, które jeżeli pozostają w roli R , to pozostają w tej roli z jakimś wystąpieniem konceptu C ,
- $m(C)$ – **modyfikacja m konceptu** C – oznaczająca koncept będący zmienionym konceptem C przez słowo m , np. m może mieć takie wystąpienia, jak: bardzo, bardziej, najbardziej lub wysoki, wyższy, najwyższy.

Koncepty, które nie są złożone, nazywamy **prostymi**.

Syntaktyka ABox

Dla dowolnych wystąpień konceptów t_1, t_2 , nazwy konceptu C oraz nazwy roli R **asercjami** są wyrażenia postaci „ $t_1 : C$ ”, „ $(t_1, t_2) : R$ ”. Czytamy je: t_1 jest wystąpieniem konceptu C , para (t_1, t_2) jest wystąpieniem roli R .

Syntaktyka aksjomatów TBox

Dla dowolnych nazw konceptów C, D i dowolnej liczby $\alpha \in [0,1]$ **aksjomatami** są wyrażenia postaci:

- $C \sqsubseteq D$ – koncept C jest konceptem D ,
- $C = D$ – koncept C jest identyczny z konceptem D ,
- $\langle C \sqsubseteq D, \alpha \rangle$ – koncept C jest konceptem D w stopniu α ,
- $\langle C = D, \alpha \rangle$ – koncept C jest identyczny z konceptem D w stopniu α .

Syntaktyka aksjomatów RBox

Dla dowolnych nazw ról R_1, R_2 i dowolnej liczby $\alpha \in [0,1]$ **aksjomatami** są wyrażenia postaci:

- $R_1 \sqsubseteq R_2$ – rola R_1 jest rolą R_2 ,
- $R_1 = R_2$ – rola R_1 jest identyczna z rolą R_2 ,
- $\langle R_1 \sqsubseteq R_2, \alpha \rangle$ – rola R_1 jest rolą R_2 w stopniu α ,
- $\langle R_1 = R_2, \alpha \rangle$ – rola R_1 jest identyczna z rolą R_2 w stopniu α .

Syntaktyka formuł

Dowolne asercje i aksjomaty są formułami. Dla dowolnych formuł φ, ϕ , dowolnej zmiennej x oraz liczby $\alpha \in [0,1]$ formułami są:

- 1) $\neg\varphi$ – negacja formuły φ ,
- 2) $\langle \varphi, \alpha \rangle$ – formuła φ prawdziwa w stopniu α ,
- 3) $\exists x. \varphi$ – kwantyfikacja egzystencjalna formuły φ dla zmiennej x ,
- 4) $\forall x. \varphi$ – kwantyfikacja ogólna formuły φ dla zmiennej x ,
- 5) $\varphi \Rightarrow \phi$ – implikacja formuł φ, ϕ ,
- 6) $\varphi \wedge \phi$ – koniunkcja formuł φ, ϕ ,
- 7) $\varphi \vee \phi$ – alternatywa formuł φ, ϕ ,
- 8) $\varphi \Leftrightarrow \phi$ – ekwiwalencja formuł φ, ϕ .

Dla określonej syntaktyki języka logiki IRL można teraz określić postulaty rozmytości, które są potrzebne w procesie wyszukiwania informacji w sieci semantycznej.

2.4.3. Propozycja postulatów rozmytości logiki IRL

Sposób modelowania i reprezentacji niepewnej lub niepełnej wiedzy w sieciach semantycznych Web jest istotnym aspektem logiki wyszukiwania informacji IRL. Istotne jest, aby określić tezaurs, ontologię oraz pewne operacje i reguły w tej logice, co pozwoli na uwzględnienie niepewności wiedzy w trakcie jej wyszukiwania oraz na otrzymanie ostrych wyników wyszukiwania na podstawie niepewnych informacji. W tym celu proponuje się przyjęcie poniższych postulatów (**P1.–P9.**) rozmytości logiki IRL.

P1. Wyróżniony jest tezaurus, zawierający zbiór pewnych wzorcowych terminów oraz formuł języka IRL. Wyrażenia tezaury reprezentują wiedzę z tej samej dziedziny co szukana informacja i znajdują się w dokumencie tekstowym: *dokumencie tezaury*.

P2. Do ontologii należą wszystkie terminy i formuły języka IRL, odnoszące się znaczeniowo do szukanej informacji. W ontologii zawarty jest tezaurus. Wszystkie formuły tezaury zbudowane są z pewnych terminów i asercji, które należą do bazowego zbioru *Tez*. Podobnie wszystkie formuły ontologii zbudowane są z pewnych terminów i asercji, które należą do bazowego zbioru *Ont*. Stopień podobieństwa znaczeniowego wyrażeń ontologii do wyrażeń tezaury wyznaczany jest przez pewien system ekspertowy uwzględniający wiedzę ekspertów.

P3. Przestrzenią IR są adresy zasobów wiedzy w sieci Web, semantycznie odnoszące się do terminów i formuł reprezentujących szukaną informację. Zasoby wiedzy są dokumentami tekstowymi dostępnymi pod tymi adresami.

P4. Wyszukiwanie pewnej informacji polega na wyszukiwaniu dokumentu tekstowego, którego struktura semantyczna, tworzona przez terminy ontologii występujące w dokumencie, jest najbardziej podobna do struktury semantycznej dokumentu tezaury [bry13a]. Jeśli oba dokumenty zawierają egzemplarze wyrażeń, które jednakowo są używane przez agentów w procesie komunikowania się, to te egzemplarze wyrażeń reprezentują tę samą wiedzę. W tym sensie wyszukiwanie informacji jest wyszukiwaniem dokumentu z danymi reprezentującymi tę samą lub najbardziej podobną wiedzę do wiedzy reprezentowanej przez dane opisane w tezausie. Zatem stosowana jest *reguła residuum* (patrz rozdz. 2.4.1.) dla wyszukiwania informacji.

P5. Podczas wyszukiwania informacji przecięcie konceptów reprezentuje wiedzę wspólną dla tych konceptów, czyli wiedzę reprezentowaną przez wspólne wystąpienia tych konceptów. *Wzajemne uzupełnianie się formuł* (ang. *complementarity formulas*) φ, ϕ jest formułą oznaczaną przez $\varphi \& \phi$, reprezentującą wspólną wiedzę reprezentowaną przez te formuły.

P6. W trakcie wyszukiwania informacji, jeżeli $x, y \in [0,1]$ są stopniami podobieństwa semantycznego pewnych wystąpień konceptów, to stopniem wystąpienia przekroju tych konceptów jest liczba $x \bullet y$, gdzie operacja $\bullet: [0,1] \times [0,1] \rightarrow [0,1]$ jest pewną t-normą (rozdz. 2.3.3.).

Podobna sytuacja zachodzi w przypadku, gdy x, y oznaczają stopień semantycznego podobieństwa formuł φ, ϕ do jakiegoś wystąpienia konceptu lub formuł należących do tezaury. Wtedy stopniem uzupełniania się tych formuł $\varphi \& \phi$ jest liczba $x \bullet y$, gdzie operacja $\bullet: [0,1] \times [0,1] \rightarrow [0,1]$ jest pewną t-normą.

T-norma posiada następujące własności (dla wszystkich $x, y, z, x_0, y_0 \in [0,1]$):

$$x \bullet y = y \bullet x \quad (49)$$

$$(x \bullet y) \bullet z = x \bullet (y \bullet z) \quad (50)$$

$$x \leq x_0 \text{ i } y \leq y_0, \text{ to } x \bullet y \leq x_0 \bullet y_0 \quad (51)$$

$$1 \bullet x = x, 0 \bullet x = 0 \quad (52)$$

Każda t-norma określa swoją unikalną implikację \rightarrow (residuum), wyznaczającą miarę podobieństwa implikacji formuł (dla wszystkich $x, y, z \in [0,1]$):

$$z \leq (x \rightarrow y) \text{ wtw } x \bullet z \leq y \quad (53)$$

lub

$$(x \rightarrow y) = \sup\{t \in [0,1] : x \bullet t \leq y\} \quad (54)$$

Implikacja formuł jest formułą podobną semantycznie do pewnej formuły tezauryusa, jeśli stopień podobieństwa jej związku z poprzednikiem implikacji jest najbliższy stopniowi podobieństwa następnika tej implikacji. Oznacza to, że w znalezionym dokumencie tekstowym jest formuła φ , z której wynika pewna formuła ϕ , zawarta w tym dokumencie lub reprezentująca szukaną informację. Gdy ϕ reprezentuje szukaną informację i nie zawiera się w dokumencie, to formuła φ została uzupełniona o formułę θ reprezentującą wiedzę ekspertów. Wtedy ich wzajemne uzupełnianie się φ & θ posiada stopień podobieństwa semantycznego identyczny ze stopniem szukanej formuły ϕ , stopień równy 1. W ten sposób, w przypadku nieostrygo poprzednika implikacji φ , następnik jest wyrażeniem ostrym o stopniu podobieństwa semantycznego równym 1.

P7. Jeśli w dokumencie tekstowym znajduje się koniunkcja formuł $\varphi \wedge \phi$, to zgodnie z klasycznym rachunkiem zdań wynika $\varphi \Rightarrow \phi$. Taką koniunkcję rozpoznaje się, rozpoznając najpierw formułę φ , a następnie uzupełnia się ją o rozpoznanie implikacji $\varphi \Rightarrow \phi$. Zatem koniunkcję $\varphi \wedge \phi$ rozpoznaje się, rozpoznając φ & $(\varphi \Rightarrow \phi)$. Dlatego w dalszym ciągu przyjmuje się, że:

$$\varphi \wedge \phi := \varphi \& (\varphi \Rightarrow \phi) \quad (55)$$

Stąd stopień podobieństwa koniunkcji $\varphi \wedge \phi$ do formuł tezauryusa określa wzór (dla $x, y \in [0,1]$):

$$x \otimes y := x \bullet (x \rightarrow y) \quad (56)$$

gdzie x, y są stopniami podobieństwa formuł φ, ϕ do formuł tezauryusa.

P8. Jeśli w dokumencie tekstowym znajduje się alternatywa formuł $\varphi \vee \phi$, to na podstawie rachunku zdań można przyjąć, że tę alternatywę rozpoznaje się na podstawie następującego przypisania (podstawienia):

$$\varphi \vee \phi := ((\varphi \Rightarrow \phi) \Rightarrow \phi) \wedge ((\phi \Rightarrow \varphi) \Rightarrow \varphi) \quad (57)$$

Stąd stopień podobieństwa alternatywy $\varphi \vee \phi$ do formuł tezauryusa będzie określał wzór (dla $x, y \in [0,1]$):

$$x \oplus y := ((x \rightarrow y) \rightarrow y) \otimes ((y \rightarrow x) \rightarrow x) \quad (58)$$

gdzie x, y są stopniami podobieństwa formuł φ, ϕ do formuł tezauryusa.

P9. Algebra $BL = \langle L, \otimes, \oplus, \bullet, \rightarrow, 0, 1 \rangle$ jest *regularną rezydualną kratą* (ang. *BL-algebra*) zgodnie z definicją przedstawioną w [haj98]. Działania $\otimes, \oplus, \bullet, \rightarrow$ zostały określone zgodnie z postulatami **P6.** – **P8.** Algebra opisana przez $\langle L, \otimes, \oplus, 0, 1 \rangle$ jest kompletną kratą z największym elementem 1 i najmniejszym elementem 0. Natomiast $\langle L, \bullet, 1 \rangle$ jest przemienną podgrupą z jednostkowym elementem 1, to znaczy \bullet jest operacją przemienną, asocjacyjną i zachodzi $1 \bullet x = x$ dla wszystkich x . Poniższe warunki zachodzą (dla wszystkich $x, y, z \in [0, 1]$):

$$z \leq (x \rightarrow y) \text{ wtw } x \bullet z \leq y \quad (59)$$

$$x \otimes y = x \bullet (x \rightarrow y) \quad (60)$$

$$x \oplus y = ((x \rightarrow y) \rightarrow y) \otimes ((y \rightarrow x) \rightarrow x) \quad (61)$$

$$(x \rightarrow y) \oplus (y \rightarrow x) = 1 \quad (62)$$

W algebrze BL można określić operację *dopełnienia* ' oraz równoważności \leftrightarrow :

$$x' := (x \rightarrow 0) \quad (63)$$

$$x \leftrightarrow y := (x \rightarrow y) \bullet (y \rightarrow x) \quad (64)$$

Przy tak określonych postulatach można określić logikę wyszukiwania informacji IRL w sieciach semantycznych Web.

2.5. Logika wyszukiwania informacji w sieciach semantycznych Web

W podrozdziale przedstawiono logikę wyszukiwania informacji IRL w sieciach semantycznych Web. Najpierw dokonano interpretacji operacji rozmywania reprezentacji wiedzy w sieciach semantycznych. Następnie określono semantykę dla rozmytej logiki opisowej. Na tej podstawie oraz podrozdziału 2.4 przedstawiono logikę wyszukiwania informacji oraz interpretację operacji wyostrażania reprezentacji wiedzy w procesie jej wyszukiwania.

Przedstawione w tym podrozdziale rozważania teoretyczne stanowią fundament koncepcji systemu wyszukującego wiedzę w sieciach semantycznych.

2.5.1. Operacja rozmywania w sieci semantycznej

Wyrażenia języka logiki IRL interpretuje się w algebrze $BL = \langle L, \otimes, \oplus, \bullet, \rightarrow, 0, 1 \rangle$ oraz w wybranej uporządkowanej algebrze zbiorów rozmytych, określonej jako: $F = \langle F, \wedge^F, \vee^F, \neg^F, c^F, e^F, 0^F, 1^F, M, F_0 \rangle$. Algebra ta składa się z rodziny zbiorów rozmytych F , wymienionych operacji i funkcji:

- \wedge^F – operacja przecięcia zbiorów rozmytych,
- \vee^F – operacja sumy zbiorów rozmytych,
- \neg^F – operacja dopełnienia zbiorów rozmytych,
- c^F – funkcja zwana **stopniem zawierania się** zbiorów rozmytych [kac01, s. 47],
- e^F – funkcja zwana **stopniem równości** zbiorów rozmytych [kac01, s. 45],

oraz zbiorów:

- 0^F – zbiór rozmyty będący funkcją posiadającą tylko jedną wartość liczbową 0 (*zbiór pusty*),
- 1^F – zbiór rozmyty będący funkcją posiadającą tylko jedną wartość liczbową 1,
- M – zbiór operacji jednoargumentowych zwanych **funkcjami modyfikacji**,
- F_0 – podzbiory z określonej rodziny zbiorów rozmytych.

Operacje sumy i iloczynu zbiorów rozmytych określają odpowiednie t-normy i s-normy [kac86].

Operacje te zdefiniowane są w *regularnej rezydualnej kracie BL* $= \langle L, \otimes, \oplus, \bullet, \rightarrow, 0, 1 \rangle$ następująco (dla dowolnych zbiorów rozmytych $\mu_A, \mu_B \in F$ oraz dowolnych $x \in [0,1]$):

$$(\mu_A \wedge^F \mu_B)(x) = \mu_A(x) \otimes \mu_B(x) \quad (65)$$

$$(\mu_A \vee^F \mu_B)(x) = \mu_A(x) \oplus \mu_B(x) \quad (66)$$

$$c^F(\mu_A, \mu_B)(x) = \mu_A(x) \rightarrow \mu_B(x) \quad (67)$$

$$e^F(\mu_A, \mu_B)(x) = \mu_A(x) \leftrightarrow \mu_B(x) \quad (68)$$

$$(\neg^F \mu_A)(x) = \mu_A(x) \rightarrow 0 \quad (69)$$

Niech X jest zbiorem wszystkich adresów zasobów wiedzy (egzemplarzy danych), których dotyczy wiedza reprezentowana w sieci semantycznej Web, a $X \times X$ jest zbiorem wszystkich uporządkowanych par elementów zbioru X .

Do dalszych rozważań przyjęto, że interpretacja $I = (F^I)$ spełnia następujące warunki:

F1. Wystąpieniom konceptów t przyporządkowuje się pewne wartości $t^I \in X$, a wystąpieniom par (t_1, t_2) przyporządkowuje się pary $(t_1^I, t_2^I) \in X \times X$.

Najczęściej wystąpienia konceptów są utożsamiane z egzemplarzami danych. Egzemplarze te są uważane przez informatyków za obiekty. Tutaj przestrzeń $X \cup X \times X$ jest zbiorem adresów zasobów sieci Web i par tych adresów, adresów wszystkich dokumentów zawierających rozważane egzemplarze danych. Na przykład konkretny napis w danym miejscu ekranu monitora komputera jest egzemplarzem pewnej danej, wskazanej przez pewien adres tego zasobu wiedzy. Egzemplarzami danych są też ustalenia konkretnych związków pomiędzy danymi, wskazane przez pary adresów tych danych w sieci semantycznej Web.

F2. Nazwom konceptów C przyporządkowuje się zbiory rozmyte $C^I: X \cup X \times X \rightarrow [0,1]$, takie, że dla każdego $x, y \in X$, $C^I(x, y) = C^I((x, y)) = C^I(y)$.

F3. Nazwom roli R przyporządkowuje się zbiory rozmyte $R^I: X \cup X \times X \rightarrow [0,1]$, równe 0 dla argumentów ze zbioru X .

F4. Modyfikatorom m przyporządkowuje się funkcje $m^I: [0,1] \rightarrow [0,1]$, gdzie $m^I \in M$.

F5. Formułom φ , w tym asercjom oraz aksjomatom, przyporządkowuje się pewną, określoną dalej, liczbę $\varphi^I \in [0,1]$.

2.5.2. Semantyka logiki IRL

Poniżej przedstawiono semantykę logiki wyszukiwania informacji dla konceptów, asercji, aksjomatów oraz propozycję semantyki formuł.

Semantyka konceptów TBox

Dla dowolnych $x \in X$ nazw konceptów C, D , nazwy roli R oraz modyfikatora m :

- F6.** $\top^I(x) = 1$;
- F7.** $\perp^I(x) = 0$;
- F8.** $(\neg C)^I(x) = (\neg^F C^I)(x)$;
- F9.** $(C \sqcap D)^I(x) = (C^I \wedge^F D^I)(x)$;
- F10.** $(C \sqcup D)^I(x) = (C^I \vee^F D^I)(x)$;
- F11.** $(\exists R. C)^I(x) = \sup_{y \in X} \{(R^I \wedge^F C^I)(x, y)\}$;
- F12.** $(\forall R. C)^I(x) = \inf_{y \in X} \{(\neg^F R^I \vee^F C^I)(x, y)\}$;
- F13.** $(m(C))^I(x) = m^I(C^I(x))$.

Semantyka asercji ABox

Dla dowolnych wystąpień konceptów t_1, t_2 , konceptu C , roli R :

- F14.** $(t: C)^I = C^I(t^I)$;
- F15.** $((t_1, t_2): R)^I = R^I(t_1^I, t_2^I)$.

Semantyka aksjomatów

Dla dowolnych konceptów C, D i ról R_1, R_2 :

- F16.** $(C \sqsubseteq D)^I = c^F(C^I, D^I)$;
- F17.** $(R_1 \sqsubseteq R_2)^I = c^F(R_1^I, R_2^I)$;
- F18.** $(C = D)^I = e^F(C^I, D^I)$;
- F19.** $(R_1 = R_2)^I = e^F(R_1^I, R_2^I)$;

Propozycja semantyki formuł

Dla dowolnych formuł φ, ϕ i stopni $\alpha \in [0,1]$:

- F20.** $(\neg \varphi)^I = \varphi^I \rightarrow 0$;
- F21.** $(\varphi \wedge \phi)^I = \varphi^I \otimes \phi^I$;
- F22.** $(\varphi \vee \phi)^I = \varphi^I \oplus \phi^I$;
- F23.** $(\varphi \Rightarrow \phi)^I = \varphi^I \rightarrow \phi^I$;
- F24.** $(\varphi \Leftrightarrow \phi)^I = \varphi^I \leftrightarrow \phi^I$;
- F25.** $(\exists x \varphi(x))^I = \sup\{y \in [0,1]: \text{istnieje wystąpienie konceptu } t \text{ takie, że } y = (\varphi(t))^I\}$;
- F26.** $(\forall x \varphi(x))^I = \inf\{y \in [0,1]: \text{istnieje wystąpienie konceptu } t \text{ takie, że } y = (\varphi(t))^I\}$;

$$\mathbf{F27.} \quad \langle \varphi, \alpha \rangle^I = \begin{cases} 1, & \text{gdy } \varphi^I \geq \alpha \\ \varphi^I, & \varphi^I < \alpha \end{cases}.$$

Gdy interpretacja I spełnia warunki **F1.–F27.**, to nazywamy ją **rozmywaniem języka logiki IRL**. Jeżeli w wyniku rozmywania otrzymujemy tylko funkcje charakterystyczne dla wszystkich konceptów i ról, to taką interpretację nazywamy **dokładną**. Dla terminologii, asercji i aksjomatów warunki **F1.–F19.** są równoważne warunkom standardowej interpretacji logiki opisowej DL.

2.5.3. Proponowana logika wyszukiwania informacji IRL

Można rozszerzyć działanie systemu wyszukiwania informacji IR o szukanie wiarygodnych dla ekspertów podzbiorów $X \cup X \times X$, gdzie X jest pewnym zbiorem dostępnych adresów zasobów internetowych, odnoszących się do pewnej wybranej dziedziny wiedzy. Podzbiory te dla danego zapytania o wiedzę wskazują wiarygodne dla ekspertów adresy sieci Web reprezentujące tę wiedzę. Za pomocą tych zbiorów można wiarygodnie interpretować wyrażenia logiki IRL. W tym celu, podobnie jak w statystyce, używany jest **zakres ufności V** . Uważa się, że najważniejsze jest, by wszyscy eksperci, na podstawie zakresu ufności V , zaakceptowali zbiór stopni przynależności obiektu do zbioru rozmytego. Zbiór ten jest rozmyciem pewnego konceptu lub roli, określonych przez **bazę wiedzy $K = \langle Tez, Fuz, V, Ont \rangle$** , gdzie:

- Tez jest zbiorem rozważanych konceptów i ról tezaurusa,
- Ont – zbiorem rozważanych konceptów i ról ontologii, określonych zgodnie z postulatem **P2**,
- Fuz jest zbiorem możliwych do użycia interpretacji (rozmywania), określonych zgodnie z postulatami **P1.–P9.** oraz warunkami **F1.–F27.**

Wszystkie formuły logiki IRL zbudowane są z wyrażeń zbioru $Tez \cup Ont$ i interpretowane są w algebrze F za pomocą rozmywania ze zbioru Fuz . Formuła φ jest *prawdziwa w bazie wiedzy K* , gdy dla dowolnego rozmywania $I \in Fuz$, $\varphi^I = 1$. Ponieważ interpretacje formuł są zarazem interpretacjami w algebrze BL, algebrze wyznaczonej jednoznacznie przez te interpretacje (przez algebrę F), dlatego piszemy $val_{BL}(\varphi) = 1$. Zbiór wszystkich formuł logiki IRL należy do klasy zbiorów formuł języków logik rozmytych $BL\forall$ interpretowanych w algebrze BL. Te logiki badał Petr Hájek [haj98]. W $BL\forall$ określamy poniższe reguły wnioskowania:

- 1) *modus ponens*: z φ i $\varphi \rightarrow \phi$ wynika ϕ ,
- 2) *reguła zamiany*: można zamienić każdą formułę przez odpowiednie zmienne,
- 3) *generalizacja*: z φ wynika $\forall x \varphi(x)$.

Twierdzenie (pełność i kompletność, ang. soundness and completeness) [cer10] 2.2.

Niech φ będzie formułą z algebry $BL\forall$, a T – z $BL\forall$ -teorii. Wtedy można określić poniższe warunki:

- formuła φ jest wyprowadzana w teorii T z użyciem reguł wnioskowania,

- $val_{BL}(\varphi) = 1$ dla każdej algebry BL (z nieskończonym przecięciem i sumą) jest modelem dla T .

Logika IRL jest dwuwartościowym fragmentem logiki drugiego poziomu. Wartościami wszystkich zmiennych predykatywnych są koncepty i role. Gdy usuniemy formuły ze zmiennymi predykatywnymi i rolami będącymi predykatami, to otrzymamy monadyczny rachunek predykatów. Ten fragment logiki IRL jest rozstrzygalny. Zastosowania pewnych fragmentów logiki IRL, które są rozstrzygalne, można znaleźć w literaturze [cer10], [hus04].

2.5.4. Proponowana interpretacja wyostrzania w logice IRL

Dla bazy wiedzy $K = \langle Tez, Fuz, V, Ont \rangle$, dla pewnego rozmywania $I \in Fuz$, konceptu C lub roli R ich interpretacje należą do zbioru zakresu ufności rozmycia V , zaakceptowanego przez wszystkich ekspertów, określonego następująco:

- $V(C) \subseteq \{\alpha: \text{dla pewnych wystąpień } t \text{ konceptu } C \text{ oraz } I \in Fuz, \alpha = (t: C)^I\}$,
- $V(R) \subseteq \{\alpha: \text{dla pewnych wystąpień } (t_1, t_2) \text{ roli } R \text{ oraz } I \in Fuz, \alpha = ((t_1, t_2): R)^I\}$.

Eksperti rozważają jako wiedzę dokładną, w ramach możliwych interpretacji, wiedzę o stopniach rozmycia należących do zakresu ufności rozmycia. Wyznaczenie takich podzbiorów będzie identyfikowane z **wyostrzaniem reprezentacji wiedzy** o obiektach należących do przestrzeni X lub $X \times X$ [bry12a]. Tak rozszerzony system wyszukiwania informacji IR jest pewnym **agentem wyszukiwania informacji w sieci semantycznej Web**, o ile spełnia postulaty **P1.–P9**. Wprowadzony aparat pojęciowy będzie służył do przygotowania projektu takiego agenta, wspomagającego efektywne wykorzystywanie wyszukiwarek internetowych.

Poniżej przedstawiono, jak interpretować logikę *fuzzyDL* w logice DL.

Funkcję $(\cdot)^{Def}$ nazywamy **wyostrzającą interpretacją dla bazy wiedzy** $K = \langle Tez, Fuz, V, Ont \rangle$, jeśli dla dowolnych konceptów: C i D , ról: R, R_1, R_2 , wystąpień konceptów t, t_1, t_2 z tej bazy zachodzą wzory:

$$\mathbf{D1.} \perp^{Def} = \emptyset, \top^{Def} = X;$$

Jeśli C jest prostym konceptem, to:

$$\mathbf{D2.} C^{Def} = X_C, \text{ gdzie } X_C \subseteq X,$$

$x \in X_C$ wtw, gdy istnieje rozmywanie $I \in Fuz$ takie, że $(t: C)^I \in V(C)$ oraz $x = t^{Def}$;

$$\mathbf{D3.} R^{Def} = (X \times X)_R, \text{ gdzie } (X \times X)_R \subseteq X \times X,$$

$(x, y) \in (X \times X)_R$ wtw, gdy istnieje rozmywanie $I \in Fuz$ takie, że $((t_1, t_2): R)^I \in V(R)$

oraz $x = t_1^{Def}, y = t_2^{Def}$;

Jeśli C jest dowolnym konceptem, to:

$$\mathbf{D4.} (\neg C)^{Def} = X \setminus C^{Def};$$

$$\mathbf{D5.} (C \sqcup D)^{Def} = C^{Def} \cup D^{Def};$$

$$\mathbf{D6.} (C \sqcap D)^{Def} = C^{Def} \cap D^{Def};$$

$$\mathbf{D7.} (\exists R. C)^{Def} = \{x \in X: \text{istnieje } y \text{ taki, że } (x, y) \in R^{Def} \text{ i } y \in C^{Def}\};$$

$$\mathbf{D8.} (\forall R. C)^{Def} = \{x \in X: \text{dla każdego } y, \text{ jeśli } (x, y) \in R^{Def}, \text{ to } y \in C^{Def}\};$$

D9. $(t: C)^{Def}$ wtw $t^{Def} \in C^{Def}$;

D10. $((t_1, t_2): R)^{Def}$ wtw $(t_1^{Def}, t_2^{Def}) \in R^{Def}$;

D11. $(C \sqsubseteq D)^{Def}$ wtw $C^{Def} \subseteq D^{Def}$ i $(C = D)^{Def}$ wtw $C^{Def} = D^{Def}$;

D12. $(R_1 \sqsubseteq R_2)^{Def}$ wtw $R_1^{Def} \subseteq R_2^{Def}$ i $(R_1 = R_2)^{Def}$ wtw $R_1^{Def} = R_2^{Def}$;

D13. $\langle E, \alpha \rangle^{Def}$ wtw E^{Def} , jeśli $K \mid - \langle E, \alpha \rangle$, albo $\langle E, \alpha \rangle^{Def}$ wtw $\neg E^{Def}$, gdy nie zachodzi $K \mid - \langle E, \alpha \rangle$.

E jest asercją lub aksjomatem bazy wiedzy K .

Aksjomat, którego wyostrzenie zachodzi dla dowolnej bazy wiedzy K , nazywamy **adekwatnym** [bry13b].

Z powyższej definicji semantyki *fuzzyDL* i semantyki logiki DL wynika, że każdy zachodzący aksjomat logiki DL jest aksjomatem, który w logice *fuzzyDL* jest aksjomatem adekwatnym.

Warunek **D13.** umożliwia, przy zastosowaniu wyostrowania $(\cdot)^{Def}$ dla aksjomatu E , szukanie największego stopnia α , takiego, że zachodzi E^{Def} , jeśli zachodzi $K \mid - \langle E, \alpha \rangle$ [bob08]. Taka analiza jest także przydatna przy weryfikacji przez ekspertów różnych wariantów określenia zbioru Fuz i funkcji zakresu ufności rozmycia V [bry12a].

2.6. Metoda inteligentnego wyszukiwania wiedzy diagnostycznej z wykorzystaniem logiki IRL

Niniejszy podrozdział stanowi krótkie podsumowanie badań teoretycznych zawartych w pracy, które prowadzą do utworzenia metody inteligentnego wyszukiwania wiedzy diagnostycznej z wykorzystaniem rozmytej logiki opisowej.

Urządzenia techniczne, procesy technologiczne i produkcyjne cechuje coraz większa złożoność. Powoduje to, że przy rozwiązywaniu pewnych zadań technicznych i inżynierskich dostępna wiedza jest niepewna, niejasna lub niepełna. Dotyczy to również diagnostyki technicznej oraz wiedzy o symptomach stanów, w których pozostają badane obiekty. Stan diagnozowanego obiektu określa wystąpienie pewnej usterki, wady czy awarii. Zdarza się często, że dysponując niepełną wiedzą o symptomach stanów obiektów, próbujemy ustalić ich stan wcześniejszy, obecny lub przyszły [kor02].

Wiedzę przetwarzaną w procesie diagnostycznym można podzielić na poszczególne pojęcia oraz związki, jakie zachodzą pomiędzy tymi pojęciami. Wtedy do reprezentacji tej wiedzy można wykorzystać **sieci semantyczne** [ber01], [bak09], [sow91], które umożliwiają taki opis rzeczywistości. Ponadto sieci semantyczne umożliwiają na komputerowe przetwarzanie wiedzy. Pozwalają one na pozyskiwanie wiedzy z globalnych sieci komputerowych, dzięki zastosowaniu projektu sieci semantycznej Web, określonego przez W3C [w3c14].

W sieci semantycznej można również zapisywać, przetwarzać i wyszukiwać wiedzę niepewną, niejasną lub niepełną. Wtedy do opisu wiedzy w tej sieci stosuje się **rozmytą logikę opisową** (ang. *Fuzzy Description Logic – fuzzyDL*) [baa03], [bob08], [str98]. Logika opisowa umożliwia na opis wiedzy za pomocą conceptów, reprezentujących pojęcia, oraz ról,

reprezentujących związki zachodzące pomiędzy poszczególnymi pojęciami w tej sieci. Natomiast rozmyta logika opisowa dodatkowo umożliwia przypisanie stopni rozmycia, takich samych jak w logice rozmytej [zad65], do wszystkich elementów tej sieci.

Motywacją do rozpoczęcia rozważań nad problemem wyszukiwania wiedzy z zakresu diagnostyki technicznej w sieciach semantycznych, jest problem określony w pytaniu: *Jak precyzyjnie określić proces wyszukiwania wiedzy (informacji) diagnostycznej reprezentowanej w sposób rozmyty w sieciach semantycznych, w wyniku którego otrzyma się wiedzę dokładną?* Rozwiązanie problemu diagnozowania obiektów technicznych w warunkach niepewności pomaga w opracowaniu ogólnych algorytmów automatyzacji procesów wytwarzania tych obiektów oraz algorytmów sterowania nimi w środowisku rozmytym [gal04].

Przesłanki terminologiczne i teoretyczne, potrzebne do sformułowania metody inteligentnego wyszukiwania wiedzy diagnostycznej w sieciach semantycznych, były prezentowane przez autorkę w pracach [bry11a], [bry11b], [bry12a], [bry12b], [bry12c], [bry13a], [bry13b].

Wyszukiwanie informacji diagnostycznych w sieciach semantycznych Web sprowadza się do określenia z pomocą *ekspertów*, czyli agentów będących źródłem wiedzy diagnostycznej, egzemplarzy danych będących:

- wartościami jednoargumentowych atrybutów, czyli *konceptami*,
- wartościami dwuargumentowych atrybutów, czyli *rolami*.

Koncepty reprezentują wiedzę o pewnych cechach lub rodzajach obiektów diagnostycznych, natomiast role reprezentują wiedzę o własnościach obiektów lub relacjach pomiędzy dwoma obiektami diagnostycznymi. Wyróżnienie konceptów i ról odbywa się na podstawie ustalenia klasyfikatora konstrukcyjno-technologicznego będącego hierarchią klas danych konstrukcyjnych i technologicznych, ustalających wiedzę techniczną o konstrukcji i budowie diagnozowanego obiektu. Nazwy tych klas są nazwami konceptów, a nazwy relacji zawierania się tych klas są nazwami ról.

Wzorcowe koncepty i role opisane są w **tezaurusie**, utworzonym w języku rozmytej logiki opisowej, natomiast obserwowane koncepty i role opisane są w **ontologii**. Ontologia tworzona jest ze słów i formuł z tezaury oraz z zasobów internetowych. Oznacza to, że wyrażenia ontologii mogą odnosić się do innej wiedzy niż egzemplarze danych tezaury. Innymi słowy, wyrażenia ontologii mogą wynikać logicznie z wiedzy reprezentowanej przez egzemplarze danych z zasobów internetowych, niewystępujących w tezaurusie. Egzemplarze danych pozyskane w tych zasobach mogą reprezentować szukaną wiedzę diagnostyczną. Jeśli dla szukanych danych, z ontologii oraz z zaleceń (kryteriów i wiedzy) ekspertów, będzie wynikać stopień zgodności tych danych z pewnymi danymi z tezaury, to ten związek nazywany jest **residuum** wyszukiwania informacji diagnostycznej.

W pracy przyjęto następującą regułę wyszukiwania, zwaną **regułą residuum**. Pytanie o informację jest wyrażeniem ontologii zbudowanym ze słów i formuł tezaury. Jeżeli jedynym zasobem, z którego pozyskiwane są dane, jest strona internetowa tezaury, to znaczy, że zbiór szukanych adresów sieci semantycznej Web jest pusty. Jeżeli ma to miejsce dla wszystkich

przyjętych interpretacji (metod) ustalających stopień podobieństwa pozyskanych egzemplarzy danych do egzemplarzy danych tezaury, to w takim przypadku stopień podobieństwa pozyskanych egzemplarzy danych w innych zasobach wiedzy jest zerowy. W innym przypadku, to akceptowany przez ekspertów zbiór adresów zasobów wiedzy diagnostycznej reprezentuje wiedzę najbliższą znaczeniowo szukanej informacji.

Wyszukiwanie informacji diagnostycznych i wnioskowanie diagnostyczne prowadzi się zgodnie z **logiką wyszukiwania informacji IRL** (ang. *Information Retrieval Logic*), wykorzystującą powyższe zasady, regułę oraz pewne postulaty dotyczące wyszukiwania informacji w sieciach semantycznych [bry13b].

Wyszukiwanie informacji w sieci semantycznej Web rozumiane jest jako szukanie wiarygodnych dla ekspertów adresów zasobów internetowych, w których dostępne są szukane dane diagnostyczne. Szukane dane są interpretowane zgodnie z logiką IRL. W tym celu, podobnie jak w statystyce, określa się **zakres ufności** dla pozyskiwania wiedzy diagnostycznej. Jeśli stopień przynależności obiektu diagnostycznego (zasobu) zawiera się w zakresie ufności, oznacza to, że wszyscy eksperci zaakceptowali dany obiekt.

Te stopnie przynależności, zgodnie z postulatami logiki IRL, są stopniami podobieństwa wyszukiwanych danych do danych tezaury.

Podsumowując, należy stwierdzić, że aby wyszukiwać informacje diagnostyczne z wybranej dziedziny wiedzy, należy dla tej dziedziny określić **bazę wiedzy** $K = \langle Tez, Fuz, V, Ont \rangle$, gdzie *Tez* jest zbiorem rozważanych konceptów i ról tezaury, a *Ont* – zbiorem rozważanych konceptów i ról ontologii. *Fuz* jest zbiorem możliwych do użycia interpretacji rozmywania. Wszystkie formuły logiki IRL zbudowane są z wyrażeń z tezaury i ontologii i interpretowane są w wybranej przez ekspertów algebrze zbiorów rozmytych, za pomocą operacji rozmywania ze zbioru *Fuz*. Formuła φ opisująca wyszukiwaną wiedzę jest *prawdziwa* w *bazie wiedzy* *K*, gdy dla dowolnego rozmywania $I \in Fuz$, $\varphi^I = 1$.

W kolejnym rozdziale zostanie opisana między innymi koncepcja systemu na podstawie opisanej metody. Zostanie również przedstawiona implementacja tej koncepcji.

3. KONCEPCJA INTELIGENTNEGO SYSTEMU WYSZUKIWANIA WIEDZY DIAGNOSTYCZNEJ

Niniejszy rozdział przedstawia praktyczne aspekty pracy dotyczące zastosowania opracowanej metody (rozd. 2.6.) dla tworzenia systemów wyszukiwania wiedzy diagnostycznej w sieciach semantycznych. W pierwszym podrozdziale zostanie opisana koncepcja takiego systemu. Następnie zostanie scharakteryzowany język programistyczny OWL służący do zapisu sieci semantycznych na potrzeby przetwarzania komputerowego.

W podrozdziale 3.3. zostaną przedstawione dwa przykłady: pierwszy – ilustrujący zastosowanie i tworzenie klasyfikatora konstrukcyjno-technologicznego dla obiektów diagnozowanych, a drugi – przedstawiający przykładową implementację procesu diagnostyki ciśnienia sprężania w silniku samochodowym w języku OWL.

Podrozdział 3.4. przedstawia implementację opisanej koncepcji na wybranej klasie obiektów diagnozowanych, którą są klasyczne kotły gazowe małej mocy. Dla obiektu zostały utworzone tezaursus w postaci sieci semantycznej w języku OWL oraz aplikacja pozwalająca na wyszukiwanie stanu kotła na podstawie obserwowanych symptomów. Aplikacja pozwala również na ustalenie, które części kotła mogły ulec awarii.

3.1. Koncepcja implementacji

Podrozdział najpierw wskazuje na wyniki badań teoretycznych, uzyskane przez autorkę niniejszej rozprawy doktorskiej, a wykorzystane w podrozdziale 3.4., pozwalające w diagnozie technicznej wykorzystywać informacje wyszukiwane w sieci internetowej. Wyróżnione zostają poszczególne kroki diagnozy prowadzące do reprezentacji wiedzy diagnostycznej w sieciach semantycznych i wykorzystania tej reprezentacji do wyszukiwania informacji diagnostycznej (pozyskiwania danych diagnostycznych) w sieci internetowej. W drugim podrozdziale podanych zostaje kilka istotnych uwag o znaczeniu wiedzy ekspertów w wyszukiwaniu informacji diagnostycznej. Końcowy podrozdział nakreśla ogólne algorytmy, które zostały już zaimplementowane przez autorkę lub zostaną zaimplementowane w przyszłości.

3.1.1. Wyniki badań teoretycznych umożliwiające opracowanie koncepcji

W podrozdziale 2.2. precyzyjnie określono system komunikowania się agentów, w którym agenci kodują i interpretują dane z różnych dziedzin rzeczywistości, w tym dane diagnostyczne. Jeśli agenci, komunikując się, pozyskują dane bez wykorzystywania technologii informacyjnej, mówi się, za Wojciechem Moczulskim [moc02, s. 108], o *Modelu 1* pozyskiwania wiedzy, a gdy wykorzystują technologię informacyjną – o *Modelu 2*. Sieć internetowa jest systemem komunikowania się, w której działa *Model 2* pozyskiwania wiedzy, zwany tu systemem wyszukiwania informacji. Dlatego dalej formułuje się koncepcję implementacji wyszukiwania informacji diagnostycznej. W tym kontekście przyjęte w rozdziale 2. pracy rozumienie struktury systemu pomiarowego pozwala ten system oraz sieć internetową rozważać jako system komunikowania się, w którym agenci, dokonując pomiarów wielkości opisujących

obiekty diagnostyczne, tym samym dokonują diagnozy. Jako obiekty diagnostyczne rozumiane są urządzenia techniczne, procesy, cechy i własności, stany i symptomy. Obiekty te utożsamiane są z zasobami wiedzy diagnostycznej. Zasobami wiedzy diagnostycznej są diagnozowane obiekty, wiedza ekspertów, inne źródła danych służące diagnozie, w tym zasoby internetowe. Z tych zasobów pozyskiwane są dane diagnostyczne. Dane te reprezentują wiedzę diagnostyczną w systemach informacyjnych, w tym również w sieci internetowej.

Ustalono model wiedzy w systemach danych jako zbiór elementów wiedzy (rozd. 2.3.). Elementy wiedzy są klasami jednakowo używanych przez agentów, w systemach danych, typów danych, a typy danych są klasami składającymi się z egzemplarzy danych. W tym sensie uzasadniono, że system diagnostyczny jest specyficznym systemem danych, a w kontekście prowadzonych rozważań teoretycznych można go traktować jako pewien system pomiarowy odpowiadający diagnozie. Przyjęto, że procedura przetwarzania danych diagnostycznych powinna być zgodna z modelem procesu pomiarowego przedstawionym wyżej na rysunku 2.4. Stosując ten model do wyszukiwania informacji diagnostycznej w sieciach semantycznych reprezentujących tę wiedzę, przyjmie się tu następujące interpretacje elementów tego modelu:

X – zbiór obiektów diagnostycznych, czyli zasobów wiedzy diagnostycznej, dla których zbiór egzemplarzy danych pomiarowych wyznaczony jest przez procedurę pozyskiwania egzemplarzy danych diagnostycznych w wyróżnionej sieci semantycznej (wyróżnionej przez tezaurus). Koncepty i role w tej sieci można rozpoznać w tych zasobach. Zasobami są dokumenty tekstowe zawierające opisy symptomów i stanów diagnostycznych, tak więc egzemplarzami danych są napisy będące wystąpieniami tych opisów, czyli *rozważa się tylko wartości jakościowe pomiaru* (wartości mierzonych atrybutów). Zakłada się, że te zasoby są dostępne w sieci internetowej, natomiast ich zbiór, określony przez adresy internetowe tych zasobów, jest przestrzenią wyszukiwania informacji diagnostycznej zgodnie z postulatami logiki wyszukiwania informacji (rozd. 2.4.3.).

E – zbiór **etalonów**, modeli obiektów diagnozowanych rozważanych jako klasyfikatory konstrukcyjno-technologiczne tych obiektów (opisane szerzej dalej w rozdz. 3.1.3. i 3.3.1.). Są one określone przez mierzone w procesie diagnozy atrybuty i przez wzorcowe wartości tych atrybutów. Klasyfikatory są sieciami semantycznymi wyznaczonymi przez mierzone atrybuty: symptomy i stan obiektu diagnozowanego (rozd. 2.2.10.). Opisy tych atrybutów są konceptami i rolami tej sieci semantycznej (por. rozdz. 2.2.6.), a wystąpienia tych konceptów i ról są wzorcowymi napisami – słownikiem i zwrotami – służącymi do budowy języka tezaury M (rozd. 2.2.8.).

Na mechanizmy pomiaru składają się następujące procedury:

M – tezaurus, zapisany za pomocą sieci semantycznej wyznaczonej przez zbiór atrybutów etalonów E , których wartości są mierzone dla zbioru zasobów X . Tezaurus jest implementowany jako *agent wyszukiwania informacji*. Umożliwia on pozyskiwanie danych diagnostycznych opisanych przez tezaurus na wyróżnionej stronie sieci internetowej (stronie internetowej tezaury). Co się z tym wiąże, dowolny rozważany tu tezaurus jest wyznaczony przez atrybuty, których wartości są danymi o egzemplarzach dostępnych na stronie internetowej

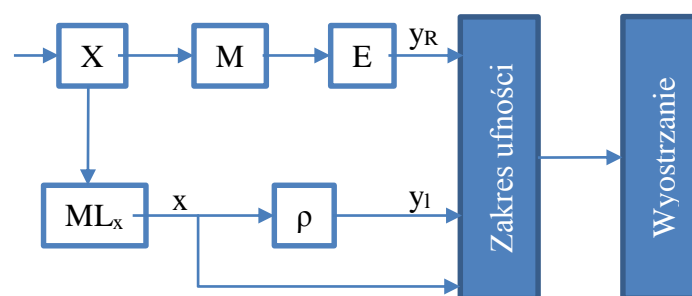
tezaurusu. Zgodnie z postulatami logiki wyszukiwania informacji (rozdz. 2.4.3.) tezaurus może reprezentować wiedzę diagnostyczną (dane diagnostyczne) dokładnie lub reprezentować wiedzę występującą w pewnym stopniu rozmycia (dane rozmyte). *Wyszukiwanie informacji w jakimś zasobie jest dokładne*, gdy pozyskana dana jest równoznaczna z daną pozyskaną na stronie internetowej tezaurusu, jest wtedy daną pozyskaną w stopniu podobieństwa równym 1 do danych tezaurusu. Jeśli stopień podobieństwa napisów reprezentujących pozyskiwaną daną do napisów dostępnych na stronie tezaurusu jest mniejszy od 1, to *wyszukiwanie informacji jest rozmyte*. Zatem wyszukiwanie informacji na stronie internetowej tezaurusu jest dokładne.

Modelem wyszukiwania informacji jest interpretacja wystąpień wyrażen języka logiki opisowej i formuł logiki wyszukiwania informacji, tj. interpretacja opisów danych diagnostycznych w strukturze zbiorów rozmytych $F = \langle F, \wedge^F, \vee^F, \neg^F, c^F, e^F, 0^F, 1^F, M, F_0 \rangle$ określonych na zbiorze $X \cup X \times X$ za pomocą operacji z algebry $BL = \langle L, \otimes, \oplus, \bullet, \rightarrow, 0, 1 \rangle$ (patrz rozdz. 2.5.).

ML_x – procedura tworzenia ontologii poprzez losowe przypisywanie poszczególnym atrybutom tezaurusu pewnych egzemplarzy danych x pozyskiwanych w innych zasobach, poza stroną tezaurusu. Rozszerzenie zbioru atrybutów wyznaczających tezaurus o atrybuty, których wartościami są dane pozyskiwane w sieci internetowej poza tezaurusem, prowadzi do ustalenia nowego zbioru atrybutów, który wyznacza sieć semantyczną zwaną *ontologią*. Tezaurus jest więc częścią ontologii (por. postulaty **P1.**, **P2.** oraz rozdz. 2.2.8.).

A_I – procedura ustalająca wyszukiwanie w zasobach X losowo wybranych wyrażen języka tezaurusu zgodnie z kryterium ρ , określonym przez postulaty **P4.–P6.** logiki wyszukiwania informacji (rozdz. 2.4.3.).

Pozostałe procedury pomiaru: $ML_{\bar{x}}, ML_z, ML_c, ML_k, A_R, \rho', A_S, U$, ze względu na rozważanie tylko jakościowych wartości atrybutów (rysunek 3.1.), tj. językowych opisów symptomów i stanów obiektu diagnozowania, zostają sprowadzone do wyznaczenia zakresu ufności dla wyszukiwania informacji.



Rysunek 3.1. Model procesu pomiarowego dla wartości jakościowych. Opracowanie własne.

Zakres ufności jest operacją ustalającą akceptację (ufność do) zasobów, z których można pozyskać dane diagnostyczne najbliższe znaczeniowo danym pozyskiwanym ze strony tezaurusu, tym samym można dokonać wyostrzenia wyszukiwania informacji (rozdz. 2.5.3., 2.5.4.).

3.1.2. Procesy decyzyjne ekspertów prowadzące do ustalenia zakresu ufności

Wyszukiwanie informacji diagnostycznych w sieciach semantycznych Web sprowadza się do określenia z pomocą *ekspertów* (agentów będących źródłem wiedzy diagnostycznej) egzemplarzy danych będących wartościami jedno- i dwuargumentowych atrybutów, czyli konceptów i ról. Następnie koncepty i role zapisuje się za pomocą klasyfikatora konstrukcyjno-technologicznego, w którym wiedza techniczna jest zapisana za pomocą hierarchii klas.

Do opisu konceptów i ról wykorzystywany jest język **logiki opisowej DL**. Język logiki DL opisujący koncepty i role może zostać rozszerzony o pewne formuły logiki pierwszego rzędu, także logiki rozmytej (formuły prawdziwe w pewnym stopniu). W tak rozszerzonym języku tworzy się **tezaurus** opisujący wzorcowe koncepty i role oraz **ontologię** opisującą obserwowane koncepty i role, czyli wyszukiwane dane. Język ontologii jest tworzony ze słów i formuł tezaury oraz innych słów i formuł występujących w zasobach internetowych. Oznacza to, że wyrażenia ontologii mogą odnosić się do innej wiedzy niż egzemplarze danych tezaury, czyli wiedzy mogącej wynikać logicznie z wiedzy reprezentowanej przez inne egzemplarze danych. Egzemplarze danych opisanych przez te wyrażenia mogą występować w innych zasobach internetowych niż strona tezaury. Egzemplarze danych pozyskane w tych zasobach mogą reprezentować szukaną wiedzę diagnostyczną. Jeśli dla szukanych danych z ontologii oraz z zaleceń (kryteriów i wiedzy) ekspertów będzie wynikać stopień zgodności tych danych z pewnymi danymi z tezaury, to ten związek nazywany jest **residuum** wyszukiwania informacji diagnostycznej.

Przyjęto, że istotą wyszukiwania informacji jest stosowanie reguły wyszukiwania, zwanej **regułą residuum** (rozd. 2.4.1.). Wyszukiwanie informacji diagnostycznych i wnioskowanie diagnostyczne oparte na tej regule oraz postulatach **P1.–P9.** (rozd. 2.4.3.) prowadzi się zgodnie z **logiką wyszukiwania informacji IRL**.

Przez pozyskiwanie danych diagnostycznych w sieci internetowej, czyli wyszukiwanie informacji, będzie się rozumiało szukanie wiarygodnych dla ekspertów adresów zasobów internetowych, tj. takich podzbiorów, które odnoszą się do pewnej wybranej dziedziny wiedzy diagnostycznej. Podzbiory te, dla danego zapytania dotyczącego tej wiedzy, wskazują wiarygodne dla ekspertów adresy zasobów internetowych, w których dostępne są szukane dane diagnostyczne. Można więc za pomocą tych zbiorów wiarygodnie interpretować wyrażenia logiki IRL. W tym celu, podobnie jak w statystyce, powinien być określony **zakres ufności V** dla pozyskiwania wiedzy diagnostycznej. Uważa się, że najważniejsze jest, by wszyscy eksperci, na podstawie zakresu ufności V , zaakceptowali jakiś zbiór stopni przynależności obiektu diagnostycznego (zasobu) do dowolnego zbioru rozmytego:

$$A = \{(x, \mu_A(x)); x \in X \cup X \times X; \mu_A: X \cup X \times X \rightarrow [0,1]\} \quad (70)$$

Stopnie przynależności, zgodnie z postulatami logiki IRL, są stopniami podobieństwa wyszukiwanych danych do danych tezaury.

W celu wyszukania informacji diagnostycznej z wybranej dziedziny wiedzy, należy dla tej dziedziny określić **bazę wiedzy** $K = \langle Tez, Fuz, V, Ont \rangle$, gdzie *Tez* to tezaurus, *Ont* to ontologia, określona zgodnie z postulatem **P2**, a *Fuz* to interpretacje rozmywania, określone zgodnie z postulatami **P1.–P9.** oraz warunkami **F1.–F27.** (rozdz. 2.5.1.). Dalej *Fuz* będzie utożsamiany z użyciem jednej metody ustalania podobieństwa pomiędzy wyszukaną wiedzą a wiedzą reprezentowaną przez tezaurus w różnych momentach czasowych. Wszystkie formuły logiki IRL zbudowane są z wyrażeń zbioru $Tez \cup Ont$ i interpretowane są w algebrze **F** za pomocą operacji rozmywania ze zbioru *Fuz*.

3.1.3. Opis koncepcji systemu wyszukiwania wiedzy w sieciach semantycznych

Poniżej przedstawiono koncepcję systemu wyszukiwania wiedzy w sieciach semantycznych, która jest celem niniejszej rozprawy doktorskiej. Można wyróżnić trzy etapy realizacji koncepcji systemu jako realizację trzech tematów programistycznych:

- a) Agent wyszukujący automatycznie informację diagnostyczną w tezaurusie w zadanym zakresie wiedzy.
- b) Interfejs pozwalający określić ontologię diagnozy.
- c) Agent wyszukujący automatycznie informacje diagnostyczne w zasobach internetowych.

Na tej podstawie poniżej określono procedury wyznaczania tezaurusu i ontologii, które są później wykorzystywane w procesie wyszukiwania wiedzy diagnostycznej w sieci semantycznej ze źródeł dostępnych w sieci internetowej. Problematyka badawcza prezentowana poniżej została zainspirowana monografią [kno02] oraz wynikami badań autorki niniejszej pracy [bry12b]. Prezentowany jest ogólny opis algorytmu wyszukiwania wiedzy diagnostycznej w sieciach WWW. Algorytm jest pomysłem autorki niniejszej pracy.

Procedura wyznaczania tezaurusu

Do wyszukiwania informacji należy najpierw przygotować klasyfikator konstrukcyjno-technologiczny dla obiektu diagnostycznego. W klasyfikatorze zostaną uwzględnione części tego obiektu w zależności od technologii, jakiej dotyczą. Dla obiektów o wzorcach O_1, O_2, \dots, O_k tworzony jest model $Model_CT(O_1, O_2, \dots, O_k)$. Poniżej określono algorytm tworzenia klasyfikatora konstrukcyjno-technologicznego, zapisanego w tezaurusie w sieci semantycznej.

Blok 1. Wyznaczanie tezaurusu

Należy zdefiniować w języku OWL:

- 1) Klasy przedmiotów o wzorcach konstrukcyjnych obiektów O_1, O_2, \dots, O_n .
- 2) Podklasy $T_1^i, T_2^i, \dots, T_t^i$ dla *i*-tej klasy, odpowiadające wyróżnionym technologiom.
- 3) Zespoły konstrukcyjne $K_1^{i,j}, K_2^{i,j}, \dots, K_t^{i,j}$ dla *j*-tej technologii.
- 4) Części bazowe $B_1^{i,j,k}, B_2^{i,j,k}, \dots, B_m^{i,j,k}$ dla *k*-tego zespołu konstrukcyjno-technologicznego, spełniające następujące warunki: posiadają jednoznacznie ustalony

symbol albo są określone przez zakres wymiarowy lub przez przedział wymiarowy. Niespełnienie tych warunków przez obiekt diagnozowany jest jego brakiem, wadą lub uszkodzeniem.

- 5) Relacje zawierania pomiędzy klasami: $T_s^i \subseteq O_i, K_s^{i,j} \subseteq T_j^i, B_s^{i,j,k} \subseteq K_k^{i,j}$.
- 6) Modyfikatory i synonimy wybranych nazw klas i nazw relacji. W tym celu można wykorzystać na przykład projekt plWordNet [plw14], w którym zapisana jest semantyka słów.
- 7) Schematy wyrażeń języka logiki deskrypcyjnej.
- 8) Aksjomaty logiki deskrypcyjnej.

Hierarchia klas zdefiniowana w punktach 1)–5) jest klasyfikatorem konstrukcyjno-technologicznym obiektów o wzorcach O_1, O_2, \dots, O_k , co oznaczamy $Model_CT(O_1, O_2, \dots, O_k)$. Natomiast ontologia zdefiniowana w punktach 1)–8) jest tezauresem dla obiektów O_1, O_2, \dots, O_k , co oznaczamy $Tezaurus(O_1, O_2, \dots, O_k)$.

Tabele morfologiczną dla $Model_CT(O_1, O_2, \dots, O_k)$ można określić jako zbiór wszystkich układów klas postaci $[O_i, T_j^i, K_k^{i,j}, B_s^{i,j,k}]$. Ponadto można ustalić sieć semantyczną dla $Model_CT(O_1, O_2, \dots, O_k)$, definiując wierzchołki i krawędzie tej sieci następująco:

- Zbiór wierzchołków $\Omega = \{\omega: \omega = i \wedge O_i \in Class\} \cup \{\omega: \omega = (i, j) \wedge T_j^i \in Class\} \cup \{\omega: \omega = (i, j, k) \wedge K_k^{i,j} \in Class\} \cup \{\omega: \omega = (i, j, k, s) \wedge B_s^{i,j,k} \in Class\}$.
- Nazwy klas $O_i, T_j^i, K_k^{i,j}, B_s^{i,j,k}$ są opisami wierzchołków: $i, (i, j), (i, j, k), (i, j, k, s)$.
- Zbiór krawędzi to zbiór wszystkich par wierzchołków, których opisy spełniają jedną z relacji postaci: $T_s^i \subseteq O_i, K_s^{i,j} \subseteq T_j^i, B_s^{i,j,k} \subseteq K_k^{i,j}$. Natomiast nazwy tych relacji są opisami krawędzi.

Procedura wyznaczania ontologii

Ontologia określa nam klasyfikator konstrukcyjno-technologiczny podobieństwa obiektu diagnozowanego O do obiektu wzorcowego $O_0 \in \{O_1, O_2, \dots, O_k\}$ w zasobie internetowym X , co oznaczamy $Diag_CT(x, O_0, O)$. Blok 2. przedstawia algorytm wyznaczania ontologii. Diagnozę obiektu O w sieci semantycznej Web definiuje się w oparciu o $Tezaurus(O_1, O_2, \dots, O_k)$.

Blok 2. Wyznaczanie ontologii

Trzeba zdefiniować w języku OWL przestrzeń zasobów internetowych X oraz należy:

- 1) Wyróżnić klasę obiektu diagnostycznego O .
- 2) Wyróżnić obiekt wzorcowy $O_0 \in \{O_1, O_2, \dots, O_k\}$, do którego porównywany będzie obiekt O .
- 3) Zawęzić klasyfikator $Model_CT(O_1, O_2, \dots, O_k)$ tylko do klas opisujących obiekt wzorcowy O_0 . Dla tego klasyfikatora można przyjąć oznaczenie $Model(O_0)$.

- 4) Określić klasy tezaury opisyujące obiekt O_0 , pozostające w relacji tezaury z klasą obiektu diagnostycznego O .
- 5) Wyznaczyć relacje tezaury ograniczonego do klasyfikatora $Model(O_0)$.
- 6) Zdefiniować i wyróżnić nowe klasy konstrukcyjno-technologiczne, z którymi w pewnych relacjach tezaury pozostaje klasa obiektu O .
- 7) Sporządzić listę zasobów internetowych $L(X)$, wraz z zasobem zdefiniowanym w wyszukiwarce, w których wykryte zostały wyróżnione klasy konstrukcyjno-technologiczne zwane symptomami diagnozy, pozostające w relacjach tezaury z klasami tezaury ograniczonego do klasyfikatora $Model(O_0)$. Pozostałe klasy tezaury ograniczonego do klasyfikatora $Model(O_0)$ zwane są stanami diagnozy. Stany diagnozy wskazują na pewne braki wiedzy, istotne odchylenia, usterki lub uszkodzenia obiektu diagnozowanego.
- 8) Dla każdego zasobu $x \in L(X)$ zdefiniować zbiór symptomów diagnozy $Symptom(x, O_0, O)$ oraz zbiór stanów diagnozy $State(x, O_0, O)$.
- 9) Wyznaczyć klasyfikator konstrukcyjno-technologiczny podobieństwa obiektu diagnozowanego O do obiektu wzorcowego $O_0 \in \{O_1, O_2, \dots, O_k\}$ w zasobie internetowym $x \in L(X)$: $Diag_CT(x, O_0, O)$ jako zbiór symptomów i stanów diagnozy z relacjami tezaury pomiędzy nimi.
- 10) Zdefiniować ontologię dla $Diag_CT(x, O_0, O)$, zwana ontologią diagnozy obiektu O , na podstawie podobieństwa O do obiektu wzorcowego O_0 w zasobie $x \in L(X)$: $DiagOnt(x, O_0, O)$.
- 11) Dla każdego zasobu internetowego $x \in L(X)$ określić metodą podaną dalej wartość wskaźnika podobieństwa $\mu(x, O_0, O) = [\omega_0]$ obiektu diagnozowanego O do obiektu wzorcowego O_0 w zasobie $x \in L(X)$, gdzie ω_0 jest wierzchołkiem sieci semantycznej klasyfikatora $Model(O_0)$, wierzchołka o nazwie O_0 .

Dyskretyzacja wartości podobieństwa konceptów i ról do danych wzorcowych

Numeryczną, zdyskredytowaną wartość podobieństwa konceptów i ról określoną w węzłach sieci ustalono następująco. Przyjęto, że dla obiektu wzorcowego $O_0 \in \{O_1, O_2, \dots, O_k\}$ oraz obiektu diagnozowanego O określone są, dla zasobu $x \in L(X)$, stany diagnozy $State(x, O_0, O)$. W stanach diagnozy obiekty O_0 i O nie są podobne dla zasobu x . Dla pozostałych klas konstrukcyjno-technologicznych można dokonać dyskretyzacji, osobno dla obiektu wzorcowego i osobno dla obiektu diagnozowanego. Dyskretyzacja ustali dla poszczególnych klas wartości atrybutów dla elementów tych klas. Są to wartości nominalne, zakresowe lub przedziałowe.

Jeśli po dyskretyzacji numeryczna wartość porównywanej klasy ma zakres wyrażony liczbą d i jest wyrażona jedną liczbą (wynosi a dla obiektu wzorcowego, b dla obiektu porównywanego z obiektem wzorcowym), to współczynnik podobieństwa tych obiektów α jest określony wzorem:

$$\alpha = \begin{cases} \frac{1-|b-a|}{d}, & \text{dla } |b-a| < d \\ 0, & \text{dla } |b-a| \geq d \end{cases} \quad (71)$$

Gdy numeryczne wartości porównywanej cechy są określone w sposób przybliżony za pomocą przedziałów $[a_1, a_2]$ dla obiektu wzorcowego oraz $[b_1, b_2]$ dla obiektu porównywanego z obiektem wzorcowym, wtedy współczynnik podobieństwa tych obiektów α określony jest wzorem:

$$\alpha = \begin{cases} \frac{1-|a_2-b_1|}{|a_1-b_2|}, & \text{jeśli } [a_1, a_2] \cap [b_1, b_2] = \emptyset \text{ i } a_1 < b_1 \\ \frac{1-|a_1-b_2|}{|a_2-b_1|}, & \text{jeśli } [a_1, a_2] \cap [b_1, b_2] = \emptyset \text{ i } a_1 > b_1 \\ 0, & \text{jeśli } [a_1, a_2] \cap [b_1, b_2] \neq \emptyset \end{cases} \quad (72)$$

Niech $\sigma(\omega)$ jest miarą zgodności strukturalnej wierzchołka (węzła) ω . Dla dyskretyzacji klasy, której nazwa opisuje ten wierzchołek, gdy dla obiektu wzorcowego i diagnozowanego wartości dyskretyzacji są różnego typu lub są nominalne, to $\sigma(\omega) = 0$. Natomiast $\sigma(\omega) = 1$, gdy wartości dyskretyzacji są takiego samego typu i nie są nominalne.

Przyjmuje się, że $w(\omega)$ jest wagą wierzchołka (węzła) ω , będącą pewną liczbą naturalną. Wskaźnik podobieństwa $[\omega]$ obiektu diagnozowanego do wzorcowego w wierzchołku ω sieci semantycznej klasyfikatora $Model_CT(O_0)$ określamy rekurencyjnie:

- $[\omega] = 0$, gdy ω opisane są przez nazwy stanów diagnozy,
- $[\omega] = \alpha$, w przeciwnym wypadku, gdzie α jest wyliczona wzorem (71) – dla cech wyrażonych liczbowo, a wzorem (72) – dla cech wyrażonych przedziałowo.

Gdy cechy określone w węźle ω wyrażone są w węzłach $\omega_1, \omega_2, \dots, \omega_{n(\omega)}$, a $n(\omega)$ jest liczbą cech określonych w węźle ω , to stosujemy wzór rekurencyjny:

$$[\omega] = \sigma(\omega) \frac{\sum_{i=1}^{n(\omega)} w(\omega_i) * [\omega_i]}{\sum_{i=1}^{n(\omega)} w(\omega_i)} \quad (73)$$

Funkcję ważenia dla węzłów $\omega_i, i = 1, \dots, n(\omega)$ określamy wzorem:

$$v(\omega_i) = \frac{w(\omega_i)}{\sum_{i=1}^{n(\omega)} w(\omega_i)} \quad (74)$$

Wartość tej funkcji nazywamy stopniem ważenia węzłów. Zauważmy, że dla dowolnego węzła ω' połączonego z wyżej położonym węzłem ω , z którego wychodzą krawędzie do węzłów $\omega_1, \omega_2, \dots, \omega_{n(\omega)}$, jest jednoznacznie określony stopień ważenia ω' jako liczba:

$$v(\omega') = \frac{w(\omega')}{\sum_{i=1}^{n(\omega)} w(\omega'_i)} \quad (75)$$

Gdy z węzła ω' nie wychodzą do góry żadne krawędzie, to wartość funkcji ważenia można uważać za równą 0. Oznaczono przez $\omega(O_1, O_2)$ węzeł opisujący podobieństwo obiektów O_1 i O_2 . Współczynnik podobieństwa tych obiektów wyniesie $[\omega(O_1, O_2)]$. Zauważono, że

$[\omega(O_1, O_2)] = [\omega(O_2, O_1)]$. Z powyższego wzoru wynika, że podobieństwo nie zależy od kolejności porównywania.

Wskaźnik podobieństwa $[\omega_i, \omega]$ obiektu diagnozowanego do wzorcowego w krawędzi (ω_i, ω) zakończonej w węźle ω_i określamy za pomocą funkcji ważenia:

$$[\omega_i, \omega] = \frac{w(\omega_i)}{\sum_{i=1}^{n(\omega)} w(\omega_i)} \quad (76)$$

W celu określenia zbioru interpretacji rozmytych konceptów i ról sieci semantycznej podobieństwa obiektu diagnozowanego do obiektu wzorcowego przyjmujemy, że rozmywaną przestrzenią będzie zbiór $L(X)$ wszystkich zasobów internetowych (adresów tych zasobów), dla których badane jest podobieństwo. Wystąpieniami nazw konceptów są egzemplarze tych nazw, występujące w zasobie $x \in L(X)$, a wystąpieniami konceptów są egzemplarze nazw desygnatów nazw tych konceptów, występujące w zasobie x . Stopień rozmycia tych wystąpień w węzłach ω opisywanych przez nazwy tych wystąpień obliczamy dla zasobu x jako wskaźnik podobieństwa $[\omega]$. Wystąpienia ról obliczamy za pomocą funkcji ważenia.

Implementacja zakresu ufności rozmycia

Wskaźnik podobieństwa stopnia ważności wyszukiwania informacji diagnostycznej dla dwóch obiektów diagnozowanych (krótko: wskaźnik podobieństwa stopnia ważności dwóch obiektów) jest wyrażony następującym wzorem:

$$c(x, y) = \begin{cases} \frac{1-|x-y|}{x}, & \text{dla } x > y \\ \frac{1-|x-y|}{y}, & \text{dla } x < y \end{cases} \quad (77)$$

gdzie:

x – stopień ważności wyszukiwania informacji diagnostycznej o pierwszym obiekcie,

y – stopień ważności wyszukiwania informacji diagnostycznej o drugim obiekcie.

Oczywiście: $c(x, y) = c(y, x)$.

Stopień ważności wyszukiwania informacji (krótko: *stopień ważności informacji*) może być na przykład wyrażony przez ekspertów jako stopień odczucia wystąpienia diagnozowanego stanu: usterki, wady, błędu czy zaburzenia działania obiektu diagnozowanego O , o którym wyszukiwana jest informacja w sieci Web. Im ten stopień dla danego stanu jest niższy, tym ważność wyszukiwania o nim informacji jest niższa, czyli jest niższy stopień ważności tej informacji. Na przykład do zniszczenia obudowy kotłów gazowych dochodzi bardzo rzadko, zatem ważność szukania informacji o tym uszkodzeniu jest mała, rzadko dochodzi do szukania takiej informacji (por. stopień rozmycia stanu, tabela 3.6.).

Przyjmuje się, że wzorcowym obiektom O_1, O_2, \dots, O_k odpowiadają stopnie ważności: c_1, c_2, \dots, c_k wyszukiwania informacji diagnostycznej. Będą rozważane tylko wskaźniki podobieństwa ważności x, y , dla których $c(x, y) > \alpha_k$ (np. $\alpha_k = 0,9$), gdzie α_k jest wartością uznawaną przez ekspertów za minimalną wartość wskaźnika podobieństwa stopni ważności.

Jako wzorcowe pary obiektów O_i, O_j do porównań stopni ważności c_i, c_j przyjęto tylko takie pary, dla których spełniony jest warunek:

$$c(c_i, c_j) > 0,9 \text{ oraz } c_i \leq c_j \quad (78)$$

Założono, że można mieć zaufanie tylko do takiego obliczania wskaźnika podobieństwa $[\omega(O_i, O_j)]$ obiektów O_i, O_j , w którym przyjęto taką funkcję ważenia $v(\omega)$, że dla wszystkich wyróżnionych wzorcowych par (O_i, O_j) wskaźnik podobieństwa $[\omega(O_i, O_j)]$ różni się minimalnie od wskaźnika podobieństwa stopni ważności $c(c_i, c_j)$. Eksperti mogą za wiarygodne uznać np. kryterium:

$$\frac{|[\omega(O_i, O_j)] - c(c_i, c_j)|}{c(c_i, c_j)} < 0,1 \quad (79)$$

Oznacza to, że będą akceptować różnicę podobieństwa obiektów od podobieństwa stopni ważności w granicach 10%.

Wyróżnienie wskazanego wyżej zbioru funkcji ważenia pozwala wyznaczyć bazowy zbiór *Fuz* interpretacji rozmytych oraz wyznaczyć zbiór wszystkich wskazanych sposobów obliczania wskaźników podobieństwa dowolnego obiektu diagnozowanego O do dowolnego obiektu wzorcowego $O_i, i = 1, 2, \dots, k$. Dla dowolnego węzła ω sieci semantycznej klasyfikatora podobieństwa obiektu diagnozowanego O do obiektu wzorcowego O_i w węźle ω stopień ważenia w tym węźle jest *stopniem rozmycia roli* opisanej przez związek konceptu w węźle ω i konceptu w wyżej położonym węźle. Zgodnie z kryteriami przyjętymi przez ekspertów przyjmuje się, że *zakres ufności rozmycia konceptu C w węźle $\omega \neq \omega(O_i, O)$* jest zbiorem $V(C)$ wskaźników podobieństwa $[\omega]$, a dla $\omega = \omega(O_i, O)$, $[\omega] > 0,9$. Natomiast *zakres ufności rozmycia roli R wyznaczonej w węźle ω* jest zbiorem $V(R)$ wartości $v(\omega)$ pewnej wyróżnionej funkcji ważenia V .

Wyostrenie reprezentacji wiedzy wyszukiwanej z wykorzystaniem bazy wiedzy i reguł wnioskowania

Dysponując zakresem ufności rozmycia i wybierając bazowe terminy TBox, asercje ABox oraz aksjomaty RBox, na podstawie konceptów i ról konkretnej sieci semantycznej, można dokonać wyostrenia reprezentacji wyszukiwanej wiedzy diagnostycznej dla tych wyrażeń bazowych. Wyostrenia dokonuje się za pomocą zaproponowanych wcześniej wzorów **D1.–D13.** (rozd. 2.5.4.). Wtedy określa się, dla danej grupy wzorcowych obiektów diagnozowanych, zbiory zasobów reprezentujących wyszukiwaną informację. Są to te zasoby, których opisywane cechy mają stopień rozmycia zgodny z zakresem ufności przyjętym przez ekspertów.

Za adekwatne można uważać wszystkie opisy cech konstytutywnych obiektów diagnozowanych, a więc tych cech, bez których miara zgodności strukturalnych jest w danym węźle sieci równa zero. Stopień rozmycia takich cech musi wynosić 1, co odpowiada warunkowi adekwatności.

3.2. Mapowanie reprezentacji wiedzy w języku OWL DL

OWL (ang. *Web Ontology Language*) jest językiem opartym na składni XML. Semantyka tego języka z kolei jest oparta na logice opisowej DL. OWL stanowi rozszerzenie RDF (ang. *Resource Description Framework*). Języki RDF i OWL są bardzo podobne i dotyczą tego samego problemu, jednak OWL jest językiem bardziej rozbudowanym, z większym słownikiem i mocniejszą składnią. Oba te języki służą do reprezentacji i przetwarzania danych w sieci WWW na podstawie teorii sieci semantycznych. OWL został uznany w lutym 2004 roku za standard przez W3C. Obecnie od 2009 roku został wprowadzony standard OWL w wersji 2. OWL służy do opisywania danych w postaci ontologii i budowania w ten sposób tzw. Semantycznego Internetu.

Istnieją trzy odmiany języka OWL:

- **OWL Lite** – najprostsza wersja umożliwiająca tworzenie hierarchii klasyfikacji, umożliwia szybki dostęp do tezaursów i taksonomii,
- **OWL DL** (rozszerzenie OWL Lite) – wersja zgodna z logiką opisową DL oraz z rozmytą logiką opisową *fuzzyDL*, wspiera procesy wnioskujące,
- **OWL Full** (rozszerzenie OWL DL) – wersja pełna pozwalająca między innymi na to, aby indywiduum było jednocześnie klasą; niepolecana, jeżeli do ontologii ma być przeprowadzone wnioskowanie; trudno jest przeprowadzić proces wnioskujący, który uwzględniłby wszystkie aspekty, jakie są dozwolone w tej wersji OWL.

W pracy rozpatrywana jest wersja OWL DL, która jako jedyna w pełni wspiera logikę opisową DL. W dalszej części przez język OWL będzie rozumiana wersja OWL DL.

W tabeli 3.1. ukazano różnice pomiędzy słownictwem używanym w języku OWL a logiką opisową DL przedstawioną w podrozdziałach 2.4. i 2.5. Słownictwo OWL jest słownictwem informatycznym, natomiast słownictwo DL jest słownictwem używanym w przypadku opisu wyrażeń logicznych.

Tabela 3.1. Słownictwo używane w języku OWL i logiki opisowej DL. Opracowanie własne na podstawie [goc13].

Słownictwo języka ontologii OWL	Słownictwo logiki opisowej DL
Klasa	Koncept
Obiekt (indywiduum, instancja)	Osobnik, wystąpienie konceptu
Właściwość	Rola
Datatype	Dziedzina konkretna
Dziedzina właściwości	{dziedzina roli}
Zakres właściwości	{zakres roli}

W tabeli 3.2. przedstawiono wzajemną odpowiedniość języka OWL i logiki opisowej DL. Przedstawiono, w jaki sposób w języku OWL można zapisać odpowiednie wyrażenia z logiki DL.

Tabela 3.2. Wzajemna odpowiedniość logiki DL i języka OWL (konstruktory i aksjomaty). Opracowanie własne na podstawie [goc13].

Elementy OWL	Wyrażenie DL	Opis wyrażenia
intersectionOf	$C \sqcap D$	Przecięcie klas
unionOf	$C \sqcup D$	Suma klas
complementOf	$\neg C$	Negacja klasy
someValuesFrom	$\exists R. C$	Kwantyfikacja egzystencjalna
allValuesFrom	$\forall R. C$	Kwantyfikacja ogólna
minCardinality	$\geq nR.\top$	Minimalna liczebność
maxCardinality	$\leq nR.\top$	Maksymalna liczebność
inverseOf	R^-	Rola odwrotna
oneOf	$\{t_1, t_2, \dots, t_n\}$	Jeden z obiektów
hasValue	$\exists R. \{a\}$	Obiekt posiada wartość
subClassOf	$C_1 \sqsubseteq C_2$	Podklasa, klasa zawiera się w innej klasie
equivalentClass	$C_1 \equiv C_2$	Klasy równoważne
disjointWith	$C_1 \sqsubseteq \neg C_2$	Klasy rozłączne
subPropertyOf	$R_1 \sqsubseteq R_2$	Podrola, rola zawiera się w innej roli
equivalentProperty	$R_1 \equiv R_2$	Role równoważne
sameAs	$\{t_1\} \equiv \{t_2\}$	Takie same obiekty
differentFrom	$\{t_1\} \sqsubseteq \neg\{t_2\}$	Obiekty rozłączne
FunctionalProperty	$\top \sqsubseteq \leq 1R.\top$	Właściwość funkcjonalna
InverseFunctionalProperty	$\top \sqsubseteq \leq 1R^-. \top$	Odwrotna właściwość funkcjonalna
SymmetricProperty	$R \equiv R^-$	Rola równoważna roli odwrotnej (tj. ilekroć para osobników (a, b) jest wystąpieniem roli R, tylekroć para (b, a) jest wystąpieniem roli R ⁻)
TransitiveProperty	$R \equiv R^*$	Rola równoważna roli domkniętej przechodnio (tj. osobniki, między którymi przebiega dowolnie długi łańcuch roli R, są ze sobą w relacji R [*])

W kolejnych podrozdziałach 3.3. i 3.4. będzie używane słownictwo zgodne z językiem OWL.

3.3. Wybrane przykłady wyszukiwania wiedzy w sieciach semantycznych

W podrozdziale pokazano przykłady, które ilustrują opisane algorytmy tworzenia tezaury i ontologii oraz proces wyszukiwania wiedzy w sieciach semantycznych. Najpierw zostanie przedstawiony przykładowy klasyfikator konstrukcyjno-technologiczny dla sieci semantycznej

w systemie produkcji. Następnie przedstawiono fragment sieci semantycznej dla systemu diagnostyki oraz wnioskowanie na podstawie danych zapisanych w tej sieci.

3.3.1. Klasyfikator konstrukcyjno-technologiczny dla systemu produkcji

Używając wprowadzonego aparatu pojęciowego, przedstawimy metodę rozmywania wiedzy konstrukcyjno-technologicznej o elementach obrotowo-symetrycznych reprezentowanej w sieci semantycznej, zaproponowaną przez zespół R. Knosali [kno02]. Przykład ten stał się inspiracją do utworzenia procedury wyznaczania tezaurusa oraz dyskretyzacji wartości podobieństwa konceptów i ról do danych wzorcowych opisanych w podrozdziale 3.1.3.

Przykładowy schemat sieci semantycznej dla elementów obrotowo-symetrycznych znajduje się na rysunku 3.1.

Dla tej sieci można wyróżnić następujące koncepty i role [bry12b]. *Koncepty*:

- Wałek (koncept Top – wszystkie możliwe do produkcji elementy obrotowo-symetryczne)
 - postać zewnętrzną,
 - postać wewnętrzną,
 - powierzchnie elementarne,
 - postać dwustronnie stopniowana,
 - otwór czołowy,
 - i inne.

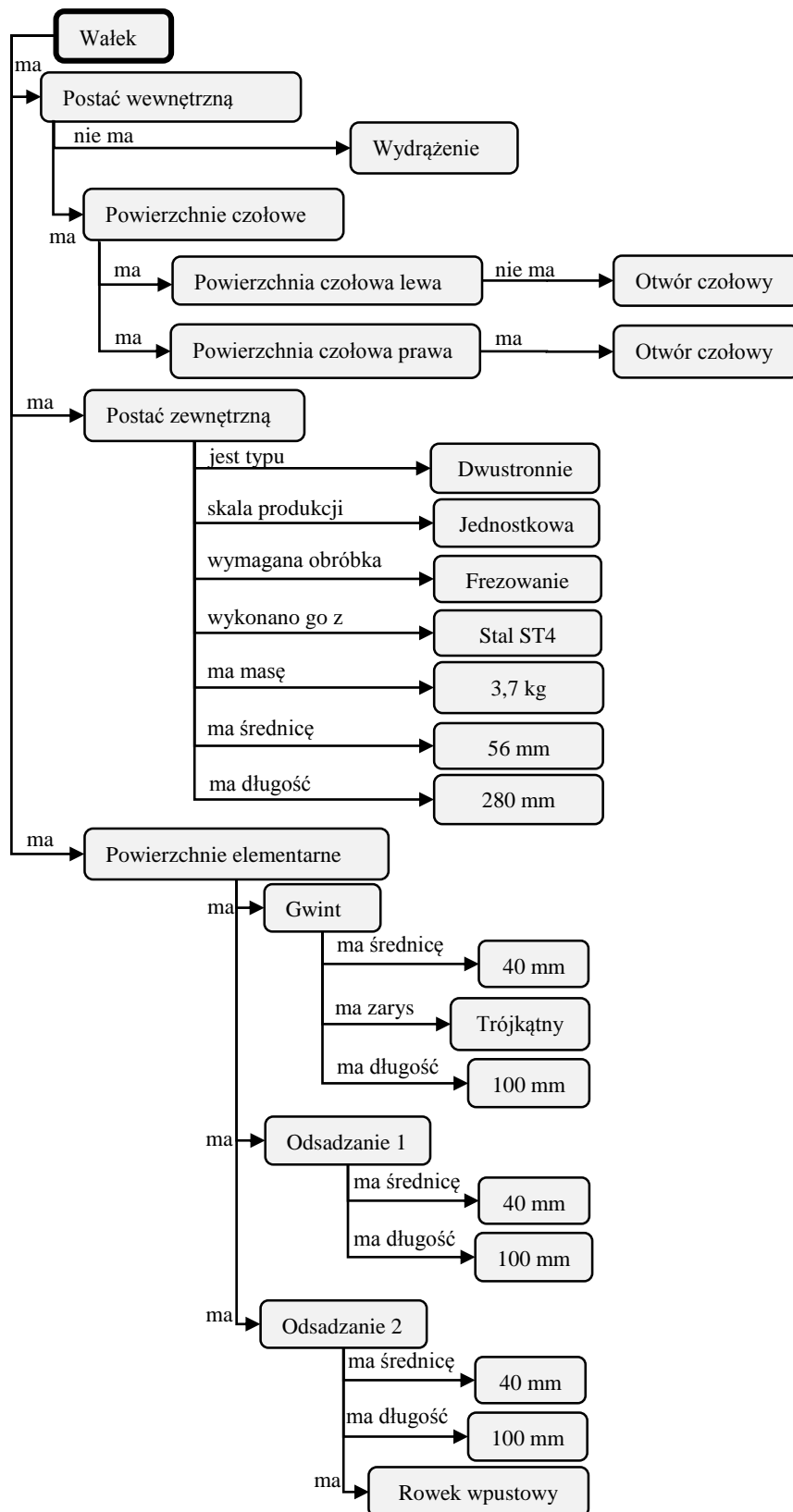
Konceptami są też odpowiednie zalecane w konstrukcji parametry techniczne.

Role: ma, nie ma, ma długość, jest typu, ma średnicę, ma masę, wykonano go z, skala produkcji, wymagana obróbka, ma zarys i inne.

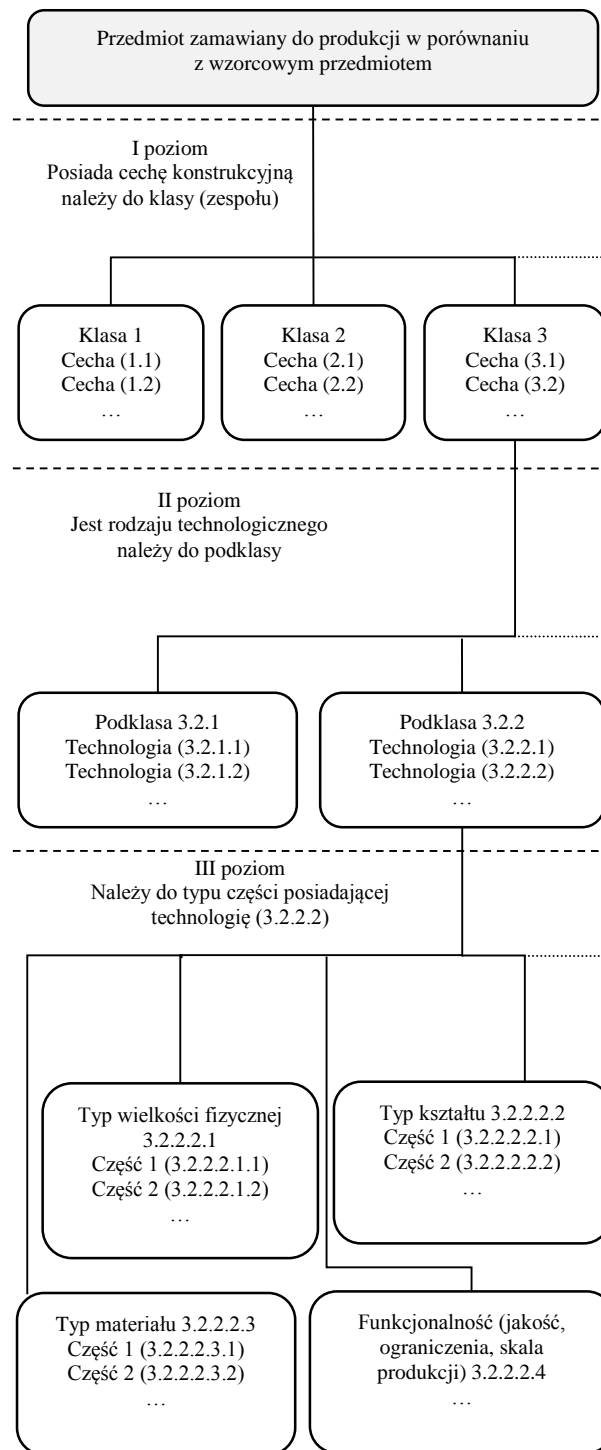
Dla tej sieci semantycznej buduje się schemat szkieletowy sieci semantycznej klasyfikatora konstrukcyjno-technologicznego według wzoru pokazanego na rysunku 3.2. Rodzina diagramów zbudowanych według tego wzoru dla konkretnych wzorcowych elementów produkcji jest siecią semantyczną w danym systemie produkcji [bry12b].

Dla przedmiotów wzorcowych $P_1, P_2, \dots, P_i, \dots, P_k$ oraz dowolnego przedmiotu P_x określone są koncepty podobieństwa dowolnego przedmiotu do przedmiotu P_i w węźle ω semantycznej sieci szkieletowej przedmiotu P_i . Rolami są opisane w tej sieci związki konceptu w węźle ω i konceptu w wyżej położonym węźle ω' .

Nazwami konceptów są nazwy określone dla ustalonych przedmiotów wzorcowych P_i i węzła ω sieci szkieletowej tego przedmiotu za pomocą schematu: „przedmiot podobny do przedmiotu P_i w węźle ω ”. Nazwami ról są nazwy związków określone według schematu „opisane w sieci szkieletowej przedmiotu P_i związki konceptu w węźle ω i konceptu w węźle wyżej położonym ω' ”. Wystąpieniami konceptów są indywidualne nazwy konkretnego zamawianego przedmiotu P_x podobnego do konkretnego wzorcowego przedmiotu P_i , natomiast wystąpieniami ról są stosowne pary wystąpień powiązanych rolą konceptów.



Rysunek 3.2. Przykładowa sieć semantyczna opisująca element typu Walek [kno02].



**Rysunek 3.3. Schemat szkieletowy sieci semantycznej klasyfikatora konstrukcyjno-technologicznego.
Opracowanie własne.**

Wskaźnik podobieństwa kosztów dwóch obiektów jest wyrażony wzorem (77), gdzie: x – koszt pierwszego przedmiotu, y – koszt drugiego przedmiotu. Przyjmijmy, zgodnie z opisaną

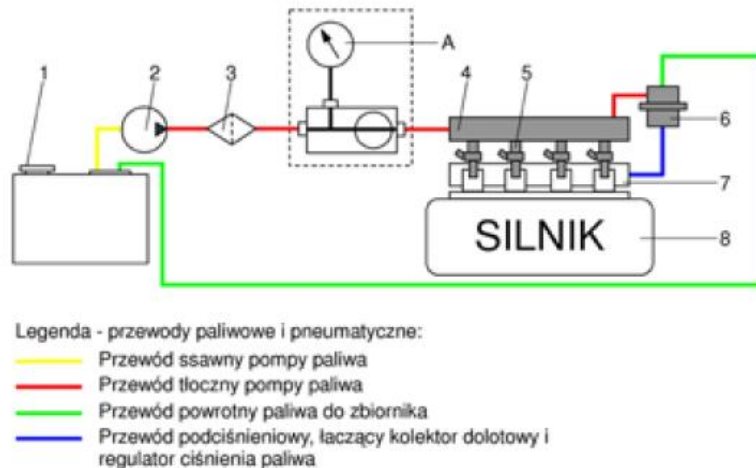
wcześniej koncepcją (rozdz. 3.1.3.), że wzorcowym przedmiotom P_1, P_2, \dots, P_k odpowiadają koszty: c_1, c_2, \dots, c_k . Będą rozważane tylko wskaźniki podobieństwa kosztów x, y , dla których $c(x, y) > \alpha_k$ (np. $\alpha_k = 0,9$), gdzie α_k jest wartością uznawaną przez ekspertów za minimalną wartość wskaźnika podobieństwa kosztów, dla którego różnice kosztów są nieistotne. Jako wzorcowe pary wzorcowych przedmiotów P_i, P_j do porównań kosztów c_i, c_j przyjmie się tylko takie pary, dla których spełniony jest warunek (78).

Zakłada się, że można mieć zaufanie tylko do takiego obliczania wskaźnika podobieństwa przedmiotów, w którym przyjęto taką funkcję ważenia $v(\omega)$, że dla wszystkich wyróżnionych wzorcowych par (P_i, P_j) wskaźnik podobieństwa $[\omega(P_i, P_j)]$ różni się minimalnie od wskaźnika podobieństwa kosztów $c(c_i, c_j)$. Eksperti mogą za wiarygodne uznać np. kryterium (79), czyli będą akceptować różnicę podobieństwa przedmiotów od podobieństwa kosztów w granicach 10% [kno02].

Wyróżnienie wskazanego wyżej zbioru funkcji ważenia pozwala wyznaczyć bazowy zbiór *Fuz* interpretacji rozmytych oraz wyznaczyć zbiór wszystkich wskazanych sposobów obliczania wskaźników podobieństwa dowolnego przedmiotu P do dowolnego przedmiotu wzorcowego P_i . Dla dowolnego węzła ω sieci semantycznej klasyfikatora podobieństwa przedmiotu P do przedmiotu wzorcowego P_i w węźle ω stopień ważenia w tym węźle jest *stopniem rozmycia roli* opisanej przez związek konceptu w węźle ω i konceptu w wyżej położonym węźle. Zgodnie z kryteriami przyjętymi przez ekspertów przyjmuje się, że *zakres ufności rozmycia konceptu C w węźle $\omega \neq \omega(P_i, P)$* jest zbiorem $V(C)$ wskaźników podobieństwa $[\omega]$. Natomiast *zakres ufności rozmycia roli R wyznaczonej w węźle ω* jest zbiorem $V(R)$ wartości $v(\omega)$ pewnej wyróżnionej funkcji ważenia.

3.3.2. Implementacja w języku OWL DL pomiaru ciśnienia sprężania w silniku samochodowym

Zdecydowana większość pośrednich układów wtrysku benzyny w silnikach samochodowych nie ma czujnika ciśnienia paliwa zasilającego wtryskiwacza, a więc sterownik nie zna jego aktualnej wartości. System diagnostyczny sterownika nie jest również w stanie wykryć uszkodzeń typu hydraulicznego w układzie paliwowym (nieszczelności, wzrost oporów przepływu paliwa przez wtryskiwacze). Ma też trudności z wykryciem uszkodzeń typu elektrycznego wtryskiwaczy, jeśli uszkodzenie nie jest jednoznaczne, np. przerwanie obwodu cewki wtryskiwacza, a jest to tylko jej przebicie [mys10]. W celach diagnostycznych dokonuje się pomiaru ciśnienia sprężania. Na rysunku 3.4. pokazano układ paliwowy przygotowany do takiego pomiaru. Na rysunku: 1 oznacza zbiornik paliwa (w zbiorniku jest jeszcze niepokazany wstępny filtr ssawny); 2 – elektryczna pompa paliwowa; 3 – dokładny filtr paliwa; 4 – listwa paliwowa zasilająca wtryskiwacze; 5 – wtryskiwacze; 6 – regulator ciśnienia paliwa; 7 – kolektor dolotowy; 8 – silnik.



Rysunek 3.4. Schemat układu paliwowego pośredniego, wielopunktowego, powrotnego układu wtryskowego, przygotowanego do pomiaru ciśnienia [mys10].

Jeśli zmierzone ciśnienie jest za **niskie**, to może być to spowodowane różnymi czynnikami [mys10]:

- Uszkodzeniem zaworu przelewowego w pompie paliwa – dla sprawdzenia tej ewentualności należy wykonać pomiar maksymalnego ciśnienia tłoczenia pompy paliwowej.
- Uszkodzeniem regulatora ciśnienia paliwa.
- Niektóre układy posiadają tłumik pulsacji ciśnienia, montowany pomiędzy pompą paliwa a listwą paliwową, jego uszkodzenie lub rozregulowanie (zasadniczo nie podlega on regulacji, choć w niektórych silnikach ma śrubę regulacyjną) może również spowodować za niskie ciśnienie tłoczenia.

Jeśli zmierzone ciśnienie jest za **wysokie**, to może być to spowodowane różnymi czynnikami [mys10]:

- Uszkodzeniem regulatora ciśnienia paliwa.
- Niedrożnym przewodem powrotnym paliwa z regulatora ciśnienia paliwa do zbiornika paliwa.
- Jeśli dany układ posiada tłumik pulsacji ciśnienia, montowany na przewodzie powrotnym paliwa z regulatora ciśnienia paliwa do zbiornika paliwa, to jego niesprawność może również spowodować wzrost ciśnienia paliwa.
- Jeśli dany układ posiada tłumik pulsacji ciśnienia, to jego uszkodzenie lub rozregulowanie może również spowodować wzrost ciśnienia tłoczenia.

Zmierzone ciśnienie może wskazywać na różne usterki w badanym układzie, jednak dla przykładu zostanie pokazana diagnostyka, czy uszkodzeniu uległ reduktor ciśnienia paliwa. Poniżej przedstawiono opis układu opisanego za pomocą OWL w celu diagnostyki uszkodzenia reduktora ciśnienia paliwa.

W pierwszej kolejności należy utworzyć klasę silnik, która będzie określać badany obiekt diagnozowany, jakim jest układ paliwowy w silniku samochodowym, oraz klasę pomiar ciśnienia, określającą przeprowadzony pomiar.

```
<owl:Class rdf:ID="Silnik"/ >
<owl:Class rdf:ID="PomiarCisnienia"/>
```

Następnie klasy silnik i pomiar ciśnienia zostają połączone ze sobą rolą (właściwością) „ma badanie”, oraz rolą odwrotną „dotyczy silnika”. Dzięki tej roli wiadomo, że dany silnik ma badanie pomiaru ciśnienia.

```
<owl:ObjectProperty rdf:ID="maBadanie">
<owl:domain rdf:resource="#Silnik"/>
<owl:range rdf:resource="#PomiarCisnienia"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="dotyczySilnika">
<owl:domain rdf:resource="#PomiarCisnienia"/>
<owl:range rdf:resource="#Silnik"/>
<owl:inverseOf rdf:resource="#maBadanie"/>
</owl:ObjectProperty>
```

Opisujemy również klasę „wynik pomiaru ciśnienia” oraz rolę „ma wynik”, która łączy klasę pomiar ciśnienia z klasą wynik pomiaru ciśnienia. Wynik pomiaru ciśnienia jest klasą, która przyjmuje wartości z kolekcji: „Niski”, „Normalny”, „Wysoki”.

```
<owl:Class rdf:ID="WynikPomiaruCisnienia">
<owl:oneOf rdf:parseType="Collection">
<owl:Thing rdf:about="#Niski"/>
<owl:Thing rdf:about="#Normalny"/>
<owl:Thing rdf:about="#Wysoki"/>
</owl:oneOf>
</owl:Class>
<owl:ObjectProperty rdf:ID="maWynik">
<owl:domain rdf:resource="#PomiarCisnienia"/>
<owl:range rdf:resource="#WynikPomiaruCisnienia"/>
</owl:ObjectProperty>
```

Opisana zostaje również klasa „pomiar ciśnienia wykazujący nadciśnienie”, która występuje, gdy pomiar ciśnienia ma wynik „wysoki”.

```
<owl:Class rdf:ID="PomiarCisnieniaWykazujacyNadcisnienie">
<owl:intersectionOf rdf:parseType="Collection">
<owl:Class rdf:about="#PomiarCisnienia"/>
<owl:Restriction>
<owl:onProperty rdf:resource="#maWyniki"/>
<owl:hasValue rdf:resource="#Wysoki"/>
</owl:Restriction>
</owl:intersectionOf>
</owl:Class>
```

Klasa „silnik zagrożony usterką regulatora ciśnienia” występuje, gdy występuje pomiar ciśnienia w silniku, a wynik badania ma pomiar ciśnienia wykazujący nadciśnienie.

```
<owl:Class rdf:ID="SilnikZagrozonyUsterkaRegulatoraCisnienia">
<owl:intersectionOf rdf:parseType="Collection">
<owl:Class rdf:about="#Silnik"/>
<owl:Restriction>
<owl:onProperty rdf:resource="#maBadanie"/>
<owl:someValuesFrom rdf:resource=
"#PomiarCisnieniaWykazujacyNadcisnienie"/>
</owl:Restriction>
</owl:intersectionOf>
</owl:Class>
```

Z opisanej wyżej ontologii za pomocą języka OWL można zauważyć, że wynikają pewne fakty:

- 1) Właściwości (role) „ma badanie” i „dotyczy silnika” są odwrotne: $\text{maBadanie}(X, Y) \Leftrightarrow \text{dotyczySilnika}(Y, X)$.
- 2) Klasa „pomiar ciśnienia wykazujący nadciśnienie” zachodzi wtedy, gdy dla danego obiektu, silnika, zostanie wykonany pomiar ciśnienia i wynik tego pomiaru będzie wysoki: $\text{PomiarCisnieniaWykazujacyNadcisnienie}(X) \Leftrightarrow \text{PomiarCisnienia}(X) \wedge \text{maWynik}(X, \#Wysoki)$.
- 3) Klasa „silnik zagrożony usterką regulatora ciśnienia” zachodzi, gdy dla pewnego silnika (obiekту) zostanie wykonane przynajmniej jedno badanie, a wynik tego badania będzie wysoki, czyli będzie spełniał klasę „pomiar ciśnienia wykazujący nadciśnienie”: $\text{SilnikZagrozonyUsterkaRegulatoraCisnienia}(X) \Leftrightarrow \text{Silnik}(X) \wedge \exists Y(\text{maBadanie}(X, Y) \wedge \text{PomiarCisnieniaWykazujacyNadcisnienie}(Y))$.

Założmy, że mamy konkretny silnik samochodowy o numerze identyfikacyjnym pojazdu VIN: 1M8GDM9A_KP042788. Zostały również wykonane dwa pomiary ciśnienia dla tego silnika:

- A. $\text{Silnik}(\#VIN1M8GDM9A_KP042788)$
- B. $\text{maBadanie}(\#VIN1M8GDM9A_KP042788, \#BadanieJK1)$
- C. $\text{PomiarCisnienia}(\#BadanieJK1)$
- D. $\text{maWynik}(\#BadanieJK1, \#Normalny)$
- E. $\text{PomiarCisnienia}(\#BadanieJK2)$
- F. $\text{dotyczySilnika}(\#BadanieJK2, \#VIN1M8GDM9A_KP042788)$
- G. $\text{maWynik}(\#BadanieJK2, \#Wysoki)$

Powyższe obiekty zostały zapisane za pomocą OWL:

```
<Silnik rdf:ID="#VIN1M8GDM9A_KP042788">
<maBadanie rdf:resource="#BadanieJK1"/>
</Silnik>
<PomiarCisnienia rdf:ID="BadanieJK1">
<maWynik rdf:resource="#Normalny"/>
</PomiarCisnienia>
```

```
<PomiarCisnienia rdf:ID=„BadanieJK2”>
  <dotyczySilnika rdf:resource=“#VIN1M8GDM9A_KP042788”/>
  <maWynik rdf:resource=“#Wysoki”/>
</PomiarCisnienia>
```

Na tej podstawie możemy odpowiedzieć na pytanie, czy silnik jest zagrożony usterką regulatora ciśnienia, czyli czy: `SilnikZagrozonyUsterkaRegulatoraCisnienia(#VIN1M8GDM9A_KP042788)`? W tym celu należy przeprowadzić wnioskowanie na podstawie opisanej ontologii.

Zauważmy, że `BadanieJK1` ma wynik „normalny”, dlatego to badanie na pewno nie wskazuje na poszukiwaną usterkę. Proces wnioskowania przeprowadzimy dla badania `JK2`. Z punktu F. i 1) można zauważyć, że jeżeli silnik ma badanie, to badanie dotyczy tego silnika:

```
dotyczySilnika(#BadanieJK2, #VIN1M8GDM9A_KP042788) ⇒
maBadanie(#VIN1M8GDM9A_KP042788, #BadanieJK2)
```

Następnie należy określić, czy pomiar ciśnienia wykazuje nadciśnienie:

```
PomiarCisnieniaWykazujacyNadcisnienie(#BadanieJK2)
```

Z punktów E., G. i 2) wynika, że rzeczywiście pomiar `BadanieJK2` wykazuje nadciśnienie:

```
PomiarCisnienia(#BadanieJK2) ∧ maWynik(#BadanieJK2, #Wysoki) ⇒
PomiarCisnieniaWykazujacyNadcisnienie(#BadanieJK2)
```

Ostatnim etapem wnioskowania jest odpowiedź na pytanie, czy silnik jest zagrożony usterką regulatora ciśnienia:

```
SilnikZagrozonyUsterkaRegulatoraCisnienia(#VIN1M8GDM9A_KP042788)
```

Z punktów A. i 3) oraz powyższych ustaleń można zauważyć:

```
Silnik(#VIN1M8GDM9A_KP042788) ∧ maBadanie(#VIN1M8GDM9A_KP042788,
#BadanieJK2) ∧ PomiarCisnieniaWykazujacyNadcisnienie(#BadanieJK2) ⇒
SilnikZagrozonyUsterkaRegulatoraCisnienia(#VIN1M8GDM9A_KP042788)
```

Z czego wynika, że silnik jest zagrożony usterką regulatora ciśnienia.

Powyższy przykład ilustruje, w jaki sposób można zapisać proces diagnostyczny za pomocą języka OWL. Został również opisany proces wnioskowania na podstawie takiego zapisu. Oczywiście, należałoby także uwzględnić inne usterki badanego obiektu, które mogą wystąpić, gdy ciśnienie jest wysokie. Można by również rozszerzyć przykład o opis badanego silnika lub przypadek, gdy ciśnienie jest niskie itd.

Należy również zauważyć, że w opisanym przykładzie do opisu wyniku pomiaru ciśnienia zostały wykorzystane zmienne lingwistyczne: „Niski”, „Normalny”, „Wysoki”, do których można określić funkcję przynależności określającą rozmyty charakter tych określeń.

3.4. Przykład wyszukiwania wiedzy diagnostycznej

W podrozdziale, jako przykład systemu wyszukującego wiedzę z zakresu diagnostyki technicznej z zastosowaniem rozmytej logiki opisowej, został pokazany system diagnostyki

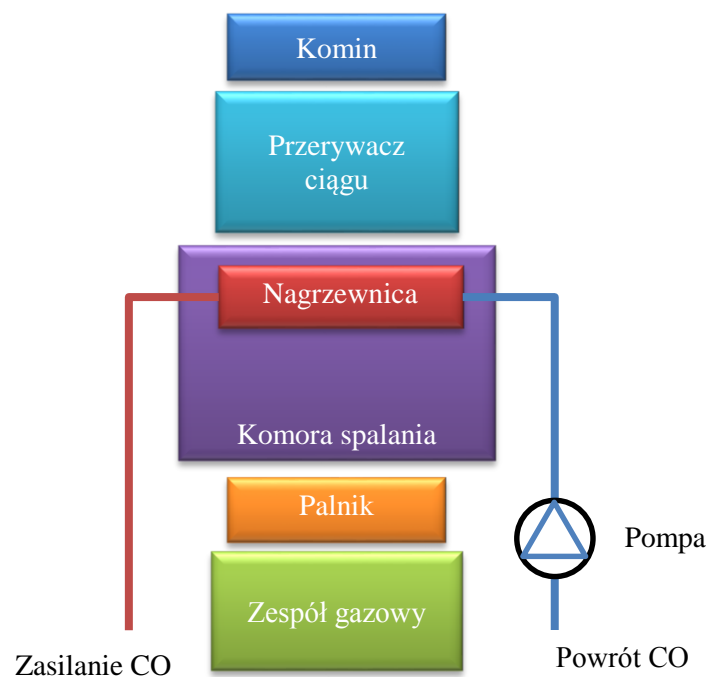
kotłów gazowych małej mocy poniżej 60 kW. Zadaniem systemu jest, na podstawie obserwowanych w urządzeniu symptomów, wyszukanie odpowiednich stanów urządzenia odpowiadających podanym symptomom, a następnie wybranie właściwego stanu urządzenia oraz określenie części urządzenia, które mogły ulec awarii.

Wiedza przetwarzana w systemie jest wiedzą wyrażoną za pomocą zmiennych lingwistycznych, czyli wszystkie określenia są wyrażone w języku naturalnym. Sprawia to, że prezentowana diagnoza jest zrozumiała dla użytkownika systemu, nawet jeśli nie jest on specjalistą w tej dziedzinie.

3.4.1. Opis diagnozowanego obiektu

Kotły rozpatrywane w systemie to kotły na paliwo gazowe, małej mocy poniżej 60 kW. Zwykle kotły używane w gospodarstwach domowych mają moc 13-24/30 kW, rzadko spotyka się kotły o wyższej mocy do zastosowań indywidualnych. W systemie brane są pod uwagę kotły klasyczne (niekondensacyjne) z ciągiem grawitacyjnym i wymuszonym, sterowanym procesorem [kry00], [rat01].

Kotły te mogą być jednofunkcyjne, wtedy służą jedynie do podgrzewania wody w centralnym ogrzewaniu (skr. CO), oraz dwufunkcyjne, wtedy oprócz ogrzewania CO zapewniają centralne ogrzewanie wody użytkowej (skr. CWU). Oba typy kotłów będą uwzględnione w systemie.



Rysunek 3.5. Schemat budowy kotła gazowego jednofunkcyjnego. Opracowanie własne.

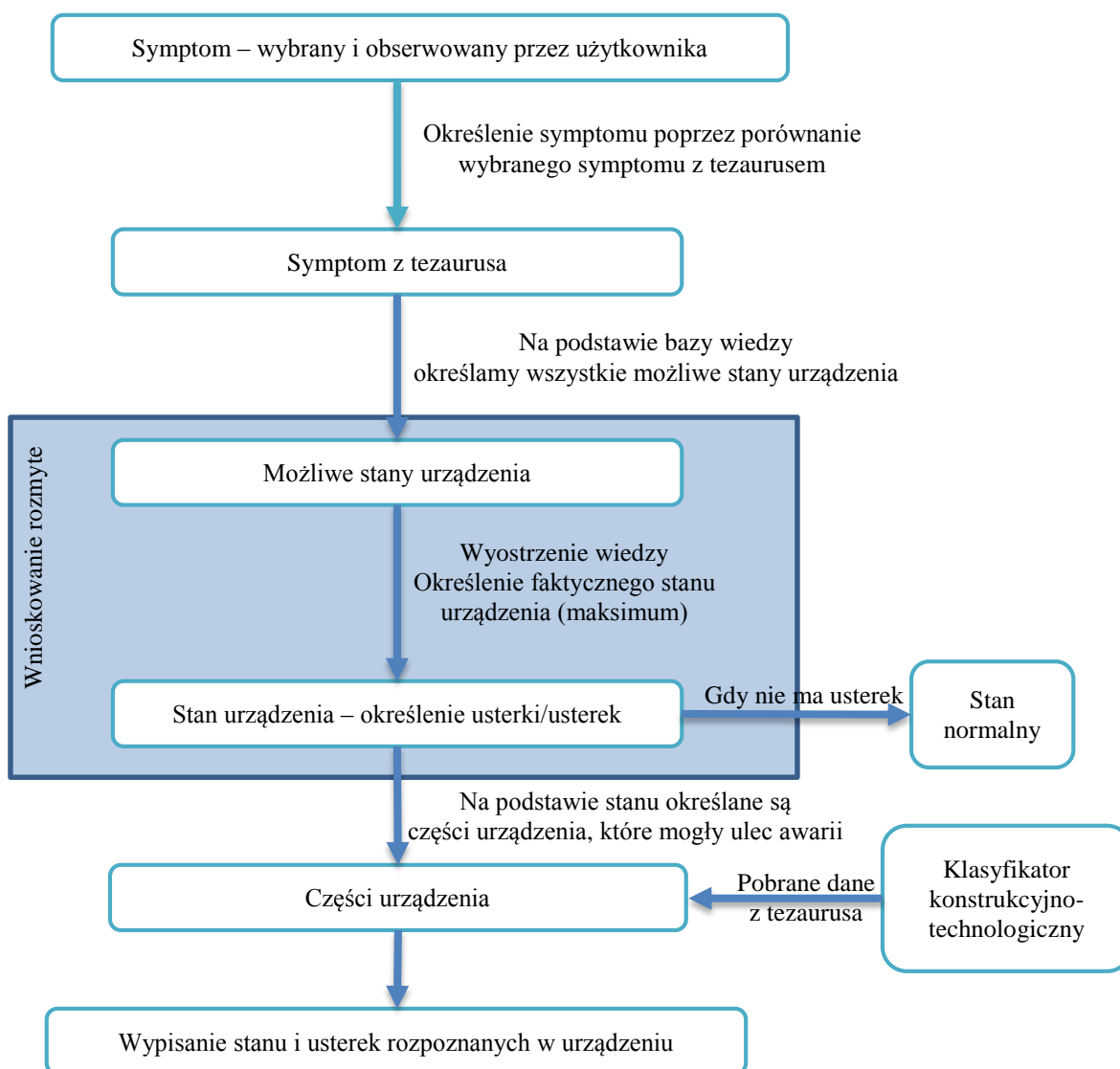
Należy jednak zaznaczyć, że system diagnostyki kotłów może zostać rozszerzony o inne typy kotłów gazowych, np. o kotły kondensacyjne, a także o kotły z innymi rodzajami

spalanego paliwa, na przykład kotły olejowe czy na paliwo stałe. Równie łatwo można uzupełnić wiedzę diagnostyczną w systemie o podobne urządzenia, np. termy wodne [kry00].

Na rysunku 3.5. przedstawiono schemat kotła gazowego jednofunkcyjnego wraz z głównymi elementami występującymi w tym urządzeniu. W przypadku kotła dwufunkcyjnego dodatkowo znajduje się zespół wodny centralnego ogrzewania wody użytkowej. Wtedy podgrzanie wody użytkowej następuje we wtórnym wymienniku ciepła, a do przełączenia pomiędzy trybami CO i CWU służy zawór 3-drogowy.

3.4.2. Schemat realizacji projektu

Na rysunku 3.6. przedstawiono schemat działania aplikacji.



Rysunek 3.6. Schemat działania systemu wyszukującego wiedzę z diagnostyki kotłów gazowych. Opracowanie własne.

Najpierw zostanie ustalony rodzaj kotła, który ma zostać poddany procesowi diagnostycznemu. Aplikacja będzie rozpatrywać kotły gazowe jedno- i dwufunkcyjne oraz z otwartą i zamkniętą komorą spalania. Możliwe jest również niepodanie rodzaju kotła, wtedy system będzie brał pod uwagę wszystkie możliwe przypadki. Następnie zostanie podany przez użytkownika symptom, który jest obserwowany w kotle. Na tej podstawie zostaną ustalone stany, w jakich może znajdować się kocioł, czyli możliwe usterki, wady lub awarie. W procesie wnioskowania, podczas którego będą zadawane dodatkowe pytania użytkownikowi, zostanie ustalony stan, o najwyższym stopniu rozmycia, w jakim znajduje się kocioł. Ze względu na rozmyty charakter związków pomiędzy symptomami a stanami może się okazać, że w procesie wnioskowania zostanie wybranych kilka stanów, które będą najbardziej prawdopodobne.

Na podstawie stanu (lub wybranych stanów) kotła zostaną określone części kotła, które mogły ulec awarii. Zostaną również wymienione inne czynniki zewnętrzne, które mogły zaistnieć i spowodować, że kocioł znajduje się w takim stanie.

Wszystkie dane o symptomach, stanach i częściach urządzenia oraz powiązania pomiędzy tymi danymi są zapisane w języku ontologii OWL za pomocą sieci semantycznej.

3.4.3. Tabela diagnostyczna reprezentująca rozmyte dane diagnostyczne

Wiedza ekspercka pozwala ustalić możliwe symptomy, które mogą zostać zaobserwowane w kotle gazowym. Ze względu na swoją charakterystykę zostały one podzielone na trzy grupy: ogólny objaw działania, działanie palnika i wydajność kotła. Poniżej przedstawiono symptomy, które będą rozpatrywane w systemie.

- 1) Ogólny objaw działania (S1):
 - (s11) Kocioł się nie włącza (w kotle panuje cisza);
 - (s12) Szumy, trzaski, dziwne dźwięki dobiegają z kotła;
 - (s13) Kocioł sam się wyłącza;
 - (s14) Kocioł działa w CO, ale nie działa w CWU;
 - (s15) Czuć zapach gazu (Czuć dziwny „zapaszek”, który zwykle określamy: „gaz się ulatnia”);
 - (s16) Gorąca obudowa (Obudowa robi się nadmiernie gorąca);
 - „-” Brak objawu.
- 2) Działanie palnika (S2):
 - (s21) Słysząc lub widząc iskrę, ale palnik się nie zapala;
 - (s22) Zapala się palnik, po kilku sekundach gaśnie;
 - (s23) Palnik gaśnie po dłuższym czasie;
 - (s24) Palnik działa, ale jest mały płomień;
 - „-” Brak objawu niewłaściwego działania palnika.
- 3) Wydajność kotła (S3):
 - (s31) Słaba wydajność kotła (CO);
 - (s32) Brak ciepłej wody lub zbyt niska temperatura CWU;
 - „-” Brak objawu zmiany wydajności.

Również na podstawie wiedzy eksperckiej zostały określone stany diagnostyczne, które mogą wystąpić w kotle gazowym. Stany te określają pewne usterki, wady lub awarie kotła, natomiast stan oznaczający prawidłowe działanie kotła nie został uwzględniony, gdyż system będzie rozważać tylko nieprawidłowe działanie kotła. Natomiast w przypadku, gdy kocioł pracuje prawidłowo, nie będą obserwowane żadne z powyżej wymienionych symptomów. Stany diagnostyczne rozpatrywane w systemie to:

- (F1) Brak zasilania (problem z bezpiecznikami, zwarcie);
- (F2) Zadziałanie zabezpieczenia kotła;
- (F3) Uszkodzona elektronika (moduł elektroniczny, części elektroniczne);
- (F4) Uszkodzony licznik gazowy;
- (F5) Uszkodzony reduktor gazowy;
- (F6) Wadliwie wykonany komin;
- (F7) Zapowietrzony grzejniki;
- (F8) Brak dopływu gazu (lub niskie ciśnienie gazu);
- (F9) Brak wody w instalacji lub kotle (lub niskie ciśnienie);
- (F10) Uszkodzona pompa cyrkulacyjna;
- (F11) Zatkany, zabrudzony filtr wody CO;
- (F12) Zatkany, zabrudzony filtr wody CWU;
- (F13) Zabrudzona lub uszkodzona nagrzewnica;
- (F14) Zapowietrzony kocioł;
- (F15) Uszkodzony zespół gazowy;
- (F16) Nieszczelność na kotle;
- (F17) Uszkodzona elektroda zapłonowa;
- (F18) Uszkodzona elektroda jonizacyjna;
- (F19) Uszkodzony lub zanieczyszczony zespół wodny;
- (F20) Przegrzew kotła;
- (F21) Uszkodzony czujnik temperatury wody grzewczej;
- (F22) Zatkane filtry, dysze lub elementy palnika;
- (F23) Zanieczyszczony wtórny wymiennik ciepła.

Wykorzystując wymienione symptomy i stany diagnostyczne kotła gazowego, opracowano tabelę 3.3., w której pokazano, jakie stany mogą wystąpić w przypadku obserwowania poszczególnych symptomów. Na tej podstawie można stworzyć tabelę diagnostyczną reprezentującą rozmyte dane diagnostyczne o diagnozowanych kotłach gazowych dla rozłącznych symptomów.

Tabela 3.3. Tablica obserwacji symptomów i odpowiadających im cech stanu diagnostycznego. Opracowanie własne.

Kategoria symptomów	Symptom	Stan diagnostyczny – usterka, wada, uszkodzenie
Ogólny objaw działania	Kocioł się nie włącza	<ul style="list-style-type: none"> • Brak zasilania (problem z bezpiecznikami, zwarcie) • Zadziałanie zabezpieczenia kotła • Uszkodzona elektronika (moduł elektroniczny, części elektroniczne) • Uszkodzony licznik gazowy • Uszkodzony reduktor gazowy • Wadliwie wykonany komin • Zapowietrzone grzejniki • Brak dopływu gazu (lub niskie ciśnienie gazu) • Brak wody w instalacji lub kotle (lub niskie ciśnienie)
	Szumy, trzaski, dziwne dźwięki dobiegają z kotła	<ul style="list-style-type: none"> • Uszkodzona pompa cyrkulacyjna • Zatkany, zabrudzony filtr wody CO • Zatkany, zabrudzony filtr wody CWU • Zabrudzona lub uszkodzona nagrzewnica • Zapowietrzone grzejniki
	Kocioł sam się wyłącza	<ul style="list-style-type: none"> • Wadliwie wykonany komin • Zapowietrzone grzejniki
	Kocioł działa w trybie CO, ale nie działa w trybie CWU	<ul style="list-style-type: none"> • Zapowietrzony kocioł
	Czuć zapach gazu	<ul style="list-style-type: none"> • Zabrudzona lub uszkodzona nagrzewnica • Uszkodzony zespół gazowy • Nieszczelność na kotle
	Gorąca obudowa	<ul style="list-style-type: none"> • Zabrudzona lub uszkodzona nagrzewnica • Uszkodzony reduktor gazowy • Nieszczelność na kotle
Działanie palnika	Słychać lub widać iskrę, ale palnik się nie zapala	<ul style="list-style-type: none"> • Uszkodzona elektronika (moduł elektroniczny, części elektroniczne) • Uszkodzony zespół gazowy • Uszkodzona elektroda zapłonowa
	Zapala się palnik, po kilku sekundach gaśnie	<ul style="list-style-type: none"> • Uszkodzony lub zanieczyszczony zespół wodny • Uszkodzony zespół gazowy • Uszkodzona elektroda jonizacyjna
	Palnik gaśnie po dłuższym czasie	<ul style="list-style-type: none"> • Przegrzew kotła • Uszkodzony czujnik temperatury wody grzewczej • Zadziałanie zabezpieczenia kotła

		<ul style="list-style-type: none"> • Uszkodzony lub zanieczyszczony zespół wodny • Uszkodzona pompa cyrkulacyjna • Uszkodzony zespół gazowy
	Palnik działa, ale jest mały płomień	<ul style="list-style-type: none"> • Uszkodzony zespół gazowy • Zatkane filtry, dysze lub elementy palnika
Wydajność kotła	Słaba wydajność kotła (CO)	<ul style="list-style-type: none"> • Uszkodzony czujnik temperatury wody grzewczej • Zatkany, zabrudzony filtr wody CO • Zabrudzona lub uszkodzona nagrzewnica
	Brak ciepłej wody lub zbyt niska temperatura CWU	<ul style="list-style-type: none"> • Zatkany, zabrudzony filtr wody CWU • Zabrudzona lub uszkodzona nagrzewnica • Zanieczyszczony wtórny wymiennik ciepła

W tabeli 3.4. przedstawiono możliwe kody błędów, które mogą zostać wyświetlone dla kotła firmy Ariston model Microgenus Plus. W zależności od producenta i modelu kody błędów są różne i mogą oznaczać inne błędy. Z tego powodu kody błędów nie zostały uwzględnione w tabeli diagnostycznej, a jedynie są wymienione poniżej. W trakcie rozwoju systemu można oczywiście uzupełnić tabelę diagnostyczną o interpretacje kodów błędów w zależności od modelu i producenta urządzenia. Nie jest to jednak konieczne ze względu na jednoznaczny charakter tych kodów (określony kod pojawia się w jednoznacznie określonej sytuacji) oraz łatwość dostępu do informacji o poszczególnych kodach (takie informacje znajdują się w instrukcji obsługi danego kotła gazowego).

Tabela 3.4. Kody błędów – zabezpieczenia występujące w kotle Ariston Microgenus Plus [mic14].

Komunikat na wyświetlaczu	Znaczenie
Kod błędu E02	Za małe ciśnienie wody – brak obiegu wody w układzie – blokada mijająca
Kod błędu E04	Sonda CWU w rozwarciu – blokada mijająca
Kod błędu E05	Sonda CWU w zwarcu – blokada mijająca
Kod błędu E06	Sonda CO na zasilaniu w rozwarciu – blokada mijająca
Kod błędu E07	Sonda CO na zasilaniu w zwarcu – blokada mijająca
Kod błędu E08	Sonda CO na powrocie w rozwarciu – blokada mijająca
Kod błędu E09	Sonda CO na powrocie w zwarcu – blokada mijająca
Kod błędu E20	Wykryty płomień bez otwarcia zaworu gazowego – blokada mijająca
Kod błędu E30	Rozwarcie w obwodzie sondy zewnętrznej – blokada mijająca
Kod błędu E31	Zwarcie w obwodzie sondy zewnętrznej – blokada mijająca
Kod błędu E33	Styki w presostacie spalin zamknięte przed startem wentylatora – presostat zamknięty przed rozpoczęciem zapłonu kotła – blokada mijająca
Kod błędu E34	Styki w presostacie spalin otwarte przy pracującym wentylatorze – brak zamknięcia presostatu spalin – blokada mijająca

Kod błędu E32	Czujnik temperatury spalin wykrywa anomalie w usuwaniu spalin – blokada mijająca
Kod błędu E80	Rozłączona sonda zewnętrzna – blokada mijająca
Kod błędu A01	Brak zapłonu palnika (brak płomienia na palniku) – blokada niemijająca
Kod błędu A03	Za wysoka temperatura wody – przegrzew kotła – blokada niemijająca
Kod błędu A90	Problem z pamięcią EEPROM – blokada niemijająca
Kod błędu A91	Błędna prędkość wentylatora – blokada niemijająca Czujnik Halla kontrolujący prędkość wentylatora jest uszkodzony lub nie jest podłączony – blokada niemijająca Wentylator pomimo podania zasilania nie wystartuje – blokada niemijająca
Kod błędu A94, A97, A98, A99	Blokada związana z błędną logiką działania – blokada niemijająca
Kod błędu A34	Prędkość wentylatora spadnie o 100 obr/min poniżej prędkości zamknięcia styków – blokada niemijająca presostatu
Kod błędu A35	Po trzech nieudanych próbach autoadaptacji wentylatora (Jeśli warunki do zamknięcia presostatu zmieniły się, np. zmiana systemu kominowego) – blokada niemijająca

W tabeli 3.5. wymieniono inne komunikaty, które mogą się pojawić na wyświetlaczu. Oznaczają one normalną pracę kotła. Dane te również zostały podane dla kotła firmy Ariston model Microgenus Plus i w zależności od modelu i producenta mogą się one różnić. W tabeli podano również, w jakim stanie znajduje się kocioł przy konkretnym komunikacie.

Tabela 3.5. Inne komunikaty, które nie oznaczają usterki lub awarii, ale normalne działanie kotła Ariston Microgenus Plus [mic14].

Komunikat	Stan kotła
Brak komunikatu	Kocioł wyłączony
On	Kocioł włączony (stand by)
LED CWU zapalona, wyświetlenie temperatury	Kocioł w trybie CWU
LED CO zapalona, wyświetlenie temperatury	Kocioł w trybie CO
SC	Kocioł w trybie analizy spalin
PUP	Autoadaptacja wentylatora
Kod błędu E99	RESET naciśnięty > 5 razy w ciągu 15 minut – blokada mijająca

Dla wymienionych stanów diagnostycznych został określony stopień rozmycia. Stopień rozmycia występowania stanu (tabela 3.6.) wyznaczony został z wykorzystaniem wiedzy ekspertowej, określającej stopień ważności tych danych za pomocą ustalenia stopnia odczucia częstości występowania poszczególnych stanów w uporządkowanej dziesięcioliterowej skali ocen z przedziału [0,1].

Tabela 3.6. Stany występujące podczas usterek i awarii oraz odpowiadające im stopnie rozmycia. Opracowanie własne.

Stany występujące podczas usterek i awarii	Stopień rozmycia występowania stanu
(F1) Brak zasilania (problem z bezpiecznikami, zwarcie)	0,4
(F2) Zadziałanie zabezpieczenia kotła	0,8
(F3) Uszkodzona elektronika (moduł elektroniczny, części elektroniczne)	0,4
(F4) Uszkodzony licznik gazowy	0,4
(F5) Uszkodzony reduktor gazowy	0,4
(F6) Wadliwie wykonany komin	0,8
(F7) Zapowietrzony grzejniki	0,4
(F8) Brak dopływu gazu (lub niskie ciśnienie gazu) – reduktor gazowy lub zawór na gazie dużej mocy	0,4
(F9) Brak wody w instalacji lub kotle (lub niskie ciśnienie)	0,6
(F10) Uszkodzona pompa cyrkulacyjna	0,4
(F11) Zatkany, zabrudzony filtr wody CO	0,8
(F12) Zatkany, zabrudzony filtr wody CWU	0,8
(F13) Zabrudzona lub uszkodzona nagrzewnica	0,4
(F14) Zapowietrzony kocioł	0,4
(F15) Uszkodzony zespół gazowy	0,4
(F16) Nieszczelność na kotle	0,4
(F17) Uszkodzona elektroda zapłonowa	0,8
(F18) Uszkodzona elektroda jonizacyjna	0,8
(F19) Uszkodzony lub zanieczyszczony zespół wodny	0,6
(F20) Przegrzew kotła	0,4
(F21) Uszkodzony czujnik temperatury wody grzewczej	0,6
(F22) Zatkane filtry, dysze lub elementy palnika	0,4
(F23) Zanieczyszczony wtórny wymiennik ciepła	0,6

W tabeli 3.7. przedstawiono rozmytą tablicę diagnostyczną (rozdz. 2.2.10.), która łączy ze sobą wcześniej wymienione dane. Tabela przedstawia macierz diagnostyczną reprezentującą rozmyte dane diagnostyczne o diagnozowanych kotłach gazowych dla rozłącznych symptomów. Również ta tabela została wykonana na podstawie wiedzy ekspertów z tej dziedziny. Poniższe dane są zapisane w systemie za pomocą sieci semantycznej przy wykorzystaniu języka ontologii OWL.

Stopnie rozmycia wyznaczone są przez stopnie ważności danych stanów diagnostycznych (patrz rozdz. 3.1.3.).

Tabela 3.7. Powiązanie symptomów ze stanami diagnostycznymi oraz stopień rozmycia tych stanów. Opracowanie własne.

Symptom	Stan diagnostyczny – usterka, wada, uszkodzenie	Stopień rozmycia
(s11) Kocioł nie włącza się	(F1) Brak zasilania (problem z bezpiecznikami, zwarcie)	0,4
	(F2) Zadziałanie zabezpieczenia kotła	0,8
	(F3) Uszkodzona elektronika (moduł elektroniczny, części elektroniczne)	0,4
	(F4) Uszkodzony licznik gazowy	0,4
	(F5) Uszkodzony reduktor gazowy	0,4
	(F6) Wadliwie wykonany komin	0,8
	(F7) Zapowietrzony grzejniki	0,4
	(F8) Brak dopływu gazu (lub niskie ciśnienie gazu) – reduktor gazowy lub zawór na gazie dużej mocy	0,4
	(F9) Brak wody w instalacji lub kotle (lub niskie ciśnienie)	0,6
(s12) Szумы, trzaski, dziwne dźwięki dobiegają z kotła	(F10) Uszkodzona pompa cyrkulacyjna	0,4
	(F11) Zatkany, zabrudzony filtr wody CO	0,8
	(F12) Zatkany, zabrudzony filtr wody CWU	0,8
	(F13) Zabrudzona lub uszkodzona nagrzewnica	0,4
	(F7) Zapowietrzony grzejniki	0,4
(s13) Kocioł sam się wyłącza	(F6) Wadliwie wykonany komin	0,8
	(F7) Zapowietrzony grzejniki	0,4
(s14) Kocioł działa w trybie CO, ale nie działa w trybie CWU	(F14) Zapowietrzony kocioł	1
(s15) Czuć zapach gazu	(F13) Zabrudzona lub uszkodzona nagrzewnica	0,4
	(F15) Uszkodzony zespół gazowy	0,4
	(F16) Nieszczelność na kotle	0,4
(s16) Gorąca obudowa	(F13) Zabrudzona lub uszkodzona nagrzewnica	0,4
	(F5) Uszkodzony reduktor gazowy	0,4
	(F16) Nieszczelność na kotle	0,4
(s21) Słychać lub widać iskrę, ale palnik się nie zapala	(F3) Uszkodzona elektronika (moduł elektroniczny, części elektroniczne)	0,4
	(F15) Uszkodzony zespół gazowy	0,4
	(F17) Uszkodzona elektroda zapłonowa	0,8
(s22) Zapala się palnik, po kilku sekundach gaśnie	(F19) Uszkodzony lub zanieczyszczony zespół wodny	0,6
	(F15) Uszkodzony zespół gazowy	0,4
	(F18) Uszkodzona elektroda jonizacyjna	0,8
(s23) Palnik gaśnie po dłuższym czasie	(F20) Przegrzew kotła	0,4
	(F21) Uszkodzony czujnik temperatury wody grzewczej	0,6
	(F2) Zadziałanie zabezpieczenia kotła	0,8

	(F19) Uszkodzony lub zanieczyszczony zespół wodny	0,6
	(F10) Uszkodzona pompa cyrkulacyjna	0,4
	(F15) Uszkodzony zespół gazowy	0,4
(s24) Palnik działa, ale jest mały płomień	(F15) Uszkodzony zespół gazowy	0,4
	(F22) Zatkane filtry, dysze lub elementy palnika	0,4
(s31) Słaba wydajność kotła (CO)	(F21) Uszkodzony czujnik temperatury wody grzewczej	0,6
	(F11) Zatkany, zabrudzony filtr wody CO	0,8
	(F13) Zabrudzona lub uszkodzona nagrzewnica	0,4
(s32) Brak ciepłej wody lub zbyt niska temperatura CWU	(F12) Zatkany, zabrudzony filtr wody CWU	0,8
	(F13) Zabrudzona lub uszkodzona nagrzewnica	0,4
	(F23) Zanieczyszczony wtórny wymiennik ciepła	0,6

3.4.4. Diagnostyczna tablica decyzyjna

W podrozdziale 2.2.10. (głównie w tabeli 2.4.) zaprezentowano zastosowanie schematów tabel decyzyjnych w diagnostyce, odpowiadających tabelom diagnostycznym. Pozwala to dla podanej tablicy obserwacji (tabela 3.3.), odpowiadającej macierzy diagnostycznej przedstawionej w poprzednim podrozdziale, na stworzenie diagnostycznej tablicy decyzyjnej przedstawionej w tabeli 3.8.

Tabela 3.8. Diagnostyczna tablica decyzyjna. Opracowanie własne.

Reguła	Symptom			Stan
	S1	S2	S3	
R1	s11	–	–	{F1, F2, F3, F4, F5, F6, F7, F8, F9}
R2	s12	–	–	{F7, F10, F11, F12, F13}
R3	s13	–	–	{F6, F7}
R4	s14	–	–	{F14}
R5	s15	–	–	{F13, F15, F16}
R6	s16	–	–	{F5, F13, F16}
R7	–	s21	–	{F3, F15, F17}
R8	–	s22	–	{F15, F18, F19}
R9	–	s23	–	{F2, F10, F15, F19, F20, F21}
R10	–	s24	–	{F15, F22}

R11	–	–	s31	{F11, F13, F21}
R12	–	–	s32	{F12, F13, F23}

Reguły zawarte w tabeli 3.8. zapisujemy następująco:

R4: JEŻELI S1 = s14 i S2 = „–” i S3 = „–” TO Stan = F14

Niektóre reguły można zapisać w skróconej postaci. Na przykład:

R4: JEŻELI S1 = s14 TO Stan = F14

Reguły o tych samych warunkach można zapisać, łącząc alternatywą wnioski tych reguł (por. rozdz. 2.2.10.). Na przykład reguła R12:

R12: JEŻELI S3 = s32 TO Stan = F12 lub Stan = F13 lub Stan = F23

Nieformalnie można to wyrazić w sposób lingwistyczny następująco:

JEŚLI Brak ciepłej wody lub zbyt niska temperatura CWU, **TO** Zatkany, zabrudzony filtr wody CWU **lub** Zabrudzona, uszkodzona nagrzewnica **lub** Zanieczyszczony wtórny wymiennik ciepła.

3.4.5. Pozyskiwanie eksperckiej wiedzy diagnostycznej

W procesie diagnozy dla obserwowanego symptomu obiektu diagnozowanego określone są stany, w jakich może znajdować się badany obiekt diagnozowany, czyli kocioł gazowy. Informacja o tym związku symptomu ze stanami diagnostycznymi może być mniej lub bardziej ważna dla ekspertów zajmujących się diagnostyką obiektów technicznych, tu kotłów gazowych. *Stopień ważności* wyszukiwania takiej informacji może być wyrażony przez ekspertów na przykład jako stopień odczucia częstości wystąpienia diagnozowanego stanu (rozdz. 3.1.3.). Może być wyrażony przeciętnym stopniem odczucia częstości wystąpienia diagnozowanego stanu w skali dziesięciostopniowej: 0; 0,1; 0,2; ...; 0,9. Uzyskane wartości stopnia ważności mogą być wynikiem danych uzyskanych podczas wywiadu z ekspertami.

Poniżej podane są wyniki takiego wywiadu. Ze względu na to, że dla rozważanych dwóch rodzajów kotłów gazowych ich budowa różni się w zależności od funkcji i użytych technologii, a co za tym idzie mogą występować inne dla tych rodzajów symptomy i stany, w celu określenia rodzaju kotła zostały ustalone dwa pytania:

- 1) *Czy kocioł jest jedno- czy dwufunkcyjny?* – Określa budowę zespołu wodnego kotła. W kotle jednofunkcyjnym nie występują elementy związane z ogrzewaniem wody użytkowej CWU.
- 2) *Czy ma otwartą czy zamkniętą komorę spalania?* – Określa technologię odprowadzania spalin z kotła. Kocioł z otwartą komorą spalania odprowadza spaliny za pomocą ciągu grawitacyjnego, natomiast kocioł z zamkniętą komorą spalania wykorzystuje w tym celu wentylator.

Dla każdego symptomu zostały przygotowane dodatkowe pytania, które mają wpływ na poziom ważności stanów. Dzięki odpowiedzi na te pytania zmienia się również stopień

rozmycia stanów. Następnie z otrzymanych stanów i ich stopni rozmycia obliczane jest maksimum, czyli najwyższy stopień rozmycia, określający zakres ufności rozmycia. W wyniku tej operacji otrzymujemy jeden bądź kilka stanów, uważanych za stany, które wystąpiły w badanym kotle.

Poniżej przedstawiono schemat wnioskowania dla poszczególnych symptomów oraz listę pytań, które się pojawiają dla wybranego symptomu.

Symptom ogólny: Kocioł się nie włącza (s11)

Możliwe stany kotła o stopniu rozmycia 0,8:

(F2) Zdziałanie zabezpieczenia kotła

(F6) Wadliwie wykonany komin

Możliwe stany kotła o stopniu rozmycia 0,6:

(F9) Brak wody w instalacji lub kotle (lub niskie ciśnienie)

Możliwe stany kotła o stopniu rozmycia 0,4:

(F1) Brak zasilania (problem z bezpiecznikami, zwarcie)

(F3) Uszkodzona elektronika (moduł elektroniczny, części elektroniczne)

(F4) Uszkodzony licznik gazowy

(F5) Uszkodzony reduktor gazowy

(F7) Zapowietrzone grzejniki

(F8) Brak dopływu gazu (lub niskie ciśnienie gazu)

Lista pytań dodatkowych zadawanych użytkownikowi oraz zmiana stopnia ważności w zależności od odpowiedzi na pytanie:

- 1) Czy sprawdzono dopływ gazu do kotła (czy inne urządzenia na gaz działają)?
Tak: (F8) +0,3; Nie: (F8) -0,3; Nie wiem: bez zmian
- 2) Czy sprawdzono, czy grzejniki nie są zapowietrzone albo czy wszystkie lub większość grzejników w instalacji jest zakręconych?
Tak: (F7) -0,3; Nie: (F7) +0,3; Nie wiem: bez zmian
- 3) Czy pojawiają się problemy z ciśnieniem wody (np. jest za niskie ciśnienie, skoki ciśnienia) lub widać skroploną parę wodną na elementach instalacji?
Tak: (F9) +0,2; Nie: (F9) -0,2; Nie wiem: bez zmian
- 4) Czy elektronika kotła działa w sposób niepokojący (np. „dziwne” wartości na wyświetlaczu – nieoznaczające kody błędów; działanie kotła chwilowo lepsze po resecie), ostatnio wystąpiło przepięcie w sieci elektrycznej lub kocioł był odłączany od prądu?
Tak: (F3) +0,3; Nie: (F3) -0,3; Nie wiem: bez zmian
- 5) Czy wyświetlał się kod błędu na kotle?
Tak: (F2) +0,2; Nie: (F2) -0,2; Nie wiem: bez zmian
- 6) Czy w pobliżu komina obudowa kotła robi się nadmiernie gorąca, pojawia się kod błędu na wyświetlaczu oznaczający problemy z kominem lub występowały problemy z kominem i odprowadzaniem spalin?

- Tak: (F6) +0,3; Nie: (F6) -0,3; Nie wiem: bez zmian
- 7) Czy sprawdzono gniazdko elektryczne, do którego jest podłączony kocioł, oraz bezpieczniki?
Tak: (F1) -0,3; Nie: (F1) +0,3; Nie wiem: bez zmian
- 8) Czy są problemy z wydajnością kotła (temperatura wody jest za wysoka lub za niska)?
Tak: (F5) +0,2; Nie: (F5) -0,2; Nie wiem: bez zmian
- 9) Czy w ciągu ostatniego roku zostały przeprowadzone konserwacja i czyszczenie kotła (czy były sprawdzane filtry)?
Tak: (F4), (F5), (F6) -0,3; Nie: (F4), (F5), (F6) +0,3; Nie wiem: bez zmian
- 10) Czy kocioł jest podłączony do miejskiej sieci gazowej?
Tak: (F8) -0,3; Nie: (F8) +0,3; Nie wiem: bez zmian
- 11) Czy występowały problemy z ciśnieniem gazu (np. skoki ciśnienia, za niskie ciśnienie)?
Tak: (F4), (F5) +0,2; Nie: (F4), (F5) -0,2; Nie wiem: bez zmian

Symptom ogólny: Szумы, trzaski, dziwne dźwięki dobiegają z kotła (s12)

Możliwe stany kotła o stopniu rozmycia 0,8:

(F11) Zatkany, zabrudzony filtr wody CO

(F12) Zatkany, zabrudzony filtr wody CWU

Możliwe stany kotła o stopniu rozmycia 0,4:

(F10) Uszkodzona pompa cyrkulacyjna

(F13) Zabrudzona lub uszkodzona nagrzewnica

(F7) Zapowietrzone grzejniki

Lista pytań dodatkowych zadawanych użytkownikowi oraz zmiana stopnia ważności w zależności od odpowiedzi na pytanie:

- 1) Czy sprawdzono, czy grzejniki nie są zapowietrzone albo czy wszystkie lub większość grzejników w instalacji jest zakręconych?
Tak: (F7) -0,3; Nie: (F7) +0,3; Nie wiem: bez zmian
- 2) Czy pojawiają się problemy z ciśnieniem wody (np. jest za niskie ciśnienie, skoki ciśnienia) lub widać skroploną parę wodną na elementach instalacji?
Tak: (F10) +0,3; Nie: (F10) -0,3; Nie wiem: bez zmian
- 3) Czy widać zabrudzenia lub sadzę na nagrzewnicy?
Tak: (F13) +0,3; Nie: (F13) -0,3; Nie wiem: bez zmian
- 4) Czy są problemy z wydajnością kotła (temperatura wody jest za wysoka lub za niska)?
Tak: (F10) +0,3; Nie: (F10) -0,3; Nie wiem: bez zmian
- 5) Czy w ciągu ostatniego roku zostały przeprowadzone konserwacja i czyszczenie kotła (czy były sprawdzane filtry)?
Tak: (F11), (F12) -0,3, (F13) -0,2; Nie: (F11), (F12) +0,3, (F13) +0,2; Nie wiem: bez zmian

Symptom ogólny: Kocioł sam się wyłącza (s13)

Możliwe stany kotła o stopniu rozmycia 0,8:

(F6) Wadliwie wykonany komin

Możliwe stany kotła o stopniu rozmycia 0,4:

(F7) Zapowietrzony grzejniki

Lista pytań dodatkowych zadawanych użytkownikowi oraz zmiana stopnia ważności w zależności od odpowiedzi na pytanie:

1) Czy sprawdzono, czy grzejniki nie są zapowietrzony albo czy wszystkie lub większość grzejników w instalacji jest zakręconych?

Tak: (F7) -0,3; Nie: (F7) +0,3; Nie wiem: bez zmian

2) Czy w pobliżu komina obudowa kotła robi się nadmiernie gorąca, pojawia się kod błędu na wyświetlaczu oznaczający problemy z kominem lub występowały problemy z kominem i odprowadzaniem spalin?

Tak: (F6) +0,3; Nie: (F6) -0,3; Nie wiem: bez zmian

3) Czy w ciągu ostatniego roku zostały przeprowadzone konserwacja i czyszczenie kotła (czy były sprawdzane filtry)?

Tak: (F6) -0,3; Nie: (F6) +0,3; Nie wiem: bez zmian

Symptom ogólny: Kocioł działa w trybie CO, ale nie działa w trybie CWU (s14)

Symptom może występować tylko w przypadku kotła dwufunkcyjnego.

Możliwe stany kotła o stopniu rozmycia 0,4:

(F14) Zapowietrzony kocioł

Bez dodatkowych pytań

Symptom ogólny: Czuć zapach gazu (s15)

Możliwe stany kotła o stopniu rozmycia 0,4:

(F13) Zabrudzona lub uszkodzona nagrzewnica

(F15) Uszkodzony zespół gazowy

(F16) Nieszczelność na kotle

Lista pytań dodatkowych zadawanych użytkownikowi oraz zmiana stopnia ważności w zależności od odpowiedzi na pytanie:

1) Czy widać zabrudzenia lub sadzę na nagrzewnicy?

Tak: (F13) +0,3; Nie: (F13) -0,3; Nie wiem: bez zmian

2) Czy występowały problemy z ciśnieniem gazu (np. skoki ciśnienia, za niskie ciśnienie)?

Tak: (F15) +0,3; Nie: (F15) -0,3; Nie wiem: bez zmian

3) Czy kocioł ma więcej niż 5 lat?

Tak: (F16) +0,3; Nie: (F16) -0,3; Nie wiem: bez zmian

4) Czy są problemy z wydajnością kotła (temperatura wody jest za wysoka lub za niska)?

Tak: (F15) +0,3; Nie: (F15) -0,3; Nie wiem: bez zmian

5) Czy w ciągu ostatniego roku zostały przeprowadzone konserwacja i czyszczenie kotła (czy były sprawdzane filtry)?

Tak: (F16) +0,2, (F13) -0,2; Nie: (F16) -0,2, (F13), (F15) +0,2; Nie wiem: bez zmian

Symptom ogólny: Gorąca obudowa (s16)

Możliwe stany kotła o stopniu rozmycia 0,4:

(F13) Zabrudzona lub uszkodzona nagrzewnica

(F5) Uszkodzony reduktor gazowy

(F16) Nieszczelność na kotle

Lista pytań dodatkowych zadawanych użytkownikowi oraz zmiana stopnia ważności w zależności od odpowiedzi na pytanie:

1) Czy widać zabrudzenia lub sadzę na nagrzewnicy?

Tak: (F13) +0,3; Nie: (F13) -0,3; Nie wiem: bez zmian

2) Czy kocioł ma więcej niż 5 lat?

Tak: (F16) +0,3; Nie: (F16) -0,3; Nie wiem: bez zmian

3) Czy są problemy z wydajnością kotła (temperatura wody jest za wysoka lub za niska)?

Tak: (F5) +0,2; Nie: (F5) -0,2; Nie wiem: bez zmian

4) Czy w ciągu ostatniego roku zostały przeprowadzone konserwacja i czyszczenie kotła (czy były sprawdzane filtry)?

Tak: (F5) -0,3, (F16) +0,2, (F13) -0,2; Nie: (F5) +0,3, (F16) -0,2, (F13) +0,2; Nie wiem: bez zmian

5) Czy występowały problemy z ciśnieniem gazu (np. skoki ciśnienia, za niskie ciśnienie)?

Tak: (F15) +0,3, (F5) +0,2; Nie: (F15) -0,3, (F5) -0,2; Nie wiem: bez zmian

Działanie palnika: Słyszać lub widzieć iskrę, ale palnik się nie zapala (s21)

Możliwe stany kotła o stopniu rozmycia 0,8:

(F17) Uszkodzona elektroda zapłonowa

Możliwe stany kotła o stopniu rozmycia 0,4:

(F3) Uszkodzona elektronika (moduł elektroniczny, części elektroniczne)

(F15) Uszkodzony zespół gazowy

Lista pytań dodatkowych zadawanych użytkownikowi oraz zmiana stopnia ważności w zależności od odpowiedzi na pytanie:

1) Czy elektronika kotła działa w sposób niepokojący (np. dziwne wartości na wyświetlaczu – nieoznaczające kody błędów; działanie kotła chwilowo lepsze po resecie), ostatnio wystąpiło przepięcie w sieci elektrycznej lub kocioł był odłączany od prądu?

Tak: (F3) +0,3; Nie: (F3) -0,3; Nie wiem: bez zmian

2) Czy występowały problemy z ciśnieniem gazu (np. skoki ciśnienia, za niskie ciśnienie)?

Tak: (F15) +0,3; Nie: (F15) -0,3; Nie wiem: bez zmian

3) Czy jest iskra, ale nie w tym miejscu, w którym powinna być?

Tak: (F17) +0,2; Nie: (F17) -0,2; Nie wiem: bez zmian

4) Czy są problemy z wydajnością kotła (temperatura wody jest za wysoka lub za niska)?

Tak: (F15) +0,3; Nie: (F15) -0,3; Nie wiem: bez zmian

- 5) Czy w ciągu ostatniego roku zostały przeprowadzone konserwacja i czyszczenie kotła (czy były sprawdzane filtry)?

Tak: (F15) -0,2; Nie: (F15) +0,2; Nie wiem: bez zmian

Działanie palnika: Zapala się palnik, po kilku sekundach gaśnie (s22)

Możliwe stany kotła o stopniu rozmycia 0,8:

(F18) Uszkodzona elektroda jonizacyjna

Możliwe stany kotła o stopniu rozmycia 0,6:

(F19) Uszkodzony lub zanieczyszczony zespół wodny

Możliwe stany kotła o stopniu rozmycia 0,4:

(F15) Uszkodzony zespół gazowy

Lista pytań dodatkowych zadawanych użytkownikowi oraz zmiana stopnia ważności w zależności od odpowiedzi na pytanie:

- 1) Czy pojawiają się problemy z ciśnieniem wody (np. jest za niskie ciśnienie, skoki ciśnienia) lub widać skroploną parę wodną na elementach instalacji?

Tak: (F19) +0,2; Nie: (F19) -0,2; Nie wiem: bez zmian

- 2) Czy występowały problemy z ciśnieniem gazu (np. skoki ciśnienia, za niskie ciśnienie)?

Tak: (F15) +0,3; Nie: (F15) -0,3; Nie wiem: bez zmian

- 3) Czy na palniku lub elektrodach widoczne są ślady zabrudzeń lub sadzy?

Tak: (F18) +0,2; Nie: (F18) -0,2; Nie wiem: bez zmian

- 4) Czy są problemy z wydajnością kotła (temperatura wody jest za wysoka lub za niska)?

Tak: (F15) +0,3; Nie: (F15) -0,3; Nie wiem: bez zmian

- 5) Czy w ciągu ostatniego roku zostały przeprowadzone konserwacja i czyszczenie kotła (czy były sprawdzane filtry)?

Tak: (F15), (F19) -0,2; Nie: (F15), (F19) +0,2; Nie wiem: bez zmian

Działanie palnika: Palnik gaśnie po dłuższym czasie (s23)

Możliwe stany kotła o stopniu rozmycia 0,8:

(F2) Zadziałanie zabezpieczenia kotła

Możliwe stany kotła o stopniu rozmycia 0,6:

(F21) Uszkodzony czujnik temperatury wody grzewczej

(F19) Uszkodzony lub zanieczyszczony zespół wodny

Możliwe stany kotła o stopniu rozmycia 0,4:

(F20) Przegrzew kotła

(F10) Uszkodzona pompa cyrkulacyjna

(F15) Uszkodzony zespół gazowy

Lista pytań dodatkowych zadawanych użytkownikowi oraz zmiana stopnia ważności w zależności od odpowiedzi na pytanie:

- 1) Czy pojawiają się problemy z ciśnieniem wody (np. jest za niskie ciśnienie, skoki ciśnienia) lub widać skroploną parę wodną na elementach instalacji?
Tak: (F19) +0,2, (F10) +0,3; Nie: (F19) -0,2, (F10) -0,3; Nie wiem: bez zmian
- 2) Czy wyświetlał się kod błędu na kotle?
Tak: (F2) +0,2; Nie: (F2) -0,2; Nie wiem: bez zmian
- 3) Czy obudowa jest nadmiernie gorąca przed wystąpieniem objawu?
Tak: (F20) +0,3; Nie: (F20) -0,3; Nie wiem: bez zmian
- 4) Czy występowały problemy z ciśnieniem gazu (np. skoki ciśnienia, za niskie ciśnienie)?
Tak: (F15) +0,3; Nie: (F15) -0,3; Nie wiem: bez zmian
- 5) Czy pojawia się kod błędu oznaczający usterkę czujnika temperatury lub na wyświetlaczu pokazuje się wyższa temperatura niż jest faktycznie?
Tak: (F21) +0,3; Nie: (F21) -0,3; Nie wiem: bez zmian
- 6) Czy są problemy z wydajnością kotła (temperatura wody jest za wysoka lub za niska)?
Tak: (F10), (F15) +0,3; Nie: (F10), (F15) -0,3; Nie wiem: bez zmian
- 7) Czy w ciągu ostatniego roku zostały przeprowadzone konserwacja i czyszczenie kotła (czy były sprawdzane filtry)?
Tak: (F15), (F19) -0,2; Nie: (F15), (F19) +0,2; Nie wiem: bez zmian

Działanie palnika: Palnik działa, ale jest mały płomień (s24)

Możliwe stany kotła o stopniu rozmycia 0,4:

(F15) Uszkodzony zespół gazowy

(F22) Zatkane filtry, dysze lub elementy palnika

Lista pytań dodatkowych zadawanych użytkownikowi oraz zmiana stopnia ważności w zależności od odpowiedzi na pytanie:

- 1) Czy występowały problemy z ciśnieniem gazu (np. skoki ciśnienia, za niskie ciśnienie)?
Tak: (F15) +0,3; Nie: (F15) -0,3; Nie wiem: bez zmian
- 2) Czy na palniku lub elektrodach widoczne są ślady zabrudzeń lub sadzy?
Tak: (F22) +0,3; Nie: (F22) -0,3; Nie wiem: bez zmian
- 3) Czy są problemy z wydajnością kotła (temperatura wody jest za wysoka lub za niska)?
Tak: (F15) +0,3; Nie: (F15) -0,3; Nie wiem: bez zmian
- 4) Czy w ciągu ostatniego roku zostały przeprowadzone konserwacja i czyszczenie kotła (czy były sprawdzane filtry)?
Tak: (F15) -0,2; Nie: (F15) +0,2; Nie wiem: bez zmian

Wydajność kotła: Słaba wydajność kotła CO (s31)

Możliwe stany kotła o stopniu rozmycia 0,8:

(F11) Zatkany, zabrudzony filtr wody CO

Możliwe stany kotła o stopniu rozmycia 0,6:

(F21) Uszkodzony czujnik temperatury wody grzewczej

Możliwe stany kotła o stopniu rozmycia 0,4:

(F13) Zabrudzona lub uszkodzona nagrzewnica

Lista pytań dodatkowych zadawanych użytkownikowi oraz zmiana stopnia ważności w zależności od odpowiedzi na pytanie:

- 1) Czy widać zabrudzenia lub sadzę na nagrzewnicy?
Tak: (F13) +0,3; Nie: (F13) -0,3; Nie wiem: bez zmian
- 2) Czy pojawia się kod błędu oznaczający usterkę czujnika temperatury lub na wyświetlaczu pokazuje się wyższa temperatura, niż jest faktycznie?
Tak: (F21) +0,3; Nie: (F21) -0,3; Nie wiem: bez zmian
- 3) Czy w ciągu ostatniego roku zostały przeprowadzone konserwacja i czyszczenie kotła (czy były sprawdzane filtry)?
Tak: (F11) -0,3, (F13) -0,2; Nie: (F11) +0,3, (F13) +0,2; Nie wiem: bez zmian

Wydatność kotła: Brak ciepłej wody lub zbyt niska temperatura CWU (s32)

Symptom może występować tylko w przypadku kotła dwufunkcyjnego.

Możliwe stany kotła o stopniu rozmycia 0,8:

(F12) Zatkany, zabrudzony filtr wody CWU

Możliwe stany kotła o stopniu rozmycia 0,6:

(F23) Zanieczyszczony wtórny wymiennik ciepła

Możliwe stany kotła o stopniu rozmycia 0,4:

(F13) Zabrudzona lub uszkodzona nagrzewnica

Lista pytań dodatkowych zadawanych użytkownikowi oraz zmiana stopnia ważności w zależności od odpowiedzi na pytanie:

- 1) Czy widać zabrudzenia lub sadzę na nagrzewnicy?
Tak: (F13) +0,3; Nie: (F13) -0,3; Nie wiem: bez zmian
- 2) Czy w ciągu ostatniego roku zostały przeprowadzone konserwacja i czyszczenie kotła (czy były sprawdzane filtry)?
Tak: (F12) -0,3, (F13) -0,2; Nie: (F12) +0,3, (F13) +0,2; Nie wiem: bez zmian

3.4.6. Klasyfikator konstrukcyjno-technologiczny kotła gazowego

Zgodnie z ogólnym algorytmem tezaury (rozdz. 3.1.3.), w tym i w następnym podrozdziale zostanie opisana struktura tezaury dla wiedzy diagnostycznej o przykładowych rozważanych rodzajach kotła gazowego. Najpierw określony zostanie klasyfikator konstrukcyjno-technologiczny. W klasyfikatorze zostały uwzględnione wszystkie części występujące w kotle gazowym z otwartą i zamkniętą komorą spalania, jedno- i dwufunkcyjnym. Części te zostały pogrupowane ze względu na technologie i zespoły. Poniżej znajduje się klasyfikator utworzony zgodnie z algorytmem opisanym w podrozdziale 3.1.3.

Elementy kotła O1:

- | | |
|----------------------------------|-------------|
| 1. Technologia spalania – Palnik | T_1^1 |
| 1.1. Osłona palnika | $K_1^{1.1}$ |
| 1.2. Elektroda zapłonowa | $K_2^{1.1}$ |

1.3. Elektroda jonizacyjna	K ₃ ¹
1.4. Filtr palnika	K ₄ ^{1.1}
1.5. Dysze palnika	K ₅ ^{1.1}
2. Technologia cyrkulacji wody – Pompa cyrkulacyjna	T ₂ ¹
2.1. Pompa cyrkulacyjna z odpowietrznikiem	K ₁ ^{1.2}
2.2. Odpowietrznik	K ₂ ^{1.2}
3. Technologia hydrauliki wody – Układ hydrauliczny	T ₃ ¹
3.1. Zespół wodny CO	K ₁ ^{1.3}
3.1.1. Sonda na zasilaniu CO	B ₁ ^{1.3.1}
3.1.2. Sonda na powrocie CO	B ₂ ^{1.3.1}
3.1.3. Powrót CO	B ₃ ^{1.3.1}
3.1.4. Zasilanie CO	B ₄ ^{1.3.1}
3.1.5. Zawór 3-drogowy z silnikiem	B ₅ ^{1.3.1}
3.1.6. Zawór bezpieczeństwa	B ₆ ^{1.3.1}
3.1.7. Naczynie wzbiorcze	B ₇ ^{1.3.1}
3.1.8. Termostat przegrzewu (czujnik temperatury wody grzewczej)	B ₈ ^{1.3.1}
3.1.9. Czujnik ciśnienia minimalnego	B ₉ ^{1.3.1}
3.1.10. Zawór napełniania	B ₁₀ ^{1.3.1}
3.1.11. Wymiennik pierwotny	B ₁₁ ^{1.3.1}
3.1.12. Filtr wody CO	B ₁₂ ^{1.3.1}
3.2. Zespół wodny CWU (tylko w kotłach dwufunkcyjnych)	K ₂ ^{1.3}
3.2.1. Wymiennik wtórny	B ₁ ^{1.3.2}
3.2.2. Sonda CWU	B ₂ ^{1.3.2}
3.2.3. Filtr wody na wejściu CWU	B ₃ ^{1.3.2}
3.2.4. Wyjście CWU	B ₄ ^{1.3.2}
3.2.5. Wejście CWU	B ₅ ^{1.3.2}
3.2.6. Czujnik przepływu CWU	B ₆ ^{1.3.2}
4. Technologia elektroniczna – Zespół elektryczny	T ₄ ¹
4.1. Moduł elektryczny	K ₁ ^{1.4}
4.2. Zasilanie	K ₂ ^{1.4}
4.3. Wyświetlacz	K ₃ ^{1.4}
4.4. Elementy elektryczne	K ₄ ^{1.4}
5. Technologia przepływu gazu – Zespół gazowy	T ₅ ¹
5.1. Zawór gazowy	K ₁ ^{1.5}
5.2. Generator zapłonu	K ₂ ^{1.5}
5.3. Wejście gazu	K ₃ ^{1.5}
5.4. Licznik gazowy	K ₄ ^{1.5}
5.5. Reduktor gazowy	K ₅ ^{1.5}
5.6. Sonda zewnętrzna (mierzy temperaturę na zewnątrz kotła)	K ₆ ^{1.5}
6. Technologia pozyskiwania ciepła ze spalania – Komora spalania	T ₆ ¹

6.1. Komora spalania	K ₁ ^{1.6}
6.2. Izolacja komory spalania	K ₂ ^{1.6}
6.3. Nagrzewnica (wymiennik pierwotny)	K ₃ ^{1.6}
7. Technologia regulacji procesu spalania, odprowadzania spalin	T ₇ ¹
7.1. Komin	K ₁ ^{1.7}
7.2. Otwarta komora spalania	K ₂ ^{1.7}
7.2.1. Kryza komina	B ₁ ^{1.7.2}
7.2.2. Przerwywacz ciągu	B ₂ ^{1.7.2}
7.2.3. Sonda spalin (czujnik temperatury spalin)	B ₃ ^{1.7.2}
7.3. Zamknięta komora spalania	K ₃ ^{1.7}
7.3.1. Złącze spalin	B ₁ ^{1.7.3}
7.3.2. Podłączenie powietrza	B ₂ ^{1.7.3}
7.3.3. Wentylator modulowany	B ₃ ^{1.7.3}
7.3.4. Czujnik Halla wentylatora	B ₄ ^{1.7.3}
7.3.5. Kolektor spalin	B ₅ ^{1.7.3}
7.3.6. Presostat spalin	B ₆ ^{1.7.3}
7.3.7. Punkt analizy spalin	B ₇ ^{1.7.3}

Na wystąpienie określonego stanu kotła mogą wpływać nie tylko usterki i awarie części kotła, ale również czynniki zewnętrzne. Poniżej wymieniono, jakie czynniki zewnętrzne zostały rozpatrzone w systemie.

Czynniki zewnętrzne:

- (Z1) Niskie ciśnienie wody, niewystarczające ciśnienie wody w sieci wodociągowej.
- (Z2) Awaria jednego z reduktorów gazu na zasilaniu doprowadzonego do kotła.
- (Z3) Uszkodzony zawór gazowy dużej mocy.
- (Z4) Powietrze w instalacji CO z powodu: minimalnych nieszczelności, przedostawania się powietrza z wody, zbyt szybkiego napełniania kotła.
- (Z5) Zatkanie, zabrudzenie elementów na dopływie wody do kotła (np. filtry, sitko).

3.4.7. Powiązanie klasyfikatora ze stanami określającymi usterki i awarie

Dalszą część określenia tezauryusa stanowi powiązanie klasyfikatora ze stanami: usterkami i wadami. W tabeli 3.9. przedstawiono stany diagnostyczne kotła gazowego rozpatrywane w systemie wraz z odpowiadającymi im elementami klasyfikatora konstrukcyjno-technologicznego i czynnikami zewnętrznymi. Każdy ze stanów odpowiada konkretnym usterkom lub awariom części kotła. W zależności od rodzaju kotła, który powinien zostać ustalony, mogą ulec usterece różne elementy. W tabeli przedstawiono elementy kotła, które mogły ulec usterece dla kotłów z otwartą i zamkniętą komorą spalania, oraz elementy dodatkowe, występujące tylko w kotłach dwufunkcyjnych.

Tabela 3.9. Stany kotła i odpowiadające im elementy klasyfikatora konstrukcyjno-technologicznego. Opracowanie własne.

Stan diagnostyczny kotła	Symbol stanu	Element klasyfikatora konstrukcyjno-technologicznego lub czynniki zewnętrzne
Brak zasilania (problem z bezpiecznikami, zwarcie)	F1	Zasilanie K ₂ ^{1.4}
Uszkodzona elektronika (moduł elektroniczny, części elektroniczne)	F3	Moduł elektryczny K ₁ ^{1.4} Elementy elektryczne K ₄ ^{1.4}
Uszkodzony licznik gazowy	F4	Licznik gazowy K ₄ ^{1.5}
Uszkodzony reduktor gazowy	F5	Reduktor gazowy K ₅ ^{1.5}
Wadliwie wykonany komin	F6	Komin K ₁ ^{1.7} Otwarta komora spalania: Kryza komina B ₁ ^{1.7.2} Przerywacz ciągu B ₂ ^{1.7.2} Zamknięta komora spalania: Złącze spalin B ₁ ^{1.7.3} Podłączenie powietrza B ₂ ^{1.7.3} Kolektor spalin B ₅ ^{1.7.3}
Uszkodzona pompa cyrkulacyjna	F10	Pompa cyrkulacyjna K ₁ ^{1.2} Odpowietrznik K ₂ ^{1.2}
Zatkany, zabrudzony filtr wody CO	F11	Filtr wody CO B ₁₂ ^{1.3.1}
Zatkany, zabrudzony filtr wody CWU	F12	Kocioł dwufunkcyjny: Filtr wody na wejściu CWU B ₃ ^{1.3.2}
Zabrudzona lub uszkodzona nagrzewnica	F13	Nagrzewnica K ₃ ^{1.6}
Uszkodzony zespół gazowy	F15	Zawór gazowy K ₁ ^{1.5} Generator zapłonu K ₂ ^{1.5} Licznik gazowy K ₄ ^{1.5} Reduktor gazowy K ₅ ^{1.5}
Uszkodzona elektroda zapłonowa	F17	Elektroda zapłonowa K ₂ ^{1.1}
Uszkodzona elektroda jonizacyjna	F18	Elektroda jonizacyjna K ₃ ¹
Uszkodzony lub zanieczyszczony zespół wodny	F19	Sonda na zasilaniu CO B ₁ ^{1.3.1} Sonda na powrocie CO B ₂ ^{1.3.1} Zawór 3-drogowy B ₅ ^{1.3.1} Zawór bezpieczeństwa B ₆ ^{1.3.1} Naczynie wzbiorcze B ₇ ^{1.3.1} Termostat przegrzewu B ₈ ^{1.3.1} Czujnik ciśnienia minimalnego B ₉ ^{1.3.1} Zawór napełniania B ₁₀ ^{1.3.1} Wymiennik pierwotny B ₁₁ ^{1.3.1} Filtr wody CO B ₁₂ ^{1.3.1} Kocioł dwufunkcyjny: Wymiennik wtórny B ₁ ^{1.3.2} Sonda CWU B ₂ ^{1.3.2} Filtr wody na wejściu CWU B ₃ ^{1.3.2} Czujnik przepływu CWU B ₆ ^{1.3.2}

Uszkodzony czujnik temperatury wody grzewczej	F21	Termostat przegrzewu B ₈ ^{1.3.1}
Zatkane filtry, dysze lub elementy palnika	F22	Filtr palnika K ₄ ^{1.1} Dysze palnika K ₅ ^{1.1}
Zanieczyszczony wtórny wymiennik ciepła	F23	Kocioł dwufunkcyjny: Wymiennik wtórny B ₁ ^{1.3.2}
Zadziałanie zabezpieczenia kotła	F2	Zawór gazowy K ₁ ^{1.5} Moduł elektryczny K ₁ ^{1.4} Elektroda zapłonowa K ₂ ^{1.1} Elektroda jonizacyjna K ₃ ¹ Filtr palnika K ₄ ^{1.1} Dysze palnika K ₅ ^{1.1} Zamknięta komora spalania: Presostat spalin B ₆ ^{1.7.3} Wentylator modulowany B ₃ ^{1.7.3} lub czynniki zewnętrzne: (Z1) Niskie ciśnienie wody
Zapowietrzony kocioł	F14	Pompa cyrkulacyjna z odpowietrznikiem K ₁ ^{1.2} Odpowietrznik K ₂ ^{1.2} lub czynniki zewnętrzne: (Z4) Powietrze w instalacji CO
Nieszczelność na kotle	F16	Ośłona palnika K ₁ ^{1.1} Izolacja komory spalania K ₂ ^{1.6} Otwarta komora spalania: Kryza komina B ₁ ^{1.7.2} Przerywacz ciągu B ₂ ^{1.7.2} Sonda spalin B ₃ ^{1.7.2} Zamknięta komora spalania: Złącze spalin B ₁ ^{1.7.3} Podłączenie powietrza B ₂ ^{1.7.3} Kolektor spalin B ₅ ^{1.7.3} Presostat spalin B ₆ ^{1.7.3} Punkt analizy spalin B ₇ ^{1.7.3}
Przegrzew kotła	F20	Termostat przegrzewu B ₈ ^{1.3.1} Otwarta komora spalania: Sonda spalin B ₃ ^{1.7.2} Zamknięta komora spalania: Punkt analizy spalin B ₇ ^{1.7.3} Wentylator modulowany B ₃ ^{1.7.3} Czujnik Halla wentylatora B ₄ ^{1.7.3}
Zapowietrzzone grzejniki	F7	Zawór napełniania B ₁₀ ^{1.3.1} lub czynniki zewnętrzne: (Z4) Powietrze w instalacji CO
Brak dopływu gazu (lub niskie ciśnienie gazu) – reduktor gazowy lub zawór na gazie dużej mocy	F8	Zawór gazowy K ₁ ^{1.5} lub czynniki zewnętrzne:

		(Z2) Awaria jednego z reduktorów na zasilaniu gazu (Z3) Uszkodzony zawór gazowy dużej mocy
Brak wody w instalacji lub kotle (lub niskie ciśnienie)	F9	Sonda na zasilaniu CO B ₁ ^{1.3.1} Sonda na powrocie CO B ₂ ^{1.3.1} Zawór 3-drogowy z silnikiem B ₅ ^{1.3.1} Naczynie wzbiorcze B ₇ ^{1.3.1} Czujnik ciśnienia minimalnego B ₉ ^{1.3.1} Zawór napełniania B ₁₀ ^{1.3.1} Filtr wody CO B ₁₂ ^{1.3.1} Kocioł dwufunkcyjny: Sonda CWU B ₂ ^{1.3.2} Filtr wody na wejściu CWU B ₃ ^{1.3.2} Czujnik przepływu CWU B ₆ ^{1.3.2} lub czynniki zewnętrzne: (Z1) Niskie ciśnienie wody

W momencie, kiedy zostanie ustalony stan (lub możliwe stany), w którym znajduje się badany kocioł, oraz to, jaki rodzaj kotła jest rozpatrywany, na podstawie tabeli 3.9. można ustalić, które elementy mogły ulec awarii i co należy sprawdzić. System poda również możliwe czynniki zewnętrzne, które mogły spowodować zaistniałą sytuację. Oczywiście, podane są najbardziej prawdopodobne części, które mogły ulec awarii.

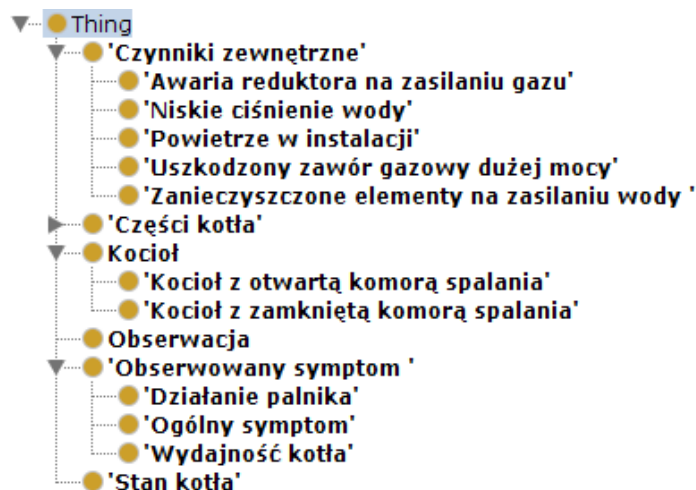
Należy również podkreślić, że większość tych usterek i awarii wymaga do dokonania naprawy specjalisty zajmującego się naprawą i konserwacją kotłów gazowych, z odpowiednimi uprawnieniami. Natomiast niektóre usterki mogą zostać naprawione przez zwykłego użytkownika, na przykład dla stanu: „Zapowietrzony kocioł”, gdy występuje czynnik zewnętrzny „Powietrze w instalacji CO”, wystarczy odpowietrzyć kocioł i grzejniki w instalacji, aby kocioł działał prawidłowo.

3.4.8. Działanie aplikacji

Określenie tezaurytu diagnozowanych obiektów pozwala na dokonanie jego implementacji w języku OWL. Ten tezaurus został określony na podstawie informacji przedstawionych w podrozdziałach 3.4.1.–3.4.7. dla systemu diagnostyki kotłów gazowych. Dlatego zostanie wykorzystany do zbudowania aplikacji służącej do diagnostyki kotłów gazowych zgodnie z tablicą obserwowanych symptomów (tabela 3.7.). Implementacja tezaurytu została utworzona za pomocą programu Protégé, który umożliwia tworzenie plików OWL zgodnie ze standardem W3C. Tezaurus jest również zgodny z logiką opisową i rozmytą logiką opisową, gdyż jest napisany w standardzie OWL DL. Dzięki temu aplikacja pełnić będzie funkcje agenta wyszukującego informacje diagnostyczne na podstawie wiedzy diagnostycznej reprezentowanej w tym standardzie, co daje możliwość umieszczenia tej aplikacji na wybranej stronie sieci Web. Kod źródłowy tezaurytu napisany w języku OWL znajduje się w dodatku A niniejszej pracy.

Obecnie bardzo trudno uzyskać dostęp do plików OWL zawierających tezaursus utworzony dla wiedzy technicznej, w szczególności dla wiedzy z zakresu diagnostyki technicznej. Większość z ogólnie dostępnych sieci semantycznych zawiera wiedzę z medycyny, ekonomii, mechaniki czy wiedzy ogólnej. Z tego względu utworzony tezaursus może posłużyć również w innych projektach jako wzorcowa sieć semantyczna.

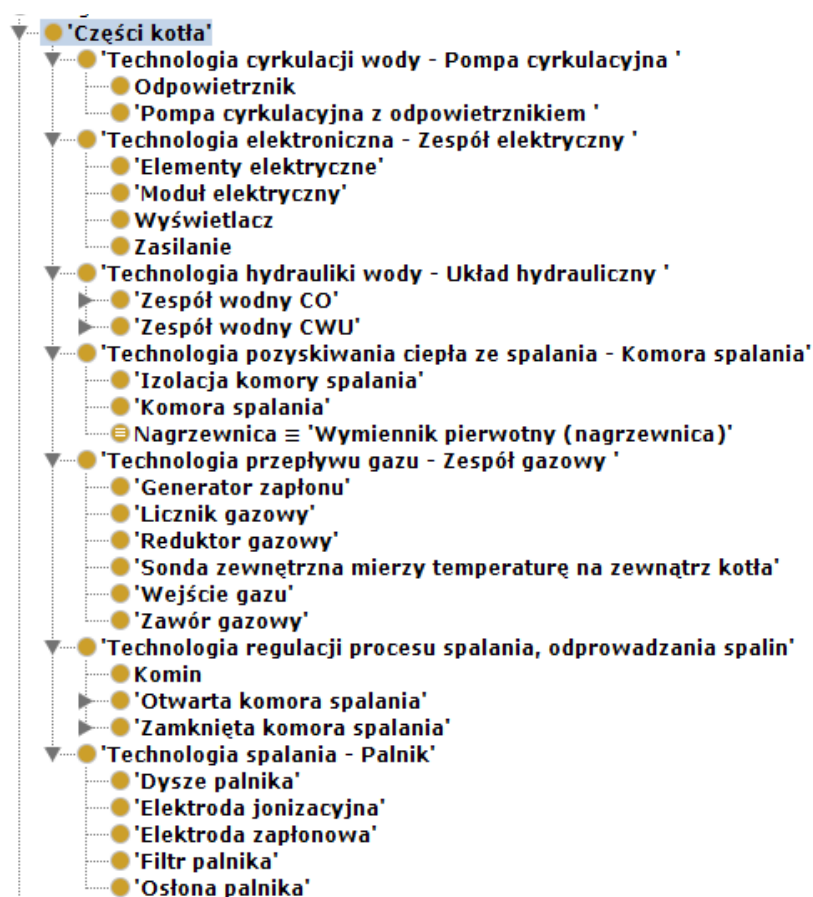
Na rysunku 3.7. przedstawiono podstawowe klasy utworzone w tezaursie aplikacji.



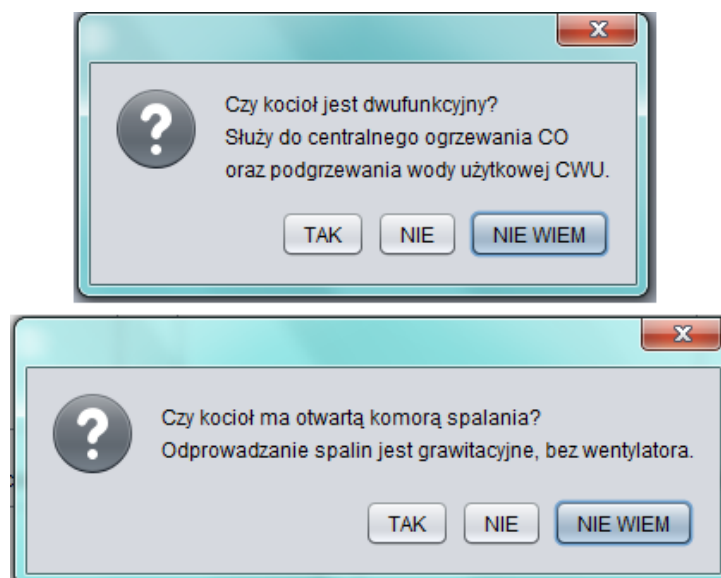
Rysunek 3.7. Klasy utworzone w tezaursie dla kotła gazowego. Opracowanie własne.

Klasa „Części kotła” zawiera klasyfikator konstrukcyjno-technologiczny dla kotła gazowego, który został opisany w podrozdziale 3.4.6. W klasyfikatorze zostały uwzględnione części, zarówno dla kotła jedno- i dwufunkcyjnego, jak i kotła z otwartą i zamkniętą komorą spalania (rysunek 3.8.). Natomiast w trakcie działania aplikacji wybierany jest typ kotła i na tej podstawie wybierane są części kotła, które występują w danym typie kotła.

Na rysunku 3.9. przedstawiono pytania, które są zadawane użytkownikowi przy uruchomieniu aplikacji. Użytkownikiem aplikacji może być zarówno ekspert zajmujący się naprawą i konserwacją wybranych typów kotłów gazowych, jak i osoba będąca użytkownikiem kotła. Na podstawie odpowiedzi na te pytania wybierane są odpowiednie części kotła z tezaursusa oraz symptomy, jakie mogą występować w obserwowanym obiekcie. Istnieje również możliwość zaznaczenia odpowiedzi „Nie wiem” na te pytania. W takim przypadku, gdy nie znamy typu kotła, rozpatrywane są wszystkie możliwe elementy kotła i symptomy, które mogą wystąpić.

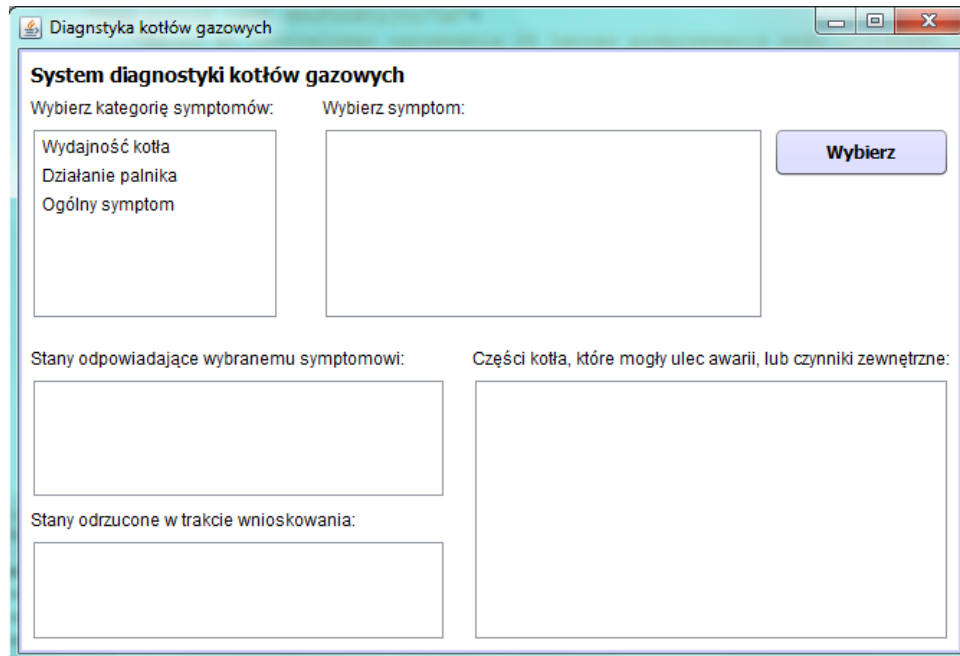


Rysunek 3.8. Klasyfikator konstrukcyjno-technologiczny dla kotła gazowego. Opracowanie własne.

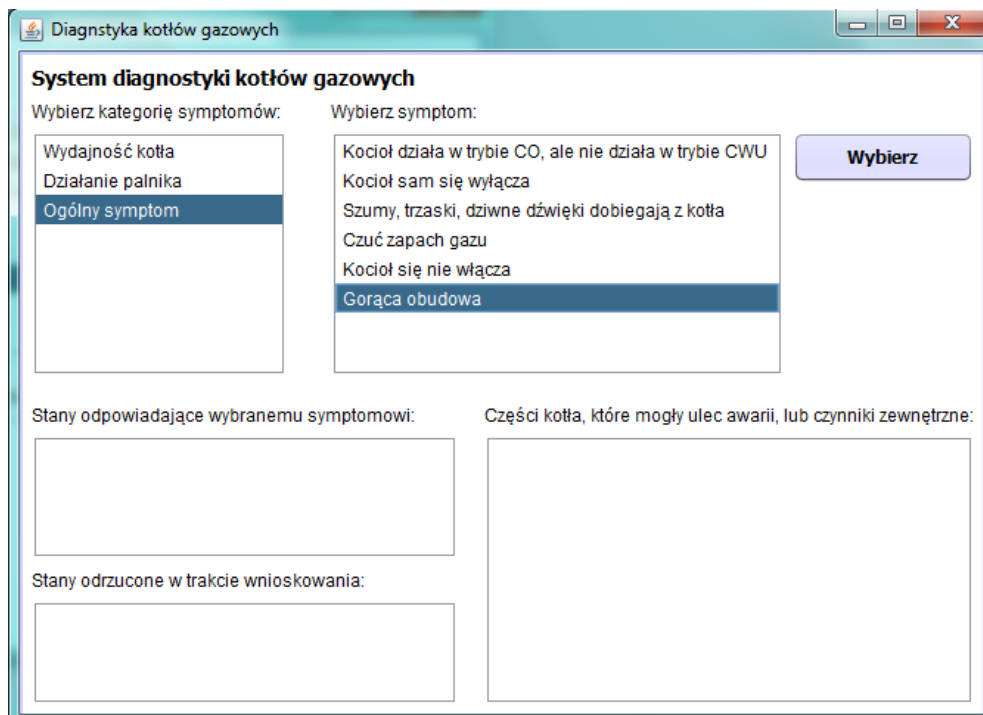


Rysunek 3.9. Pytania zadawane przy starcie aplikacji w celu ustalenia typu kotła gazowego. Opracowanie własne.

Po udzieleniu odpowiedzi na podane pytania mamy dostęp do aplikacji (rysunek 3.10.). Interfejs aplikacji jest przejrzysty i łatwy w obsłudze. Najpierw użytkownik wybiera kategorię symptomów z podanych na podstawie tezauryusa zapisanego w OWL. Na tej podstawie pokazane zostają symptomy, które mogą być obserwowane w kotle. Wszystkie symptomy wyrażone są lingwistycznie (rysunek 3.11.).

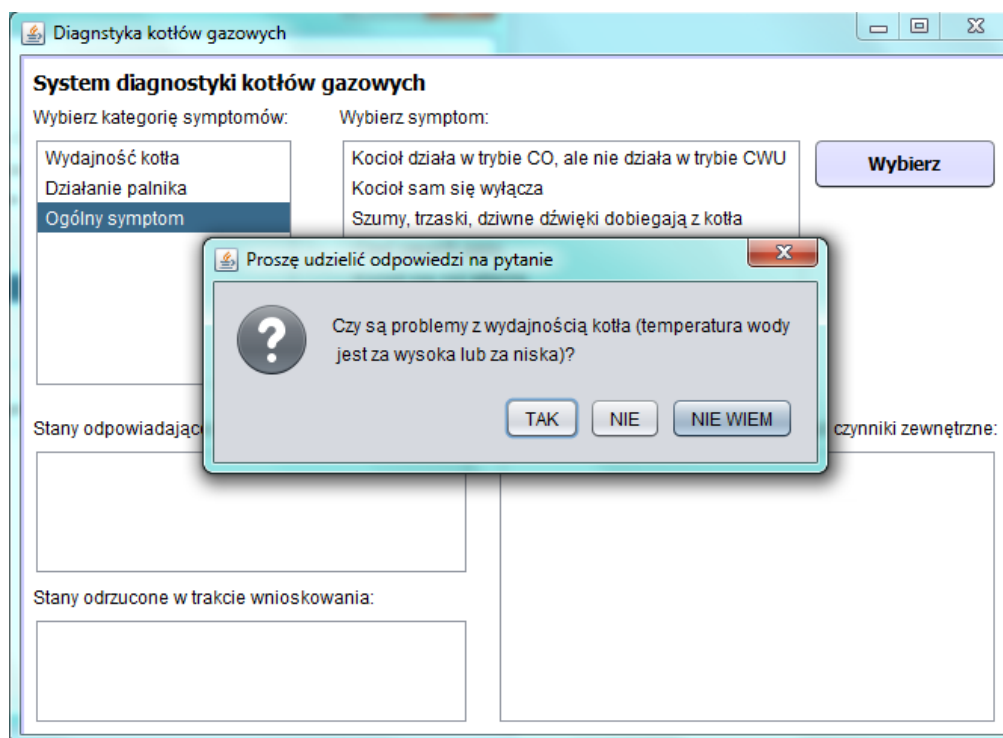


Rysunek 3.10. Wygląd aplikacji po uruchomieniu. Opracowanie własne.



Rysunek 3.11. Wybór symptomu występującego w badanym kotle. Opracowanie własne.

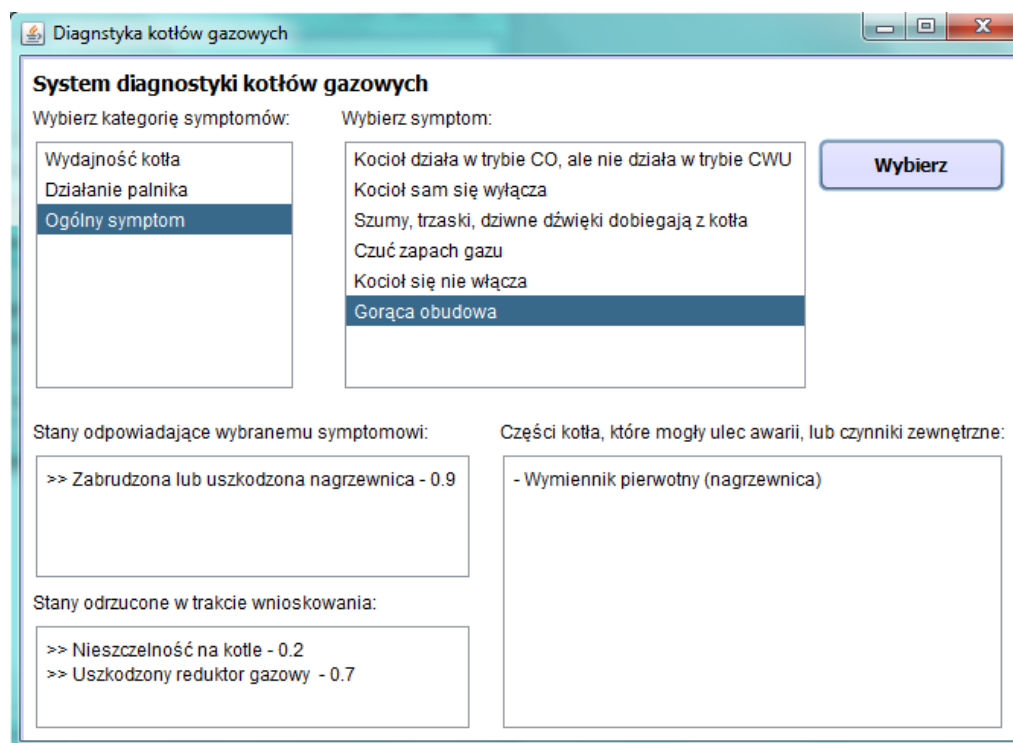
Gdy użytkownik znajdzie symptom, który jest obserwowany w badanym kotle, naciska przycisk „Wybierz”. W tym momencie system tworzy listę możliwych stanów, w których może znajdować się badany kocioł, na podstawie danych zapisanych w tezaurusie. Do stanów przypisany jest stopień rozmycia, jednak w zależności od różnych innych objawów lub informacji ważność stanów może ulec zmianie. Aby uzyskać od użytkownika jak najwięcej informacji na temat badanego kotła, system zadaje dodatkowe pytania (rysunek 3.12.).



Rysunek 3.12. Przykładowe pytanie, które jest zadawane do wybrania symptomu, w trakcie wnioskowania. Opracowanie własne.

Użytkownik, odpowiadając na te pytania, pozwala na ustalenie stopnia rozmycia stanów. Na każde pytanie można odpowiedzieć „Tak”, „Nie” lub „Nie wiem”. Ostatnia odpowiedź została wprowadzona, w przypadku gdy użytkownik nie jest w stanie udzielić odpowiedzi, na przykład gdy nie jest w stanie zaobserwować zabrudzeń na nagrzewnicy lub nie zwrócił uwagi, czy na wyświetlaczu pojawiał się kod błędu.

Następnie aplikacja wybiera stan kotła, który ma najwyższy stopień rozmycia. Aplikacja wyświetla wybrany stan, a także stany, które zostały odrzucone w procesie wnioskowania (rysunek 3.13.). Obok stanów pokazany jest również stopień rozmycia. Dla wybranego stanu lub stanów wymienione zostają części kotła, które mogły ulec awarii lub usterce. Wymienione zostają również czynniki zewnętrzne, jeśli mogły takie zaistnieć i spowodować określony stan kotła.



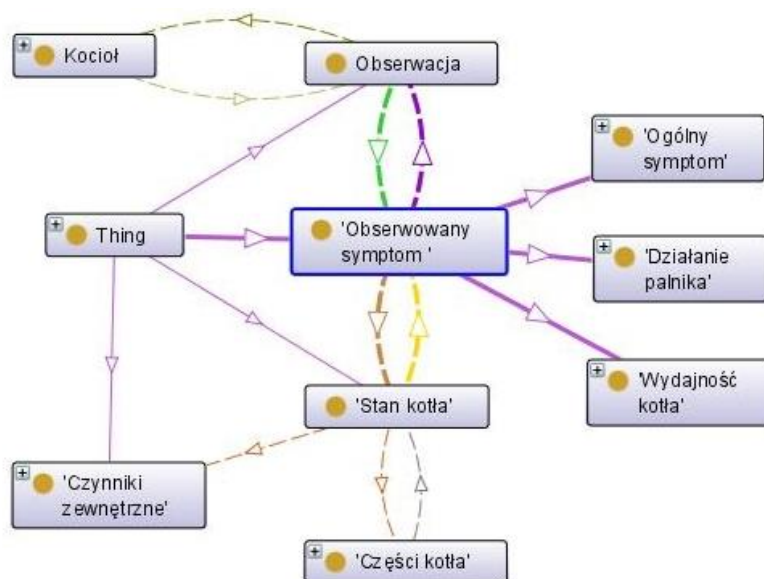
Rysunek 3.13. Przykładowy wynik wnioskowania. Informacja o stanie, w jakim znajduje się kocioł, oraz o częściach, które mogły ulec awarii. Opracowanie własne.

Na rysunku 3.7. przedstawiono główne klasy, które występują w utworzonym teaurusie dla systemu diagnostyki kotłów gazowych. Na rysunku 3.14. przedstawiono te klasy w sposób graficzny, dzięki czemu można zauważyć powiązania utworzone pomiędzy różnymi klasami. Powiązania, czyli w języku logiki opisowej role, a w języku OWL właściwości, jakie występują w utworzonej sieci, to:

- „Jest częścią”, właściwość wiążąca klasy Części kotła – Jest częścią – Kocioł.
- „Ma część”, właściwość wiążąca klasy Kocioł – Ma część – Części kotła, odwrotna do „Jest częścią”.
- „Dotyczy kotła”, właściwość wiążąca klasy Obserwacja – Dotyczy kotła – Kocioł.
- „Wynika z”, właściwość wiążąca klasy Obserwowany symptom – Wynika z – Obserwacja.
- „Pokazuje”, właściwość wiążąca klasy Obserwacja – Pokazuje – Obserwowany symptom, odwrotna do „Wynika z”.
- „Jest poddany obserwacji”, właściwość wiążąca klasy Kocioł – Jest poddany obserwacji – Obserwacja.
- „Jest w stanie”, właściwość wiążąca klasy Części kotła – Jest w stanie – Stan kotła.
- „Oznacza usterkę”, właściwość wiążąca klasy Stan kotła – Oznacza usterkę – Części kotła lub Czynniki zewnętrzne.
- „Powoduje”, właściwość wiążąca klasy Stan kotła – Powoduje – Obserwowany symptom.

- „Wskazuje na”, właściwość wiążąca klasy Obserwowany symptom – Wskazuje na – Stan kotła, odwrotna do „Powoduje”.

Powiązania te są wykorzystywane w aplikacji do ustalania symptomów, stanów, części kotła oraz czynników zewnętrznych.



Rysunek 3.14. Graficzna reprezentacja powiązań między klasami – fragment. Opracowanie własne.

Więcej fragmentów graficznych reprezentacji sieci semantycznej tezaury dla diagnostyki kotłów gazowych, w tym przede wszystkim powiązania pomiędzy poszczególnymi symptomami i stanami kotła oraz stanami i częściami kotła, znajduje się w dodatku B niniejszej pracy.

3.4.9. Oprogramowanie wykorzystane do aplikacji

Do utworzenia tezaury zapisanego w pliku OWL został wykorzystany program **Protégé**. Protégé jest darmowym edytorem ontologii na licencji *open-source* o strukturze opartej na bazie wiedzy. Platforma Protégé pozwala na modelowanie ontologii w formatach OWL, RDF(S) i XML Schema. Ma zaimplementowany bogaty zbiór struktur do modelowania wiedzy, które wspierają tworzenie, wizualizację i edycję ontologii w wielu formatach. Program ten ma możliwość dodawania przyjaznych nazw domen oraz ułatwia wprowadzanie danych do ontologii poprzez intuicyjny interfejs użytkownika.

Protégé jest dostępny jako aplikacja internetowa oraz uruchamiana lokalnie na dysku. Jest on oparty na języku Java, co sprawia, że w łatwy sposób można go poszerzyć o dodatkowe funkcje i biblioteki. Program umożliwia tworzenie ontologii opisywanych przez koncepty, role i aksjomaty, a także tworzenie taksonomii, klasyfikacji i schematów baz danych zgodnie z wytycznymi W3C.

Protégé wspiera dwa główne sposoby modelowania ontologii:

- 1) Protégé-Frames – edytor umożliwiający budowanie i rozpowszechnianie ontologii zgodnych z protokołem Open Knowledge Base Connectivity (OKBC). W tym modelu ontologia zawiera zbiory klas zorganizowane w sposób hierarchiczny reprezentujące najistotniejsze koncepty domen oraz ich właściwości i powiązania oraz konkretne indywidua i ich właściwości.
- 2) Protégé-OWL – edytor umożliwiający budowanie ontologii dla sieci semantycznych Web, w szczególności opartych na języku Web Ontology Language (OWL) określonym przez konsorcjum W3C.

W pracy został wykorzystany edytor Protégé-OWL, zapisujący ontologię w języku OWL, który obecnie jest najnowszym i często wykorzystywanym standardem do opisywania ontologii. Protégé-OWL jest zintegrowany z serwerem Jena, a dzięki licencji *open-source* Java API jest przystosowany do implementacji niestandardowych dostosowanych elementów interfejsu użytkownika lub dowolnych semantycznych usług sieciowych.

Edytor Protégé-OWL umożliwia:

- ładowanie i zapisywanie ontologii jako plików OWL i RDF,
- edycję i wizualizację klas, właściwości i reguł SWRL,
- definiowanie logicznych charakterystyk klas jako wyrażeń OWL,
- wykonywanie wnioskowania, takiego jak klasyfikator logiki opisowej,
- edytowanie indywiduów OWL zgodnie z wytycznymi sieci semantycznych Web.

Obecnie program Protégé jest najczęściej wykorzystywany w projektach biznesowych i naukowych jako edytor ontologii ze względu na swoje szerokie zastosowanie, przyjazny interfejs oraz zgodność ze standardem sieci semantycznych Web [pro14].

Natomiast aplikacja powstała w języku programowania **Java** za pomocą Maven Framework. **Maven** jest narzędziem, które może być używane do budowania i zarządzania każdym projektem opartym na języku Java. Jest on dostępny w środowisku programistycznym NetBeans IDE. Do zadań tej platformy zaliczamy:

- ułatwienie procesu budowania aplikacji,
- stworzenie jednolitego systemu gromadzenia kodów,
- zapewnienie wysokiej jakości informacji o projekcie,
- zapewnienie wytycznych dotyczących najlepszych praktyk programistycznych,
- umożliwienie transparentnej migracji nowych funkcji.

Platforma Maven ma możliwość tworzenia projektów opartych na przetwarzaniu ontologii, dzięki integracji z serwerem **Jena**.

Darmowy serwer Apache Jena służy do budowania sieci semantycznych Web i połączonych aplikacji danych. Zapewnia bogate biblioteki Java, pomagające programistom tworzyć aplikacje, które obsługują RDF, RDFS, RDFa, OWL i SPARQL zgodnie z rekomendacjami W3C. Jena zawiera również silnik wnioskowania oparty na regułach ontologii OWL i RDFS oraz różne strategie pamięci masowej do przechowywania trójek RDF w pamięci lub na dysku.

Jena zapewnia API do pobierania i zapisywania danych z plików RDF lub OWL. Dane te są reprezentowane jako model abstrakcyjny, który może być pozyskiwany z danych z plików, baz danych, adresów URL lub ich kombinacji. Do modelu można tworzyć zapytania z języku SPARQL oraz aktualizować go dzięki SPARUL. Jena wspiera język OWL oraz różne systemy wnioskujące, również uwzględniające zasady logiki opisowej DL, np. *Pellet reasoner* oraz rozmyte systemy wnioskujące, np. *fuzzyDL reasoner*.

4. PODSUMOWANIE I WNIOSKI KOŃCOWE

Autorka niniejszej dysertacji przedstawiła w sposób uporządkowany wiedzę teoretyczną z zakresu systemów wyszukiwania informacji w sieci semantycznej, w tym sieci semantycznej Web, i sformułowała dla opisu tego systemu rozmytą logikę opisową, zwaną w pracy logiką wyszukiwania informacji IRL. Wyniki teoretyczne zostały zreferowane na międzynarodowych konferencjach i opublikowane w czasopismach międzynarodowych z zakresu sztucznej inteligencji. Do oryginalnych pomysłów autorki, prezentowanych w publikacjach i w niniejszej pracy, należą:

- system diagnostyczny jako system pomiarowy,
- paradoks pozyskania wiedzy dokładnej na podstawie posiadanych niedoskonałych (niedokładnych lub niepewnych) reprezentacji tej wiedzy,
- zakres ufności dla pozyskiwanej informacji,
- wyostrzenie wyszukiwania informacji przez wskazanie zasobów internetowych reprezentujących wiedzę w zakresie ufności.

Ponadto zostały wskazane, z wykorzystaniem aparatu teoretycznego, następujące elementy diagnozy technicznej:

- Określenie wiedzy pozwalającej na klasyfikację obiektów diagnostycznych zgodnie z opisującymi je atrybutami: stan obiektu oraz symptomy tego stanu.
- Atrybuty wyznaczające tablicę decyzyjną, w której atrybutami warunkowymi są symptomy, a atrybutem decyzyjnym jest stan obiektu diagnostycznego. Wartościami symptomów są opisy mierzonych danych, a stan jest opisem usterek (uszkodzeń, wad, błędów, zaburzeń itp.), jednej lub zbioru usterek, odpowiadającym wartościom atrybutów.
- Wiedzę reprezentowaną przez tablice decyzyjne. Określono ją jako reprezentowaną adekwatnie lub dokładnie, jeśli występujące w niej dane pozwalają dokładnie wskazać obiekty, do których odnoszą się te dane. W przeciwnym wypadku uznano, że wiedza jest reprezentowana niedoskonale, czyli występuje niepewność, niedokładność czy nieostrość. Gdy atrybuty są zmiennymi lingwistycznymi, wiedza ta jest reprezentowana za pomocą zbiorów rozmytych.
- Ustalenie zbioru atrybutów, których wartości reprezentują adekwatnie lub dokładnie wiedzę o obiektach diagnostycznych. Ponadto wyznaczają one klasyfikacje tworzące hierarchie klas odpowiadające cechom i własnościom tych obiektów, ustalonym przez inżynierów, ekspertów lub projektantów czy wytwórców tych obiektów. Zbiór tych atrybutów ustala tezaurus.
- Rozszerzenie tezaurusu o atrybuty o wartościach rozmytych, których występowanie pozwala dokładnie wyznaczyć stany obiektów diagnostycznych.
- Rozważanie danych opisanych przez tezaurus jako danych pozyskiwanych z ustalonej części sieci internetowej, reprezentującej w swoich zasobach tę samą wiedzę co te dane, zgodnie z postulowaną logiką wyszukiwania informacji (rozdz. 2.4., 2.5.). Zgodnie

z rozumieniem obiektu diagnostycznego zasoby internetowe reprezentujące wiedzę diagnostyczną (wiedzę o obiektach diagnozowanych) są obiektami diagnostycznymi.

- Określenie bazy wiedzy dla dziedziny szukanej wiedzy diagnostycznej, w której wybrane są wyrażenia tezaury i ontologii sformułowane zgodnie z budową języka logiki opisowej. Ponadto przedstawiono sposób rozmywania reprezentacji tej wiedzy oraz zakres ufności wyszukiwania informacji w sieci Web. Dzięki temu możliwe jest określenie wyostrzającej interpretacji dla tej bazy, co pozwala dokonać ostrej klasyfikacji obiektów diagnostycznych opisanych przez tę bazę wiedzy.

Dzięki wprowadzeniu precyzyjnych metod wyszukiwania informacji diagnostycznej, omówionych wyżej, została uzasadniona teza, że:

Interpretacje rozmyte i wyostrzające wyrażen rozmytej logiki opisowej dają możliwość dokonania ostrej klasyfikacji obiektów diagnostycznych opisywanych na podstawie wiedzy wyszukiwanej w sieciach semantycznych.

Rozdział 3. niniejszej pracy prezentuje główny cel badań:

Utworzenie koncepcji systemu wyszukiwania wiedzy w sieciach semantycznych jako systemu dokonującego klasyfikacji obiektów diagnostycznych na podstawie metody wyostrzania wiedzy diagnostycznej reprezentowanej w języku rozmytej logiki opisowej.

Do uzasadnienia powyższej tezy oraz celu głównego dysertacji zostały zrealizowane wszystkie określone cele szczegółowe.

Można wyróżnić trzy etapy realizacji celu głównego, określone przez trzy tematy konceptualno-programistyczne:

- a) Agent wyszukujący automatycznie informację diagnostyczną w tezaurusie, w zadanym zakresie wiedzy.
- b) Interfejs pozwalający określić ontologię diagnozy.
- c) Agent wyszukujący automatycznie informacje diagnostyczne w zasobach internetowych.

Na podstawie wyników teoretycznych autorka dysertacji opracowała ogólny algorytm wyszukiwania informacji diagnostycznych w sieci internetowej (rozd. 3.1.3.). Przedstawiona w pracy koncepcja systemu wyszukiwania wiedzy w sieciach semantycznych zawiera dokładny opis realizacji wszystkich trzech etapów wyróżnionych wyżej. W rozdziale 3.4. została przedstawiona implementacja systemu wyszukiwania wiedzy tylko dla pierwszego tematu. Implementacja pozostałych dwóch tematów, czyli interfejsu określającego ontologię diagnozy oraz agenta wyszukującego automatycznie informacje w zasobach internetowych, ze względu na złożoność programistyczną, jest realizowana przy współpracy wielu ekspertów

i programistów jako duże projekty. Z tego względu implementacja rozszerzonego systemu wyszukiwania informacji pozostaje jedynie w perspektywie realizacji przyszłych badań.

Jednak należy podkreślić, że stworzona przez autorkę implementacja systemu wyszukiwania wiedzy jest w pełni funkcjonalną aplikacją wyszukującą informacje diagnostyczne w stworzonym teaurusie i realizuje wszystkie zadania praktyczne określone jako cele szczegółowe oraz cel główny pracy. Natomiast algorytmy opisane w punktach b) i c) pozwoliłyby jedynie na rozszerzenie tego systemu o wiedzę dostępną z innych źródeł internetowych.

Opisana implementacja systemu wyszukiwania została zrealizowana na przykładzie systemu wyszukiwania wiedzy diagnostycznej dotyczącej diagnozy wybranego typu kotła gazowego. Aplikacja dotyczy utworzenia agenta dostępnego na stronie internetowej teaurusu, reprezentującego wiedzę diagnostyczną opracowaną przez ekspertów, zajmujących się konserwacją i naprawą kotłów gazowych tego typu.

Na potrzeby aplikacji został stworzony teaurus dla kotłów gazowych małej mocy przedstawiony w dodatku A. Teaurus ma strukturę sieci semantycznej i został utworzony przez autorkę pracy w języku OWL DL w programie Protégé. Obecnie bardzo trudno uzyskać dostęp do tego typu plików, w szczególności zawierających wiedzę techniczną. Z tego względu utworzony teaurus może posłużyć również w innych projektach jako wzorcowa sieć semantyczna.

Aplikację można również rozszerzyć na inne typy kotłów gazowych, jak np. kotły kondensacyjne czy kotły o większej mocy. Można też uwzględnić kotły olejowe czy na paliwo stałe lub uwzględnić podobne ze względu na budowę obiekty diagnostyczne, np. termy gazowe. Wadą systemu jest to, że w przypadku, gdy system jest rozszerzany o kolejne diagnozowane obiekty, należy dla tych obiektów rozszerzyć również teaurus. Proces tworzenia teaurusu można częściowo zautomatyzować, jednak i tak musi on przejść weryfikację na podstawie wiedzy pozyskanej od ekspertów.

LITERATURA

- [abe02] Abelson H., Sussman G. J., Sussman J.: *Struktura i interpretacja programów komputerowych*. WNT, Warszawa 2002.
- [ant04] Antoniou G., Harmelen F. van: *Web Ontology Language: OWL*, w: Staab S., Studer R., eds., *Handbook on Ontologies*, Springer Verlag, Berlin 2004.
- [baa03] Baader F., Calvanese D., McGuinness D. L., Nardi D., Patel-Schneider P. F., eds.: *The Description Logic Handbook: Theory, Implementation and Applications*. Cambridge University Press, Cambridge 2003.
- [bac08] Baczyński M., Jayaram B.: *Fuzzy Implications*. *Studies in Fuzziness and Soft Computing*, vol. 231, Springer, Berlin 2008.
- [bac13] Baczyński M., Jayarm B., Mesiar R.: *R-implications and the Exchange Principle: The Case of Border Continuous T-norms*. *Fuzzy Sets and Systems*, vol. 224, s. 93–105, Elsevier, 2013.
- [bae99] Baeza-Yates R., Ribeiro-Neto B.: *Modern Information Retrieval*. Addison Wesley, 1999.
- [bak09] Bąk J., Jędrzejek Cz.: *Semantic Web – technologie, zastosowania, rozwój*. XV Konferencja PLOUG, s. 236–246, Kościelisko, 2009.
- [bau04] Baumeister J.: *Agile Development of Diagnostic Knowledge Systems*. Infix, Akademische Verlagsgesellschaft Aka GmbH, Berlin 2004.
- [bel11] Belard N., Pencil'e Y., Combacau M.: *A theory of Meta-diagnosis: Reasoning About Diagnostic Systems*. 22nd International Joint Conference on Artificial Intelligence (IJCAI'11), s. 731-737, 2011.
- [ber01] Berners-Lee T., Hendler J., Lassila O.: *The Semantic Web*. Scientific American, 2001.
- [bob07] Bobillo F., Straccia U.: *A Fuzzy Description Logic with Product T-norm*. *Proceedings of the IEEE International Conference on Fuzzy Systems (Fuzz IEEE-07)*, s. 652–657, IEEE Computer Society, 2007.
- [bob08] Bobillo F., Straccia U.: *FuzzyDL: An Expressive Fuzzy Description Logic Reasoner*. *IEEE World Congress on Computational Intelligence*, 1–6 June 2008, s. 923–930, Hong Kong 2008.
- [bob09] Bobillo F., Straccia U.: *Fuzzy Description Logics with General T-norms and Datatypes*. *Fuzzy Sets and Systems*, 160(23), s. 3382–3402, 2009.
- [bri04] Brickley D., Guha R. V.: *RDF Vocabulary Description Language 1.0: RDF Schema*. 2004 [<http://www.w3.org/TR/2000/CR-rdf-schema-20000327>, data odczytu: 11.06.2013].
- [bry11a] **Bryniarska A.**: *Rozmywanie i wyszukiwanie wiedzy rozmytej w sieciach semantycznych*. *Zeszyty Naukowe Wydziału Elektroniki, Telekomunikacji i Informatyki Politechniki Gdańskiej*, t. 1, s. 389–394, Gdańsk 2011.
- [bry11b] **Bryniarska A.**: *Adekwatna defuzyfikacja wiedzy rozmytej w sieciach semantycznych*. *Materiały konferencyjne, XIII International PhD Workshop OWD*, s. 249–254, Wisła, 2011.

- [bry12a] **Bryniarska A.**: *The Algorithm of Knowledge Defuzzification in Semantic Network*, w: Grzech A., Borzowski L., Świątek J., Wilimowska Z., eds., *Information Systems Architecture and Technology, Networks Design and Analysis*, s. 23–32, OW Politechniki Wrocławskiej, Wrocław 2012.
- [bry12b] **Bryniarska A.**: *Rozmywanie i wyostrzanie wiedzy w wybranych sieciach semantycznych systemów produkcji*. PAR, nr 4/2012, s. 98–104, 2012.
- [bry12c] **Bryniarska A.**: *Wiedza rozmyta wyszukiwana w sieciach semantycznych i jej model w systemach danych*, w: Jastriebow A., Kuźmińska-Sołśnia B., Raczyńska B., eds., *Technologie Komputerowe w Rozwoju Nauki, Techniki i Edukacji*, s. 13–23, Radom 2012.
- [bry13a] **Bryniarska A.**: *An Information Retrieval Agent in Web Resources*. IADIS International Conference Intelligent Systems and Agents 2013, Proceedings, s. 121–125, Prague, Czech Republic, 2013.
- [bry13b] **Bryniarska A.**: *The Paradox of the Fuzzy Disambiguation in the Information Retrieval*. (IJARAI) International Journal of Advanced Research in Artificial Intelligence, s. 55–58, vol. 2, No. 9, September 2013.
- [bub90] Bubnicki Z.: *Wstęp do systemów ekspertowych*. PWN, Warszawa 1990.
- [bub01] Bubnicki Z.: *Uncertain Variables and Their Application to Decision Making*. IEEE Transactions on Systems, Man, and Cybernetics, Part A: Systems and Humans, vol. 31, No. 6, s. 587–592, 2001.
- [ceg06] Ceglarek R., Rutkowski W.: *Automated Acquisition of Semantic Knowledge to Improve Efficiency of Information Retrieval Systems*. Business Information Systems 10th International Conference, BIS 2006, Proceedings, LNCS, s. 329–341, Springer, 2006.
- [ceg10] Ceglarek R., Haniewicz K., Rutkowski W.: *Semantic Compression for Specialized Information Retrieval Systems*. Advances in Intelligent Information and Database Systems, Studies in Computational Intelligence, Springer, vol. 283, s. 111–121, 2010.
- [cer10] Cerami M., Esteva F., Bou F.: *Decidability of a Description Logic over Infinite-Valued Product Logic*. Proceedings of the 12th International Conference on the Principles of Knowledge Representation and Reasoning (KR 2010), s. 203–213, 2010.
- [cho93] Cholewa W., Moczulski W.: *Diagnostyka techniczna maszyn: pomiary i analiza sygnałów*. Wydawnictwo Politechniki Śląskiej, Gliwice 1993.
- [cho02] Cholewa W.: *Systemy doradcze w diagnostyce technicznej*, w: Korbicz J., Kowalczyk Z., Cholewa W., Kościelny J. M., *Diagnostyka procesów*. WNT, Warszawa 2002.
- [czo85] Czogała E., Pedrycz W.: *Elementy i metody teorii zbiorów rozmytych*. PWN, Warszawa 1985.
- [dav92] Davis R., Hamscher W.: *Model-based Reasoning: Troubleshooting*, w: Hamscher W., Console L., Kleer de J., eds., *Model-Based Diagnosis*, s. 3–24, Morgan Kaufmann Publishers, San Mateo, CA, 1992.
- [du08] Du J., Qi G., Shen Y.: *Lexicographical Inference over Inconsistent DL-based Ontologies*. Proceedings of the 2nd International Conference on Web Reasoning and Rule Systems (RR-08), LNCS 5341, s. 58–73, 2008.

- [du11] Du J., Qi G., Pan J. Z., Shen Y.: *A Decomposition-Based Approach to OWL DL Ontology Diagnosis*. ICTAI, s. 659–664, 2011.
- [fis97] Fisher M., Wooldridge M.: *On the Formal Specification and Verification of Multi-Agent Systems*. International Journal of Cooperative Information Systems, 6(1), s. 37–65, 1997.
- [fla11] Flasiński M.: *Wstęp do sztucznej inteligencji*. PWN, Warszawa 2011.
- [fra97] Franklin S., Graesser A.: *Is It an Agent, or Just a Program?: A Taxonomy for Autonomous Agents*. Proceeding ECAI '96 Proceedings of the Workshop on Intelligent Agents III, Agent Theories, Architectures, and Languages, Springer-Verlag London, s. 21–35, 1997.
- [goc13] Goczyła K., Waloszek W., Zawadzka T., Zawadzki M.: *Bazy wiedzy*. Prezentacja. Katedra Inżynierii Oprogramowania, Wydział Elektroniki, Telekomunikacji i Informatyki Politechnika Gdańska. Online [http://www.eti.pg.gda.pl/katedry/kio/dydaktyka/Bazy_Wiedzy/BazyWiedzy_2.pdf, data odczytu: 02.12.2013].
- [gal04] Galantucci L. M., Percoco G., Spina R.: *Assembly and Disassembly by using Fuzzy Logic & Genetic Algorithms*. International Journal of Advanced Robotic Systems, vol. 1, No. 2 (2004), s. 67–74, 2004.
- [got99] Gottwald S.: *Many-Valued Logic and Fuzzy Set Theory*, w: Hohle U., Rodabaugh S.E., eds., Mathematics of Fuzzy Sets. Logic, Topology, and Measure Theory. The Handbooks of Fuzzy Sets Series, Kluwer Academic Publishers, s. 5–89, 1999.
- [got01] Gottwald S.: *A Treatise on Many-Valued Logics*. Research Studies Press, Baldock, Hertfordshire 2001.
- [haj98] Hájek P.: *Metamathematics of Fuzzy Logic*. Volume 4 of Trends in Logic, Studia Logica Library, Kluwer Academic Publishers, Dordrecht 1998.
- [han01] Han J., Kamber M.: *Data Mining: Concepts and Techniques*. Morgan Kaufmann Publishers, San Francisco 2001.
- [hel01] Heller U.: *Process Oriented Consistency Based Diagnosis: Theory, Implementation and Application*. Infix, Akademische Verlagsgesellschaft Aka GmbH, Berlin 2001.
- [hop05] Hopcroft J. E., Motwani R., Ullman J. D.: *Wprowadzenie do teorii automatów, języków i obliczeń*. PWN, Warszawa 2005.
- [hor00] Horrocks I., Fensel D., Broekstra J., Decker S., Erdmann M., Goble C., Harmelen F. van, Klein M., Staab S., Studer R., Motta E.: *The Ontology Inference Layer OIL*. Technical report, Vrije Universiteit Amsterdam, NL, 2000 [<http://www.ontoknowledge.org/oil>, data odczytu: 12.09.2013].
- [hus04] Hustadt U., Schmidt R. A., Georgieva L.: *A Survey of Decidable First-Order Fragments and Description Logics*. Journal on Relational Methods in Computer Science, vol. 1, s. 251–276, 2004.
- [ibm97] *Simple Network Management Protocol (SNMP) Support*. 1997 [<http://publib.boulder.ibm.com/iserics/v5r2/ic2924/books/c4154120.pdf>, data odczytu: 10.09.2013].
- [jan12] Jantunen I.: *System Architecture for Mobilephone-centric Ambient Intelligence Applications*. Aalto University publication series doctoral dissertations 81/2012, Helsinki 2012.

- [jay11] Jayanthi S. K., Prema S.: *Word Sense Disambiguation in Web Content Mining Using Brill's Tagger Technique*. International Journal of Computer and Electrical Engineering, vol. 3, No. 3, s. 358–362, June 2011.
- [kac86] Kacprzyk J.: *Zbiory rozmyte w analizie systemowej*. PWN, Warszawa 1986.
- [kac01] Kacprzyk J.: *Wieloetapowe sterowanie rozmyte*. WNT, Warszawa 2001.
- [kit99] Kitamura Y., Mizoguchi R.: *An Ontological Analysis of Fault Process and Category of Faults*. Proceeding of Tenth International Workshop on Principles of Diagnosis (DX-99), s. 118–128, June 8–11, 1999.
- [kle00] Klement E. P., Mesiar R., Pap E.: *Triangular Norms*. Kluwer Academic Publishers, Dordrecht 2000.
- [kle04] Klement E. P., Mesiar R., Pap E.: *Triangular Norms. Position paper I: Basic Analytical and Algebraic Properties; Position paper II: General Constructions and Parameterized Families; Position paper III: Continuous T-norms*. Fuzzy Sets and Systems, 143, 2004.
- [kli95] Klir G. J., Yuan B.: *Fuzzy Sets and Fuzzy Logic: Theory and Applications*. Prentice-Hall Inc., New Jersey 1995.
- [kno02] Knosala R. i zespół: *Zastosowanie metod sztucznej inteligencji w inżynierii produkcji*. WNT, Warszawa 2002.
- [kor09] Korbicz J., Kościelny J. M.: *Modelowanie, diagnostyka i sterowanie nadrzędne procesami. Implementacja w systemie DiaSter*. WNT, Warszawa 2009.
- [kor02] Korbicz J., Kowalczyk Z., Cholewa W., Kościelny J. M.: *Diagnostyka procesów*. WNT, Warszawa 2002.
- [kos08] Kościelny M. J., Bartys M., Syfert M.: *Diagnostics of Industrial Processes in Decentralised Structures with Application of Fuzzy Logic*. Proceedings of the 17th World Congress The International Federation of Automatic Control Seoul, Korea, July 6–11, 2008.
- [kow79a] Kowalski R. A.: *Logic for Problem Solving*. New York, North Holland 1979.
- [kow79b] Kowalski S., Mostowski A. W.: *Teoria automatów i lingwistyka matematyczna*. PWN, Warszawa 1979.
- [kry00] Krygier K., Klinke T., Sewerynik J.: *Ogrzewnictwo, wentylacja, klimatyzacja: podręcznik dla technikum*. WSiP, Warszawa 2000.
- [lam06] Lamperti G., Zanella M.: *Flexible Diagnosis of Discrete-event Systems by Similarity-based Reasoning Techniques*. Artificial Intelligence, 170(3), s. 232–297, 2006.
- [lar05] Larose D. T.: *Discovering Knowledge in Data: An Introduction to Data Mining*. Wiley, New York 2005.
- [las03] Lassila O., Swick R. R.: *Resource Description Framework (RDF): Model and Syntax Specification*. Rekomendacja W3C, 2003 [<http://www.w3.org/TR/REC-rdf-syntax/>, data odczytu: 10.06.2013].
- [lig04] Ligęza A.: *Selected Methods of Knowledge Engineering in System Diagnosis*, w: Korbicz J., Kowalczyk Z., Kościelny J. M., Cholewa W., eds., Fault Diagnosis. Models, Artificial Intelligence, Applications, chapter 16, s. 633–668, Springer-Verlag, Berlin 2004.

- [lig05] Ligęza A., Nalepa G. J.: *Conceptual Modeling Implementation and Automated of Rule-based*, w: Zieliński K., Szmuc T., eds., *Software engineering: evolution end emerging technologies*, vol. 130 of *Frontiers in Artificial Intelligence and Applications*, s. 330–340, Amsterdam 2005.
- [lig08] Ligęza A., Kościelny J. M.: *A New Approach to Multiple Fault Diagnosis. Combination of Diagnostic Matrices, Graphs, Algebraic and Rule-based Models. The Case of Two-layer Models*. *International Journal of Applied Mathematics and Computer Science*, 18(4), s. 465–476, 2008.
- [lig10] Ligęza A., Nalepa G. J.: *Metody reprezentacji i analizy systemów regulowych*, w: Szmuc T., Szpyrka M., eds., *Metody formalne w inżynierii oprogramowania systemów czasu rzeczywistego*, WNT, Warszawa 2010.
- [low96] Lowen R.: *Fuzzy Set Theory, Basic Concepts, Techniques and Bibliography*. Kluwer Academic Publishers, Dordrecht 1996.
- [łes08] Łęski J.: *Systemy neuronowo rozmyte*. WNT, Warszawa 2008.
- [mae98] Maes P.: *Intelligent Software*, w: *Labile Ordnungen*, Hans-Bedrow-Institut, 1998.
- [mag78] Magierska S., Perkowski M.: *Analogia i jej modele formalne*, w: Góralski A., eds., *Zadanie, metoda, rozwiązanie*, WNT, s. 75–118, Warszawa 1978.
- [man04] Manning Ch. D., Raghavan P., Schütze H.: *Introduction to Information Retrieval*. Cambridge University Press, Cambridge 2008.
- [man04] Manola F., Miller E.: *RDF Primer*, 2004 [<http://www.w3.org/TR/rdf-primer/>, data odczytu: 10.06.2013].
- [mar07] Markov Z., Larose D. L.: *Data Mining the Web: Uncovering Patterns in Web Content, Structure, and Usage*. John Wiley & Sons, Hoboken, NJ, 2007.
- [men42] Menger K.: *Statistical Metrics*. *Proceedings of the National Academy of Sciences USA*, 8, s. 535–537, 1942.
- [mic14] *MicroGenus Plus Ariston – instrukcja obsługi użytkownika* [http://instrukcja.pl/i/ariston_microgenus_plus_kociol_grzewczy, data odczytu: 10.01.2014].
- [mim13] *MIMOSA – An Operations and Maintenance Information Open System Alliance*, 2013 [<http://www.mimosa.org/>, data odczytu: 20.06.2013].
- [moc97] Moczulski W.: *Metody pozyskiwania wiedzy dla potrzeb diagnostyki maszyn*. *Zeszyty Naukowe Politechniki Śląskiej* nr 1382, s. Mechanika, z. 130. Gliwice 1997.
- [moc02] Moczulski W.: *Diagnostyka techniczna: Metody pozyskiwania wiedzy*. Wydawnictwo Politechniki Śląskiej, Gliwice 2002.
- [mul96] Mulawka J. J.: *Systemy ekspertowe*. WNT, Warszawa 1996.
- [mys10] Myszkowski S.: *Diagnostyka układu paliwowego układu wtrysku benzyny silnika ZI. Kompendium praktycznej wiedzy*. Dodatek techniczny do *Wiadomości Inter Cars S. A.* nr 34, marzec, 2010.
- [ngu97] Nguyen H. T., Walker E. A.: *A First Course in Fuzzy Logic*. CRC Press, Boca Raton 1997.

- [nie00] Niederliński A.: *Regulowe systemy ekspertowe*. Wydawnictwo Pracowni Komputerowej Jacka Skalmierskiego, Gliwice 2000.
- [niz00] Niziński S.: *Elementy eksploatacji obiektów technicznych*. Uniwersytet Warmińsko-Mazurski, Olsztyn 2000.
- [ode00] Odell J. eds.: *Agent Technology*. OMG, green paper produced by the OMG Agent Working Group, 2000 [www.jamesodell.com/ec2000-08-01.pdf, data odczytu: 11.01.2014].
- [owl14] *Web Ontology Language (OWL 2)*. Wersja 2, 2014 [<http://www.w3.org/TR/owl2-overview/>, data odczytu: styczeń 2014].
- [pan08] Pan J. Z., Stamou G., Stoilos G., Thomas E.: *Expressive Querying over Fuzzy DL-Lite Ontologies*. Scalable Querying Services over Fuzzy Ontologies, 17th International World-Wide-Web Conference (WWW 2008), Beijin 2008.
- [pap09] Paprzycki M.: *Agenci programowi jako metodologia tworzenia oprogramowania*, 2009 [http://www.e-informatyka.pl/wiki/Agenci_programowi_jako_metodologia_tworzenia_oprogramowania, data odczytu: 26.09.2013].
- [par97] Parunak H. Van Dyke: *Go to the Ant: Engineering Principles from Natural Multi-Agent Systems*. Annals of Operations Research, Special Issue on Artificial Intelligence and Management Science, Industrial Technology Institute, vol. 75, s. 69–101, Springer-Verlag 1997.
- [paw96] Pawlak M.: *Systemy ekspertowe w eksploatacji maszyn*. Politechnika Lubelska, Lublin 1996.
- [pen90] Peng Y., Reggia J. A.: *Abductive Inference Models for Diagnostic Problem-Solving*. Springer-Verlag, Berlin 1990.
- [pie99] Piegat A.: *Modelowanie i sterowanie rozmyte*. Akademicka Oficyna Wydawnicza EXIT, Warszawa 1999.
- [plw14] *Projekt plWordNet Słowność wersja 2.1*. Politechnika Wroclawska, 2014 [<http://plwordnet.pwr.wroc.pl/wordnet/>, data odczytu: 02.02.2014].
- [poo88] Poole D.: *A Logical Framework for Default Reasoning*. Artificial Intelligence, vol. 36 Issue 1, s. 27–47, 1988.
- [pop73] Pople H. E. Jr.: *On the Mechanization of Abductive Logic*. Proceedings of IJCAI 73, s. 147–152, 1973.
- [pro14] *Oficjalna strona programu Protégé*, 2014 [<http://protege.stanford.edu/>, data odczytu: 20.02.2014].
- [qi09] Qi G., Du J.: *Model-based Revision Operators for Terminologies in Description Logics*. Proceedings of the 21st International Joint Conference on Artificial Intelligence (IJCAI-09), s. 891–897, 2009.
- [rat01] Rataj Z. L.: *Nowoczesne kotły gazowe jako źródła ciepła grzewczego i pary technologicznej. Część I. Wodne kotły gazowe grzewcze niskotemperaturowe*. Gaz, Woda i Technika Sanitarna, nr 2, s. 42–48, 2001.
- [rdf04] *Resource Description Framework (RDF)*. RDF Working Group, 2004 [<http://www.w3.org/RDF/>, data odczytu: 10.06.2013].

- [reg83] Reggia J. A., Nau D. S., Wang P. Y.: *Diagnostic Expert System Based on a Set Covering Model*. International Journal on Man-Machine Studies, 19, s. 437–460, 1983.
- [reg85] Reggia J. A., Nau D. S., Wang P. Y.: *A Formal Model of Diagnostic Inference. Problem Formulation and Decomposition*. Information Sciences, 37, s. 227–256, 1985.
- [rei87] Reiter R.: *A Theory of Diagnosis from First Principles*. Artificial Intelligence, 32, s. 57–95, 1987.
- [roj00] Rojek R. (red.), Bartecki K., Korniak J.: *Zastosowanie sztucznych sieci neuronowych i logiki rozmytej w automatyce*. Oficyna Wydawnicza Politechniki Opolskiej, Opole 2000.
- [roz82] Rozenberg W. J.: *Wstęp do teorii błędów systemów pomiarowych*. PWN, Warszawa 1982.
- [rud11] Rudnik K.: *Koncepcja i implementacja systemu wnioskującego z probabilistyczno-rozmytą bazą wiedzy*. Rozprawa doktorska, Politechnika Opolska, Opole 2011.
- [rus95] Russell S., Norvig P.: *Artificial Intelligence: A Modern Approach*. Prentice-Hall Inc., New Jersey 1995.
- [rut97] Rutkowska D., Piliński M., Rutkowski L.: *Sieci neuronowe, algorytmy genetyczne i systemy rozmyte*, PWN, Warszawa–Łódź 1997.
- [rut05] Rutkowski L.: *Metody i techniki sztucznej inteligencji*. Wydawnictwo Naukowe PWN, Warszawa 2005.
- [ryk06] Rykaczewski K.: *Systemy rozmyte i ich zastosowania*, 2006 [<http://www-users.mat.umk.pl/~mozgun/prace/duszek.pdf>, data odczytu: 11.09.2013].
- [san94] Sanderson M.: *Word Sense Disambiguation and Information Retrieval*. Proceedings of the 17th International ACM SIGIR, Dublin 1994.
- [sar11] Saricicek I., Cimen O.: *A New Diagnosis System Based on Fuzzy Reasoning to Detect Mean and/or Variance Shifts in a Process*. International Journal of Innovative Computing, Information and Control, vol. 7, No. 12, s. 6935–6948, 2011.
- [sch83] Schweizer B., Sklar A.: *Probabilistic Metric Spaces*. North Holland Publ. Comp., Amsterdam 1983.
- [sch98] Schroeder M.: *Autonomous, Model-Based Diagnosis Agents*. Kluwer Academic Publishers, New York 1998.
- [sch05] Schlobach S., Huang Z.: *Inconsistent Ontology Diagnosis: Framework and Prototype*. EU-IST Integrated Project (IP) IST-2003-506826 SEKT, Vrije Universiteit Amsterdam, 2005.
- [sim08] Simou N., Stoilos G., Tzouvaras V., Stamou G., Kollias S.: *Storing and Querying Fuzzy Knowledge in the Semantic Web*. Proceedings of 4th International Workshop on Uncertainty Reasoning for the Semantic Web Sunday 26th October, Karlsruhe, Germany 2008.
- [sim10] Simou N., Mailis T., Stoilos G., Stamou S.: *Optimization Techniques for Fuzzy Description Logics*. Proceedings of 23rd Int. Workshop on Description Logics (DL2010), CEUR-WS 573, s. 244–254, Canada, Waterloo 2010.
- [sow91] Sowa J. F.: *Principles of Semantic Networks. Explorations in the Representation of Knowledge*. Morgan Kaufmann, San Mateo 1991.

- [sto03] Stokoe Ch., Oakes M. P., Tait J.: *Word Sense Disambiguation in Information Retrieval Revisited*. SIGIR, Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval, s. 159–166, 2003.
- [str98] Straccia, U.: *A Fuzzy Description Logic*. Proceedings of the 15th National Conference on Artificial Intelligence (AAAI-98), s. 594–599, USA, Madison 1998.
- [str01] Straccia U.: *Reasoning with Fuzzy Description Logics*. Journal of Artificial Intelligence Research, 14, s. 137–166, 2001.
- [ter78] Terano T. et al.: *Diagnosis of Engine Trouble by Fuzzy Logic*. Proceedings of IFAC World Congress, s. 1621–1628, 1978.
- [tin08] Ting Y., Lu W. B., Chen C. H., Wang G. K.: *A Fuzzy Reasoning Design for Fault Detection and Diagnosis of a Computer-Controlled System*. Engineering Applications of Artificial Intelligence, 21(2), s. 157–170, 2008.
- [tsu09] Tsukamoto Y.: *A Normative Approach to Fuzzy Logic Reasoning Using Residual Implications*. Journal of Advanced Computational Intelligence and Intelligent Informatics, vol. 13, No. 3, s. 262–267, 2009.
- [tur76] Turski W.: *Struktury danych*. WNT, Warszawa 1976.
- [tza89] Tzafestas S. G., eds.: *Knowledge-Based System Diagnosis, Supervision and Control*. Plenum Press, New York, London 1989.
- [uli96] Ulieru M., Smith M. H.: *Dynamic Fuzzy Reasoning Method for Adaptive Diagnostic Systems Design*. Biennial Conference of the North American Fuzzy Information Processing Society – NAFIPS, s. 423–427, 1996.
- [ull88] Ullman J. D.: *Systemy baz danych*. WNT, Warszawa 1988.
- [w3c14] W3C World Wide Web Consortium. Organizacja zajmująca się ustanawianiem standardów pisania i przesyłu stron WWW, 2014 [<http://www.w3.org/>, data odczytu: kwiecień 2014].
- [wai89] Waite W. M., Goos G.: *Konstrukcja kompilatorów*. WNT, Warszawa 1989.
- [wal11a] Walaszek-Babiszewska A.: *Fuzzy Modelling in Stochastic Environment; Theory, Knowledge Bases, Examples*. LAP Lambert Academic Publishing, Saarbrucken, Germany, 2011.
- [wal11b] Walaszek-Babiszewska A.: *Probabilistic – Fuzzy Knowledge Bases for Diagnostic Systems*. Recent Advances in Applied & Biomedical Informatics and Computational Engineering in Systems Applications, 2011.
- [wal13] Walaszek-Babiszewska A.: *Fuzzy Knowledge-based Approach to Diagnosis Tasks in Stochastic Environment*. Computational Intelligence and Informatics (CINTI), IEEE, Budapest 2013.
- [web83] Weber S.: *A General Concept of Fuzzy Connectives, Negations and Implications Based on T-norms and T-conorms*. Fuzzy Sets and Systems, 11, s. 115–134, 1983.
- [wie09] Wierzchoń S. T.: *Elementy teorii zbiorów rozmytych*. Materiał udostępniony na prawach rękopisu, 2009 [<http://www.ipipan.waw.pl/~stw/esi/sterowniki.pdf>, data odczytu: 10.10.2013].

- [woo02] Wooldridge M.: *An Introduction to Multiagent Systems*. John Wiley and Sons Ltd, England 2002.
- [yag83] Yager R.: *Quantifiers in the Formulation of Multiple Objective Decision Functions*. Information Sciences 31, s. 107–139, 1983.
- [yag95] Yager R., Filev D.: *Podstawy modelowania i sterowania rozmytego*. WNT, Warszawa 1995.
- [yam95a] Yamada K.: *Fuzzy Abductive Reasoning for Diagnostic Problems*. Proceedings of IFSA 95, s. 649–652, 1995.
- [yam95b] Yamada K., Mukaidono M.: *Fuzzy Abduction Based on Lukasiewicz Infinite-valued Logic and Its Approximate Solutions*. Proceedings of FUZZ-IEEE / IFES 95, s. 343–350, 1995.
- [yam95c] Yamada K.: *Fuzzy Abductive Reasoning for Diagnostic Problems*. Proceedings of IFSA 95, s. 649–652, 1995.
- [zad65] Zadeh L. A.: *Fuzzy Sets*. Information and Control, vol. 8, No. 3, s. 338–353, 1965.
- [zad75] Zadeh L. A.: *The Concept of a Linguistic Variable and Its Application to Approximate Reasoning*. Part I: Inform. Sci. 8, s. 199–249; Part II: Inform. Sci. 8, s. 301–357; Part III: Inform. Sci. 9, s. 43–80, 1975.
- [zad83] Zadeh L. A.: *A Computational Approach to Fuzzy Quantifiers in Natural Languages*. Computers and Mathematics with Applications, 9, s. 149–184, 1983.
- [zad99] Zadeh L. A.: *From Computing with Numbers to Computing with Words. From Manipulation of Measurements to Manipulation of Perceptions*. IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications, vol. 46, No. 1, s.105–119, 1999.
- [zad02] Zadeh L. A.: *Toward a Perception-based Theory of Probabilistic Reasoning with Imprecise Probabilities*. Imprecise Probability Models and their Applications, vol. 105, Issue 1, s. 233–264, Elsevier Science B.V., 2002.
- [żół12] Żółtowski B., Łukasiewicz M., Kałaczyński T.: *Techniki informatyczne w badaniach stanu maszyn*. Wydawnictwa Uczelniane Uniwersytetu Technologiczno-Przyrodniczego w Bydgoszczy, Bydgoszcz 2012.
- [żół04] Żółtowski B., Cempel C.: *Inżynieria diagnostyki maszyn*. Polskie Towarzystwo Diagnostyki Technicznej, Warszawa, Bydgoszcz, Radom 2004.
- [żół96] Żółtowski B.: *Podstawy diagnostyki maszyn*. Wydawnictwo Uczelniane Akademii Techniczno-Rolniczej, Bydgoszcz 1996.

SPIS RYSUNKÓW

Rysunek 2.1. Schemat ogólny procesu diagnostycznego. Źródło: rys. 6.17 w [żół96].	14
Rysunek 2.2. Schemat struktury systemu pomiarowego [roz82].	16
Rysunek 2.3. Typy przetwarzanej wiedzy w systemie diagnostycznym. Opracowanie własne na podstawie [kor02].	16
Rysunek 2.4. Model procesu pomiarowego [roz82].	17
Rysunek 2.5. Schemat sieci semantycznej. Opracowanie własne.	23
Rysunek 2.6. Przykładowa sieć semantyczna dla diagnostyki technicznej. Opracowanie własne.	24
Rysunek 2.7. Diagram przedstawiający gramatykę struktur zdania „Pompa wodna psuje się często”. Opracowanie własne.	27
Rysunek 2.8. Diagram przedstawiający przykład gramatyki języka wyrażeń algebraicznych. Opracowanie własne.	28
Rysunek 2.9. Powiązanie podmiotu z przedmiotem w stwierdzeniu z sieci semantycznej. Opracowanie własne.	44
Rysunek 2.10. Język RDF zastosowany do identyfikacji zasobów. Źródło [man04].	45
Rysunek 2.11. Dostęp do zasobów w sieci semantycznej Web. Źródło [w3c14].	46
Rysunek 2.12. Język zapytań SPARQL dla RDF. Źródło [w3c14].	47
Rysunek 2.13. Wnioskowanie według schematu wnioskowania w RDFS. Opracowanie własne.	52
Rysunek 2.14. Schemat blokowy diagnostycznego systemu ekspertowego. Źródło [żół12, s. 46].	57
Rysunek 3.1. Model procesu pomiarowego dla wartości jakościowych. Opracowanie własne.	90
Rysunek 3.2. Przykładowa sieć semantyczna opisująca element typu Wałek [kno02].	101
Rysunek 3.3. Schemat szkieletowy sieci semantycznej klasyfikatora konstrukcyjno-technologicznego. Opracowanie własne.	102
Rysunek 3.4. Schemat układu paliwowego pośredniego, wielopunktowego, powrotnego układu wtryskowego, przygotowanego do pomiaru ciśnienia [mys10].	104
Rysunek 3.5. Schemat budowy kotła gazowego jednofunkcyjnego. Opracowanie własne.	108
Rysunek 3.6. Schemat działania systemu wyszukującego wiedzę z diagnostyki kotłów gazowych. Opracowanie własne.	109
Rysunek 3.7. Klasy utworzone w tezaurusie dla kotła gazowego. Opracowanie własne.	131
Rysunek 3.8. Klasyfikator konstrukcyjno-technologiczny dla kotła gazowego. Opracowanie własne.	132
Rysunek 3.9. Pytania zadawane przy starcie aplikacji w celu ustalenia typu kotła gazowego. Opracowanie własne.	132
Rysunek 3.10. Wygląd aplikacji po uruchomieniu. Opracowanie własne.	133
Rysunek 3.11. Wybór symptomu występującego w badanym kotle. Opracowanie własne.	133

Rysunek 3.12. Przykładowe pytanie, które jest zadawane do wybrania symptomu, w trakcie wnioskowania. Opracowanie własne.....	134
Rysunek 3.13. Przykładowy wynik wnioskowania. Informacja o stanie, w jakim znajduje się kot, oraz o częściach, które mogły ulec awarii. Opracowanie własne.	135
Rysunek 3.14. Graficzna reprezentacja powiązań pomiędzy klasami – fragment. Opracowanie własne.	136

SPIS TABEL

Tabela 2.1. Schemat tablicy decyzyjnej [lig10].	39
Tabela 2.2. Tablica obserwacji symptomów S_m dla wybranych cech stanu obiektu X_n . Opracowanie na podstawie [żół12].	58
Tabela 2.3. Schemat tablicy decyzyjnej dla danych diagnostycznych. Opracowanie własne.	60
Tabela 2.4. Schemat tablicy decyzyjnej dla wieloznacznych wartości stanu. Opracowanie własne.	61
Tabela 2.5. Schemat tablicy decyzyjnej dla danych diagnostycznych odpowiadający tabeli 2.2. Opracowanie własne.	61
Tabela 2.6. Schemat macierzy diagnostycznej. Opracowanie własne.	62
Tabela 2.7. Wybrane operatory s-normy [yag95].	67
Tabela 2.8. Wybrane operatory t-normy [yag95].	68
Tabela 2.9. Wybrane t-operacje z parametrem [yag95].	69
Tabela 2.10. Wybrane implikacje rozmyte [łes08].	71
Tabela 3.1. Słownictwo używane w języku OWL i logiki opisowej DL. Opracowanie własne na podstawie [goc13].	98
Tabela 3.2. Wzajemna odpowiedniość logiki DL i języka OWL (konstruktory i aksjomaty). Opracowanie własne na podstawie [goc13].	99
Tabela 3.3. Tablica obserwacji symptomów i odpowiadających im cech stanu diagnostycznego. Opracowanie własne.	112
Tabela 3.4. Kody błędów – zabezpieczenia występujące w kotle Ariston Microgenus Plus [mic14].	113
Tabela 3.5. Inne komunikaty, które nie oznaczają usterki lub awarii, ale normalne działanie kotła Ariston Microgenus Plus [mic14].	114
Tabela 3.6. Stany występujące podczas usterek i awarii oraz odpowiadające im stopnie rozmycia. Opracowanie własne.	115
Tabela 3.7. Powiązanie symptomów ze stanami diagnostycznymi oraz stopień rozmycia tych stanów. Opracowanie własne.	116
Tabela 3.8. Diagnostyczna tablica decyzyjna. Opracowanie własne.	117
Tabela 3.9. Stany kotła i odpowiadające im elementy klasyfikatora konstrukcyjno-technologicznego. Opracowanie własne.	128

DODATEK A

TEZAUZUS W JĘZYKU OWL NA POTRZEBY SYSTEMU DIAGNOSTYKI KOTŁÓW GAZOWYCH

Poniżej przedstawiono tezaurus stworzony przez autorkę pracy, na potrzeby aplikacji opisanej w rozdziale 3.4 niniejszej pracy. Tezaurus został zapisany w języku programowania OWL, który przedstawia dane za pomocą sieci siemantycznej. Do stworzenia przedstawionego niżej pliku wykorzystano program Protégé. Nazwy klas i właściwości zostały zapisane w języku angielskim, jednak dla każdego elementu zostały dodane opisy w języku polskim. Takie rozwiązanie pozwala na rozszerzenie tezausa o synonimy podanych terminów.

```
<?xml version="1.0"?>
<!DOCTYPE rdf:RDF [
  <!ENTITY owl "http://www.w3.org/2002/07/owl#" >
  <!ENTITY xsd "http://www.w3.org/2001/XMLSchema#" >
  <!ENTITY rdfs "http://www.w3.org/2000/01/rdf-schema#" >
  <!ENTITY rdf "http://www.w3.org/1999/02/22-rdf-syntax-ns#" >
  <!ENTITY diag "http://www.semanticweb.org/ontologies/2014/diagnostyka#" >
]><rdf:RDF xmlns="http://www.semanticweb.org/ontologies/2014/diagnostyka#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns:diag="http://www.semanticweb.org/ontologies/2014/diagnostyka#"
  xmlns:owl="http://www.w3.org/2002/07/owl#"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">
  <owl:Ontology rdf:about="http://www.semanticweb.org/ontologies/2014/diagnostyka#">
    <owl:versionInfo>Diagnostic of gaz boilers</owl:versionInfo>
    <rdfs:isDefinedBy>Anna Bryniarska</rdfs:isDefinedBy>
    <rdfs:comment xml:lang="pl">Ontologia do diagnostyki kotłóW gazowych</rdfs:comment>
  </owl:Ontology>
  <owl:ObjectProperty rdf:about="&diag;can_have">
    <rdfs:label xml:lang="pl">może mieć, być w stanie</rdfs:label>
    <rdfs:domain rdf:resource="&diag;boiler_parts"/>
    <rdfs:range rdf:resource="&diag;state"/>
  </owl:ObjectProperty>
  <owl:ObjectProperty rdf:about="&diag;has_observation">
    <rdfs:label xml:lang="pl">ma obserwację, jest poddany obserwacji</rdfs:label>
    <rdfs:domain rdf:resource="&diag;boiler"/>
    <rdfs:range rdf:resource="&diag;obervation"/>
  </owl:ObjectProperty>
  <owl:ObjectProperty rdf:about="&diag;has_part">
    <rdfs:label xml:lang="pl">ma część</rdfs:label>
    <rdfs:domain rdf:resource="&diag;boiler"/>
    <rdfs:range rdf:resource="&diag;boiler_parts"/>
  </owl:ObjectProperty>
  <owl:ObjectProperty rdf:about="&diag;indicates">
    <rdfs:label xml:lang="pl">wskazuje na</rdfs:label>
    <rdfs:range rdf:resource="&diag;state"/>
    <rdfs:domain rdf:resource="&diag;symptom"/>
  </owl:ObjectProperty>
  <owl:ObjectProperty rdf:about="&diag;is_part_of">
    <rdfs:label xml:lang="pl">jest częścią</rdfs:label>
    <rdfs:range rdf:resource="&diag;boiler"/>
    <rdfs:domain rdf:resource="&diag;boiler_parts"/>
    <owl:inverseOf rdf:resource="&diag;has_part"/>
  </owl:ObjectProperty>
```

```

<owl:ObjectProperty rdf:about="&diag;leads_to">
<rdfs:label xml:lang="pl">prowadzi do usterki</rdfs:label>
<rdfs:range rdf:resource="&diag;boiler_parts"/>
<rdfs:range rdf:resource="&diag;external_elements"/>
<rdfs:domain rdf:resource="&diag;state"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:about="&diag;ma_czesc_otwarty">
<rdfs:domain rdf:resource="&diag;boiler_with_open_combustion_chamber"/>
<rdfs:range rdf:resource="&diag;chimney"/>
<rdfs:range rdf:resource="&diag;circulation_pump_technology"/>
<rdfs:range rdf:resource="&diag;electric_technology"/>
<rdfs:range rdf:resource="&diag;gas_technology"/>
<rdfs:subPropertyOf rdf:resource="&diag;has_part"/>
<rdfs:range rdf:resource="&diag;heat_technology"/>
<rdfs:range rdf:resource="&diag;open_combustion_chamber"/>
<rdfs:range rdf:resource="&diag;water_system_CO"/>
<rdfs:range rdf:resource="&diag;water_system_CWU"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:about="&diag;ma_czesc_zamkniety">
<rdfs:domain rdf:resource="&diag;boiler_with_closed_combustion_chamber"/>
<rdfs:range rdf:resource="&diag;chimney"/>
<rdfs:range rdf:resource="&diag;circulation_pump_technology"/>
<rdfs:range rdf:resource="&diag;closed_combustion_chamber"/>
<rdfs:range rdf:resource="&diag;electric_technology"/>
<rdfs:range rdf:resource="&diag;gas_technology"/>
<rdfs:subPropertyOf rdf:resource="&diag;has_part"/>
<rdfs:range rdf:resource="&diag;heat_technology"/>
<rdfs:range rdf:resource="&diag;water_system_CO"/>
<rdfs:range rdf:resource="&diag;water_system_CWU"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:about="&diag;refers_to">
<rdfs:label xml:lang="pl">dotyczy obserwacji</rdfs:label>
<rdfs:range rdf:resource="&diag;obervation"/>
<owl:inverseOf rdf:resource="&diag;shows"/>
<rdfs:domain rdf:resource="&diag;symptom"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:about="&diag;refers_to_boiler">
<rdfs:label xml:lang="pl">dotyczy kotła</rdfs:label>
<rdfs:range rdf:resource="&diag;boiler"/>
<owl:inverseOf rdf:resource="&diag;has_observation"/>
<rdfs:domain rdf:resource="&diag;obervation"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:about="&diag;results_in">
<rdfs:label xml:lang="pl">powoduje</rdfs:label>
<owl:inverseOf rdf:resource="&diag;indicates"/>
<rdfs:domain rdf:resource="&diag;state"/>
<rdfs:range rdf:resource="&diag;symptom"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:about="&diag;shows">
<rdfs:label xml:lang="pl">pokazuje, wskazuje na</rdfs:label>
<rdfs:domain rdf:resource="&diag;obervation"/>
<rdfs:range rdf:resource="&diag;symptom"/>
</owl:ObjectProperty>
<owl:DatatypeProperty rdf:about="&diag;fuzzyDegree">
<rdfs:domain rdf:resource="&diag;state"/>
<rdfs:range>
<rdfs:Datatype>
<owl:onDatatype rdf:resource="&xsd;float"/>
<owl:withRestrictions rdf:parseType="Collection">
<rdf:Description>

```

```

<xsd:minInclusive rdf:datatype="&xsd;integer">0</xsd:minInclusive>
</rdf:Description>
<rdf:Description>
<xsd:maxInclusive rdf:datatype="&xsd;integer">1</xsd:maxInclusive>
</rdf:Description>
</owl:withRestrictions>
</rdfs:Datatype>
</rdfs:range>
</owl:DatatypeProperty>
<owl:Class rdf:about="&diag;3_way_valve_with_motor">
<rdfs:label xml:lang="pl">Zawór 3-drogowy z silnikiem</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;water_system_CO"/>
</owl:Class>
<owl:Class rdf:about="&diag;CO_return">
<rdfs:label xml:lang="pl">Powrót CO</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;water_system_CO"/>
</owl:Class>
<owl:Class rdf:about="&diag;CO_supply">
<rdfs:label xml:lang="pl">Zasilanie CO</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;water_system_CO"/>
</owl:Class>
<owl:Class rdf:about="&diag;CWU_input">
<rdfs:label xml:lang="pl">Wejście CWU</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;water_system_CWU"/>
</owl:Class>
<owl:Class rdf:about="&diag;CWU_output">
<rdfs:label xml:lang="pl">Wyjście CWU</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;water_system_CWU"/>
</owl:Class>
<owl:Class rdf:about="&diag;CWU_sensor">
<rdfs:label xml:lang="pl">Sonda CWU</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;water_system_CWU"/>
</owl:Class>
<owl:Class rdf:about="&diag;Hall_sensor_fan">
<rdfs:label xml:lang="pl">Czujnik Hall&apos;a</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;closed_combustion_chamber"/>
</owl:Class>
<owl:Class rdf:about="&diag;air_connection">
<rdfs:label xml:lang="pl">Podłączenie powietrza</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;closed_combustion_chamber"/>
</owl:Class>
<owl:Class rdf:about="&diag;air_in_heating_system">
<rdfs:label xml:lang="pl">Powietrze w instalacji</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;external_elements"/>
</owl:Class>
<owl:Class rdf:about="&diag;boiler">
<rdfs:label xml:lang="pl">Kocioł</rdfs:label>
</owl:Class>
<owl:Class rdf:about="&diag;boiler_parts">
<rdfs:label xml:lang="pl">Części kotła</rdfs:label>
</owl:Class>
<owl:Class rdf:about="&diag;boiler_with_closed_combustion_chamber">
<rdfs:label xml:lang="pl">Kocioł z zamkniętą komorą spalania</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;boiler"/>
<owl:disjointWith rdf:resource="&diag;boiler_with_open_combustion_chamber"/>
</owl:Class>
<owl:Class rdf:about="&diag;boiler_with_open_combustion_chamber">
<rdfs:label xml:lang="pl">Kocioł z otwartą komorą spalania</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;boiler"/>
</owl:Class>

```

```

<owl:Class rdf:about="&diag;breakdown_gas_reductor">
<rdfs:label xml:lang="pl">Awaria reduktora na zasilaniu gazu</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;external_elements"/>
</owl:Class>
<owl:Class rdf:about="&diag;burner_cover">
<rdfs:label xml:lang="pl">Osłona palnika</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;burner_technology"/>
</owl:Class>
<owl:Class rdf:about="&diag;burner_filter">
<rdfs:label xml:lang="pl">Filtr palnika</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;burner_technology"/>
</owl:Class>
<owl:Class rdf:about="&diag;burner_technology">
<rdfs:label xml:lang="pl">Technologia spalania - Palnik</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;boiler_parts"/>
</owl:Class>
<owl:Class rdf:about="&diag;chimney">
<rdfs:label xml:lang="pl">Komin</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;exhaust_ducting_technology"/>
</owl:Class>
<owl:Class rdf:about="&diag;chimney_orifice">
<rdfs:label xml:lang="pl">Kryza</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;open_combustion_chamber"/>
</owl:Class>
<owl:Class rdf:about="&diag;circulation_pump">
<rdfs:label xml:lang="pl">Pompa cyrkulacyjna z odpowiecznikiem </rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;circulation_pump_technology"/>
</owl:Class>
<owl:Class rdf:about="&diag;circulation_pump_technology">
<rdfs:label xml:lang="pl">Technologia cyrkulacji wody - Pompa cyrkulacyjna </rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;boiler_parts"/>
</owl:Class>
<owl:Class rdf:about="&diag;clogged_dirty_elements_in_water_supply">
<rdfs:label xml:lang="pl">Zanieczyszczone elementy na zasilaniu wody </rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;external_elements"/>
</owl:Class>
<owl:Class rdf:about="&diag;closed_combustion_chamber">
<rdfs:label xml:lang="pl">Zamknięta komora spalania</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;exhaust_ducting_technology"/>
</owl:Class>
<owl:Class rdf:about="&diag;combustion_chamber">
<rdfs:label xml:lang="pl">Komora spalania</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;heat_technology"/>
</owl:Class>
<owl:Class rdf:about="&diag;combustion_chamber_isolation">
<rdfs:label xml:lang="pl">Izolacja komory spalania</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;heat_technology"/>
</owl:Class>
<owl:Class rdf:about="&diag;damaged_high_power_gas_valve">
<rdfs:label xml:lang="pl">Uszkodzony zawór gazowy dużej mocy</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;external_elements"/>
</owl:Class>
<owl:Class rdf:about="&diag;display">
<rdfs:label xml:lang="pl">Wyświetlacz</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;electric_technology"/>
</owl:Class>
<owl:Class rdf:about="&diag;draft_diverter">
<rdfs:label xml:lang="pl">Przerywacz ciągu</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;open_combustion_chamber"/>
</owl:Class>

```

```

<owl:Class rdf:about="&diag;electric_elements">
<rdfs:label xml:lang="pl">Elementy elektryczne</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;electric_technology"/>
</owl:Class>
<owl:Class rdf:about="&diag;electric_module">
<rdfs:label xml:lang="pl">Moduł elektryczny</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;electric_technology"/>
</owl:Class>
<owl:Class rdf:about="&diag;electric_technology">
<rdfs:label xml:lang="pl">Technologia elektroniczna - Zespół elektryczny </rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;boiler_parts"/>
</owl:Class>
<owl:Class rdf:about="&diag;exhaust_ducting_technology">
<rdfs:label xml:lang="pl">Technologia regulacji procesu spalania, odprowadzania spalin</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;boiler_parts"/>
</owl:Class>
<owl:Class rdf:about="&diag;expansion_vessel">
<rdfs:label xml:lang="pl">Naczynie wzbiorcze</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;water_system_CO"/>
</owl:Class>
<owl:Class rdf:about="&diag;external_elements">
<rdfs:label xml:lang="pl">Zewnętrzne czynniki powodujące usterki</rdfs:label>
</owl:Class>
<owl:Class rdf:about="&diag;fan">
<rdfs:label xml:lang="pl">Wentylator modulowany</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;closed_combustion_chamber"/>
</owl:Class>
<owl:Class rdf:about="&diag;filling_valve">
<rdfs:label xml:lang="pl">Zawór napełniania</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;water_system_CO"/>
</owl:Class>
<owl:Class rdf:about="&diag;flow_sensor_CWU">
<rdfs:label xml:lang="pl">Czujnik przepływu wody CWU</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;water_system_CWU"/>
</owl:Class>
<owl:Class rdf:about="&diag;flue_connector">
<rdfs:label xml:lang="pl">Złącze spalin</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;closed_combustion_chamber"/>
</owl:Class>
<owl:Class rdf:about="&diag;flue_gas_analysis_point">
<rdfs:label xml:lang="pl">Punkt analizy spalin komina</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;closed_combustion_chamber"/>
</owl:Class>
<owl:Class rdf:about="&diag;flue_gas_collector">
<rdfs:label xml:lang="pl">Kolektor</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;closed_combustion_chamber"/>
</owl:Class>
<owl:Class rdf:about="&diag;flue_gas_pressure_switch">
<rdfs:label xml:lang="pl">Presostat spalin</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;closed_combustion_chamber"/>
</owl:Class>
<owl:Class rdf:about="&diag;fumes_sensor">
<rdfs:label xml:lang="pl">Sonda spalin</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;open_combustion_chamber"/>
</owl:Class>
<owl:Class rdf:about="&diag;gas_input">
<rdfs:label xml:lang="pl">Wejście gazu</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;gas_technology"/>
</owl:Class>
<owl:Class rdf:about="&diag;gas_meter">

```

```

<rdfs:label xml:lang="pl">Licznik gazowy</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;gas_technology"/>
</owl:Class>
<owl:Class rdf:about="&diag;gas_reducer">
<rdfs:label xml:lang="pl">Reduktor gazowy</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;gas_technology"/>
</owl:Class>
<owl:Class rdf:about="&diag;gas_technology">
<rdfs:label xml:lang="pl">Technologia przepływu gazu - Zespół gazowy </rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;boiler_parts"/>
</owl:Class>
<owl:Class rdf:about="&diag;gas_valve">
<rdfs:label xml:lang="pl">Zawór gazowy</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;gas_technology"/>
</owl:Class>
<owl:Class rdf:about="&diag;general_symptom">
<rdfs:label xml:lang="pl">Ogólny symptom</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;symptom"/>
</owl:Class>
<owl:Class rdf:about="&diag;heat_technology">
<rdfs:label xml:lang="pl">Technologia pozyskiwania ciepła ze spalania - Komora spalania</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;boiler_parts"/>
</owl:Class>
<owl:Class rdf:about="&diag;heater">
<rdfs:label xml:lang="pl">Nagrzewnica</rdfs:label>
<owl:equivalentClass rdf:resource="&diag;primary_heat_exchanger"/>
<rdfs:subClassOf rdf:resource="&diag;heat_technology"/>
</owl:Class>
<owl:Class rdf:about="&diag;hydraulics">
<rdfs:label xml:lang="pl">Technologia hydrauliki wody - Układ hydrauliczny </rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;boiler_parts"/>
</owl:Class>
<owl:Class rdf:about="&diag;ignition_electrode">
<rdfs:label xml:lang="pl">Elektroda zapłonowa</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;burner_technology"/>
</owl:Class>
<owl:Class rdf:about="&diag;ignition_generator">
<rdfs:label xml:lang="pl">Generator zapłonu</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;gas_technology"/>
</owl:Class>
<owl:Class rdf:about="&diag;ionization_electrode">
<rdfs:label xml:lang="pl">Elektroda jonizacyjna</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;burner_technology"/>
</owl:Class>
<owl:Class rdf:about="&diag;low_water_pressure_in_water_supply_system">
<rdfs:label xml:lang="pl">Niskie ciśnienie wody</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;external_elements"/>
</owl:Class>
<owl:Class rdf:about="&diag;minimum_pressure_sensor">
<rdfs:label xml:lang="pl">Czujnik ciśnienia minimalnego</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;water_system_CO"/>
</owl:Class>
<owl:Class rdf:about="&diag;nozzles">
<rdfs:label xml:lang="pl">Dysze palnika</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;burner_technology"/>
</owl:Class>
<owl:Class rdf:about="&diag;obervation">
<rdfs:label xml:lang="pl">Obserwacja</rdfs:label>
</owl:Class>
<owl:Class rdf:about="&diag;open_combustion_chamber">

```

```

<rdfs:label xml:lang="pl">Otwarta komora spalania</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;exhaust_ducting_technology"/>
</owl:Class>
<owl:Class rdf:about="&diag;outside_sensor">
<rdfs:label xml:lang="pl">Sonda zewnętrzna mierzy temperaturę na zewnątrz kotła</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;gas_technology"/>
</owl:Class>
<owl:Class rdf:about="&diag;power">
<rdfs:label xml:lang="pl">Zasilanie</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;electric_technology"/>
</owl:Class>
<owl:Class rdf:about="&diag;primary_heat_exchanger">
<rdfs:label xml:lang="pl">Wymiennik pierwotny (nagrzewnica)</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;water_system_CO"/>
</owl:Class>
<owl:Class rdf:about="&diag;safety_valve">
<rdfs:label xml:lang="pl">Zawór bezpieczeństwa</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;water_system_CO"/>
</owl:Class>
<owl:Class rdf:about="&diag;second_heat_exchanger">
<rdfs:label xml:lang="pl">Wymiennik wtórny</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;water_system_CWU"/>
</owl:Class>
<owl:Class rdf:about="&diag;sensor_on_CO_flow">
<rdfs:label xml:lang="pl">Sonda na zasilaniu CO</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;water_system_CO"/>
</owl:Class>
<owl:Class rdf:about="&diag;sensor_on_CO_return">
<rdfs:label xml:lang="pl">Sonda na powrocie CO</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;water_system_CO"/>
</owl:Class>
<owl:Class rdf:about="&diag;state">
<rdfs:label xml:lang="pl">Stan kotła (usterka lub awaria)</rdfs:label>
</owl:Class>
<owl:Class rdf:about="&diag;symptom">
<rdfs:label xml:lang="pl">Obserwowany symptom </rdfs:label>
</owl:Class>
<owl:Class rdf:about="&diag;symptom_dzialanie_palnika">
<rdfs:label xml:lang="pl">Działanie palnika</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;symptom"/>
</owl:Class>
<owl:Class rdf:about="&diag;symptom_wydajnosc_kotla">
<rdfs:label xml:lang="pl">Wydajność kotła</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;symptom"/>
</owl:Class>
<owl:Class rdf:about="&diag;temperature_sensor">
<rdfs:label xml:lang="pl">Termostat przegrzewu - Czujnik temperatury wody grzewczej</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;water_system_CO"/>
</owl:Class>
<owl:Class rdf:about="&diag;vent">
<rdfs:label xml:lang="pl">Odpowietrznik</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;circulation_pump_technology"/>
</owl:Class>
<owl:Class rdf:about="&diag;water_filter_CO">
<rdfs:label xml:lang="pl">Filtr wody CO</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;water_system_CO"/>
</owl:Class>
<owl:Class rdf:about="&diag;water_filter_CWU">
<rdfs:label xml:lang="pl">Filtr wody na wejściu CWU</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;water_system_CWU"/>

```



```

</owl:Class>
<owl:Class rdf:about="&diag;water_system_CO">
<rdfs:label xml:lang="pl">Zespół wodny CO</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;hydraulics"/>
</owl:Class>
<owl:Class rdf:about="&diag;water_system_CWU">
<rdfs:label xml:lang="pl">Zespół wodny CWU</rdfs:label>
<rdfs:subClassOf rdf:resource="&diag;hydraulics"/>
</owl:Class>
<owl:NamedIndividual rdf:about="&diag;Awaria_reduktora_gazu_na_zasilaniu_do_kotla">
<rdf:type rdf:resource="&diag;breakdown_gas_reductor"/>
<rdfs:label xml:lang="pl">Awaria jednego z reduktorów gazu na zasilaniu doprowadzonego do
kotła</rdfs:label>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;Czujnik_Hall&apos;a_wentylatora">
<rdf:type rdf:resource="&diag;Hall_sensor_fan"/>
<rdfs:label xml:lang="pl">Czujnik Hall&apos;a wentylatora</rdfs:label>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;Filtr_wody_CWU">
<rdf:type rdf:resource="&diag;water_filter_CWU"/>
<rdfs:label xml:lang="pl">Filtr wody na wejściu CWU</rdfs:label>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;Kocioł_Microgenus_Plus">
<rdf:type rdf:resource="&diag;boiler_with_open_combustion_chamber"/>
<rdfs:label xml:lang="pl">Kocioł Microgenus Plus z otwartą komorą spalania</rdfs:label>
<has_part rdf:resource="&diag;Filtr_wody_CWU"/>
<has_part rdf:resource="&diag;Modul_elektroniczny_CMP3-MCU"/>
<has_part rdf:resource="&diag;Pompa_GRUNDFOSS_UPS_15-50"/>
<has_part rdf:resource="&diag;Przylgowa_sonda_NTC_powrot_CO"/>
<has_part rdf:resource="&diag;Przylgowa_sonda_NTC_wyjscie_CWU"/>
<has_part rdf:resource="&diag;Przylgowa_sonda_NTC_zasilanie_CO"/>
<has_part rdf:resource="&diag;Zawor_3_drogowy"/>
<has_part rdf:resource="&diag;Zawor_gazowy_SIT_845_SIGMA"/>
<has_part rdf:resource="&diag;chimney"/>
<has_part rdf:resource="&diag;combustion_chamber"/>
<has_part rdf:resource="&diag;combustion_chamber_isolation"/>
<has_part rdf:resource="&diag;czujnik_cisnienia_minimalnego"/>
<has_part rdf:resource="&diag;czujnik_przeplywu_CWU"/>
<has_part rdf:resource="&diag;dysze_palnika_Polidoro"/>
<has_part rdf:resource="&diag;elektroda_jonizacyjna_Polidoro"/>
<has_part rdf:resource="&diag;elektrody_zaplonowe_Polidoro"/>
<has_part rdf:resource="&diag;expansion_vessel"/>
<has_part rdf:resource="&diag;filtr_palnika_Polidoro"/>
<has_part rdf:resource="&diag;gas_meter"/>
<has_part rdf:resource="&diag;gas_reducer"/>
<has_part rdf:resource="&diag;generator_zaplonu"/>
<has_part rdf:resource="&diag;odpowietrznik_pompy"/>
<has_part rdf:resource="&diag;osłona_palnika_Polidoro"/>
<has_part rdf:resource="&diag;power"/>
<has_part rdf:resource="&diag;primary_heat_exchanger"/>
<has_part rdf:resource="&diag;sonda_temperatury_zewnetrzna"/>
<has_part rdf:resource="&diag;wymiennik_wtorny"/>
<has_part rdf:resource="&diag;zawor_bezpieczenstwa"/>
<has_part rdf:resource="&diag;zawor_napełniania"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;Kocioł_Microgenus_Plus_zamknieta_komora">
<rdf:type rdf:resource="&diag;boiler_with_closed_combustion_chamber"/>
<rdfs:label xml:lang="pl">Kocioł Microgenus Plus z zamkniętą komorą spalania</rdfs:label>
<has_part rdf:resource="&diag;Filtr_wody_CWU"/>
<has_part rdf:resource="&diag;Modul_elektroniczny_CMP3-MCU"/>

```

```

<has_part rdf:resource="&diag;Pompa_GRUNDFOSS_UPS_15-50"/>
<has_part rdf:resource="&diag;Przylgowa_sonda_NTC_powrot_CO"/>
<has_part rdf:resource="&diag;Przylgowa_sonda_NTC_wyjście_CWU"/>
<has_part rdf:resource="&diag;Przylgowa_sonda_NTC_zasilanie_CO"/>
<has_part rdf:resource="&diag;Zawor_3_drogowy"/>
<has_part rdf:resource="&diag;Zawor_gazowy_SIT_845_SIGMA"/>
<has_part rdf:resource="&diag;chimney"/>
<has_part rdf:resource="&diag;combustion_chamber"/>
<has_part rdf:resource="&diag;combustion_chamber_isolation"/>
<has_part rdf:resource="&diag;czujnik_cisnienia_minimalnego"/>
<has_part rdf:resource="&diag;czujnik_przepływu_CWU"/>
<has_part rdf:resource="&diag;dysze_palnika_Polidoro"/>
<has_part rdf:resource="&diag;elektroda_jonizacyjna_Polidoro"/>
<has_part rdf:resource="&diag;elektrody_zapłonowe_Polidoro"/>
<has_part rdf:resource="&diag;expansion_vessel"/>
<has_part rdf:resource="&diag;filtr_palnika_Polidoro"/>
<has_part rdf:resource="&diag;gas_meter"/>
<has_part rdf:resource="&diag;gas_reducer"/>
<has_part rdf:resource="&diag;generator_zapłonu"/>
<has_part rdf:resource="&diag;odpowietrznik_pompy"/>
<has_part rdf:resource="&diag;osłona_palnika_Polidoro"/>
<has_part rdf:resource="&diag;podłączenie_powietrza"/>
<has_part rdf:resource="&diag;power"/>
<has_part rdf:resource="&diag;presostat"/>
<has_part rdf:resource="&diag;primary_heat_exchanger"/>
<has_part rdf:resource="&diag;sonda_temperatury_zewnętrzna"/>
<has_part rdf:resource="&diag;wentylator"/>
<has_part rdf:resource="&diag;wymiennik_wtórny"/>
<has_part rdf:resource="&diag;zawor_bezpieczeństwa"/>
<has_part rdf:resource="&diag;zawor_napełniania"/>
<has_part rdf:resource="&diag;złącze_spalin"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;Kolektor_spalin">
<rdf:type rdf:resource="&diag;flue_gas_collector"/>
<rdfs:label xml:lang="pl">Kolektor spalin</rdfs:label>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;Kryza_komina">
<rdf:type rdf:resource="&diag;chimney_orifice"/>
<rdfs:label xml:lang="pl">Kryza komina</rdfs:label>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;Modul_elektroniczny_CMP3-MCU">
<rdf:type rdf:resource="&diag;electric_module"/>
<rdfs:label xml:lang="pl">Moduł elektryczny</rdfs:label>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;Niewystarczające_cisnienie_wody_w_sieci_wodociągowej">
<rdf:type rdf:resource="&diag;low_water_pressure_in_water_supply_system"/>
<rdfs:label xml:lang="pl">Niewystarczające ciśnienie wody w sieci wodociągowej</rdfs:label>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;Pompa_GRUNDFOSS_UPS_15-50">
<rdf:type rdf:resource="&diag;circulation_pump"/>
<rdfs:label xml:lang="pl">Pompa cyrkulacyjna z odpowietrznikiem </rdfs:label>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;Powietrze_w_instalacji_CO">
<rdf:type rdf:resource="&diag;air_in_heating_system"/>
<rdfs:label xml:lang="pl">Powietrze w instalacji CO</rdfs:label>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;Przerywacz_ciągu">
<rdf:type rdf:resource="&diag;draft_diverter"/>
<rdfs:label xml:lang="pl">Przerywacz ciągu</rdfs:label>
</owl:NamedIndividual>

```

```

<owl:NamedIndividual rdf:about="&diag;Przylgowa_sonda_NTC_powrot_CO">
<rdf:type rdf:resource="&diag;electric_elements"/>
<rdf:type rdf:resource="&diag;sensor_on_CO_return"/>
<rdfs:label xml:lang="pl">Przylgowa sonda na powrocie CO</rdfs:label>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;Przylgowa_sonda_NTC_wyjście_CWU">
<rdf:type rdf:resource="&diag;CWU_sensor"/>
<rdf:type rdf:resource="&diag;electric_elements"/>
<rdfs:label xml:lang="pl">Przylgowa sonda NTC na CWU</rdfs:label>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;Przylgowa_sonda_NTC_zasilanie_CO">
<rdf:type rdf:resource="&diag;electric_elements"/>
<rdf:type rdf:resource="&diag;sensor_on_CO_flow"/>
<rdfs:label xml:lang="pl">Przylgowa sonda na zasilaniu CO</rdfs:label>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;Punkt_analizy_spalin">
<rdf:type rdf:resource="&diag;flue_gas_analysis_point"/>
<rdfs:label xml:lang="pl">Punkt analizy spalin</rdfs:label>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;Uszkodzony_zawor_gazowy_duzej_mocy">
<rdf:type rdf:resource="&diag;damaged_high_power_gas_valve"/>
<rdfs:label xml:lang="pl">Uszkodzony zawór gazowy dużej mocy</rdfs:label>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;Wyswietlacz">
<rdf:type rdf:resource="&diag;display"/>
<rdfs:label xml:lang="pl">Wyświetlacz kotła</rdfs:label>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;Zatkanie_elementow_na_dopływie_wody">
<rdf:type rdf:resource="&diag;clogged_dirty_elements_in_water_supply"/>
<rdfs:label xml:lang="pl">Zatkanie, zabrudzenie elementów na dopływie wody do kotła (np. filtry,
sitko)</rdfs:label>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;Zawor_3_drogowy">
<rdf:type rdf:resource="&diag;3_way_valve_with_motor"/>
<rdfs:label xml:lang="pl">Zawór 3-drogowy sterowany silnikiem elektrycznym</rdfs:label>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;Zawor_gazowy_SIT_845_SIGMA">
<rdf:type rdf:resource="&diag;gas_valve"/>
<rdfs:label xml:lang="pl">Zawór gazowy</rdfs:label>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;air_in_boiler">
<rdf:type rdf:resource="&diag;state"/>
<rdfs:label xml:Lang="pl">Zapowietrzony kocioł</rdfs:label>
<fuzzyDegree rdf:datatype="&xsd;float">0.4</fuzzyDegree>
<leads_to rdf:resource="&diag;Pompa_GRUNDFOSS_UPS_15-50"/>
<leads_to rdf:resource="&diag;Powietrze_w_instalacji_CO"/>
<leads_to rdf:resource="&diag;odpowietrznik_pompy"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;air_in_the_heaters">
<rdf:type rdf:resource="&diag;state"/>
<rdfs:label xml:lang="pl">Zapowietrzzone grzejniki</rdfs:label>
<fuzzyDegree rdf:datatype="&xsd;float">0.4</fuzzyDegree>
<leads_to rdf:resource="&diag;Powietrze_w_instalacji_CO"/>
<leads_to rdf:resource="&diag;zawor_napełniania"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;broken_electricity">
<rdf:type rdf:resource="&diag;state"/>
<rdfs:label xml:lang="pl">Uszkodzona elektronika (moduł elektroniczny, części elektroniczne)</rdfs:label>
<fuzzyDegree rdf:datatype="&xsd;float">0.4</fuzzyDegree>
<leads_to rdf:resource="&diag;Modul_elektroniczny_CMP3-MCU"/>

```

```

<leads_to rdf:resource="&diag;Przylgowa_sonda_NTC_powrot_CO"/>
<leads_to rdf:resource="&diag;Przylgowa_sonda_NTC_wyjście_CWU"/>
<leads_to rdf:resource="&diag;Przylgowa_sonda_NTC_zasilanie_CO"/>
<leads_to rdf:resource="&diag;czujnik_cisnienia_minimalnego"/>
<leads_to rdf:resource="&diag;czujnik_przepływu_CWU"/>
<leads_to rdf:resource="&diag;czujnik_temperatury_spalin"/>
<leads_to rdf:resource="&diag;czujnik_temperatury_wody"/>
<leads_to rdf:resource="&diag;presostat"/>
<leads_to rdf:resource="&diag;sonda_temperatury_zewnętrzna"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;chimney">
<rdf:type rdf:resource="&diag;chimney"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;clogged_dirty_water_filter_CO">
<rdf:type rdf:resource="&diag;state"/>
<rdfs:label xml:lang="pl">Zatkany, zabrudzony filtr wody CO</rdfs:label>
<fuzzyDegree rdf:datatype="&xsd;float">0.8</fuzzyDegree>
<leads_to rdf:resource="&diag;filtr_wody_CO"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;clogged_dirty_water_filter_CWU">
<rdf:type rdf:resource="&diag;state"/>
<rdfs:label xml:lang="pl">Zatkany, zabrudzony filtr wody CWU</rdfs:label>
<fuzzyDegree rdf:datatype="&xsd;float">0.8</fuzzyDegree>
<leads_to rdf:resource="&diag;Filtr_wody_CWU"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;combustion_chamber">
<rdf:type rdf:resource="&diag;combustion_chamber"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;combustion_chamber_isolation">
<rdf:type rdf:resource="&diag;combustion_chamber_isolation"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;czujnik_cisnienia_minimalnego">
<rdf:type rdf:resource="&diag;electric_elements"/>
<rdf:type rdf:resource="&diag;minimum_pressure_sensor"/>
<rdfs:label xml:lang="pl">Czujnik ciśnienia minimalnego CO</rdfs:label>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;czujnik_przepływu_CWU">
<rdf:type rdf:resource="&diag;electric_elements"/>
<rdf:type rdf:resource="&diag;flow_sensor_CWU"/>
<rdfs:label xml:lang="pl">Czujnik przepływu wody na CWU</rdfs:label>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;czujnik_temperatury_spalin">
<rdf:type rdf:resource="&diag;electric_elements"/>
<rdf:type rdf:resource="&diag;fumes_sensor"/>
<rdfs:label xml:lang="pl">Czujnik temperatury spalin (sonda spalin)</rdfs:label>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;czujnik_temperatury_wody">
<rdf:type rdf:resource="&diag;electric_elements"/>
<rdf:type rdf:resource="&diag;temperature_sensor"/>
<rdfs:label xml:lang="pl">Czujnik temperatury wody grzewczej, termostat przegrzewu</rdfs:label>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;damaged_circulation_pump">
<rdf:type rdf:resource="&diag;state"/>
<rdfs:label xml:lang="pl">Uszkodzona pompa cyrkulacyjna</rdfs:label>
<fuzzyDegree rdf:datatype="&xsd;float">0.4</fuzzyDegree>
<leads_to rdf:resource="&diag;Pompa_GRUNDFOSS_UPS_15-50"/>
<leads_to rdf:resource="&diag;odpowietrznik_pompy"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;damaged_dirty_burner_parts">
<rdf:type rdf:resource="&diag;state"/>

```

```

<rdfs:label xml:lang="pl">Zatkane filtry, dysze lub elementy palnika</rdfs:label>
<fuzzyDegree rdf:datatype="xsd:float">0.4</fuzzyDegree>
<leads_to rdf:resource="&diag;dysze_palnika_Polidoro"/>
<leads_to rdf:resource="&diag;filtr_palnika_Polidoro"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;damaged_dirty_second_heat_exchanger">
<rdf:type rdf:resource="&diag;state"/>
<rdfs:label xml:lang="pl">Zanieczyszczony wtórny wymiennik ciepła</rdfs:label>
<fuzzyDegree rdf:datatype="xsd:float">0.6</fuzzyDegree>
<leads_to rdf:resource="&diag;wymiennik_wtórny"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;damaged_dirty_water_system">
<rdf:type rdf:resource="&diag;state"/>
<rdfs:label xml:lang="pl">Uszkodzony lub zanieczyszczony zespół wodny</rdfs:label>
<fuzzyDegree rdf:datatype="xsd:float">0.6</fuzzyDegree>
<leads_to rdf:resource="&diag;Filtr_wody_CWU"/>
<leads_to rdf:resource="&diag;Przylgowa_sonda_NTC_powrot_CO"/>
<leads_to rdf:resource="&diag;Przylgowa_sonda_NTC_wyjście_CWU"/>
<leads_to rdf:resource="&diag;Przylgowa_sonda_NTC_zasilanie_CO"/>
<leads_to rdf:resource="&diag;Zawor_3_drogowy"/>
<leads_to rdf:resource="&diag;czujnik_cisnienia_minimalnego"/>
<leads_to rdf:resource="&diag;czujnik_przepływu_CWU"/>
<leads_to rdf:resource="&diag;czujnik_temperatury_wody"/>
<leads_to rdf:resource="&diag;expansion_vessel"/>
<leads_to rdf:resource="&diag;filtr_wody_CO"/>
<leads_to rdf:resource="&diag;primary_heat_exchanger"/>
<leads_to rdf:resource="&diag;wymiennik_wtórny"/>
<leads_to rdf:resource="&diag;zawor_bezpieczeństwa"/>
<leads_to rdf:resource="&diag;zawor_napełniania"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;damaged_gas_meter">
<rdf:type rdf:resource="&diag;state"/>
<rdfs:label xml:lang="pl">Uszkodzony licznik gazowy</rdfs:label>
<fuzzyDegree rdf:datatype="xsd:float">0.4</fuzzyDegree>
<leads_to rdf:resource="&diag;gas_meter"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;damaged_gas_pressure_regulator">
<rdf:type rdf:resource="&diag;state"/>
<rdfs:label xml:lang="pl">Uszkodzony reduktor gazowy </rdfs:label>
<fuzzyDegree rdf:datatype="xsd:float">0.4</fuzzyDegree>
<leads_to rdf:resource="&diag;gas_reducer"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;damaged_gas_system">
<rdf:type rdf:resource="&diag;state"/>
<rdfs:label xml:lang="pl">Uszkodzony zespół gazowy</rdfs:label>
<fuzzyDegree rdf:datatype="xsd:float">0.4</fuzzyDegree>
<leads_to rdf:resource="&diag;Zawor_gazowy_SIT_845_SIGMA"/>
<leads_to rdf:resource="&diag;gas_meter"/>
<leads_to rdf:resource="&diag;gas_reducer"/>
<leads_to rdf:resource="&diag;generator_zapłonu"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;damaged_ignition_electrode">
<rdf:type rdf:resource="&diag;state"/>
<rdfs:label xml:lang="pl">Uszkodzona elektroda zapłonowa </rdfs:label>
<fuzzyDegree rdf:datatype="xsd:float">0.8</fuzzyDegree>
<leads_to rdf:resource="&diag;elektrody_zapłonowe__Polidoro"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;damaged_ionization_electrode">
<rdf:type rdf:resource="&diag;state"/>
<rdfs:label xml:lang="pl">Uszkodzona elektroda jonizacyjna </rdfs:label>

```

```

<fuzzyDegree rdf:datatype="&xsd;float">0.8</fuzzyDegree>
<leads_to rdf:resource="&diag;elektroda_jonizacyjna__Polidoro"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;damaged_temperature_sensor">
<rdf:type rdf:resource="&diag;state"/>
<rdfs:label xml:lang="pl">Uszkodzony czujnik temperatury wody grzewczej</rdfs:label>
<fuzzyDegree rdf:datatype="&xsd;float">0.6</fuzzyDegree>
<leads_to rdf:resource="&diag;czujnik_temperatury_wody"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;dirty_damaged_heater">
<rdf:type rdf:resource="&diag;state"/>
<rdfs:label xml:lang="pl">Zabrudzona lub uszkodzona nagrzewnica</rdfs:label>
<fuzzyDegree rdf:datatype="&xsd;float">0.4</fuzzyDegree>
<leads_to rdf:resource="&diag;primary_heat_exchanger"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;dysze_palnika_Polidoro">
<rdf:type rdf:resource="&diag;nozzles"/>
<rdfs:label xml:lang="pl">Dysze palnika</rdfs:label>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;elektroda_jonizacyjna__Polidoro">
<rdf:type rdf:resource="&diag;ionization_electrode"/>
<rdfs:label xml:lang="pl">Elektroda jonizacyjna</rdfs:label>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;elektrody_zaplonowe__Polidoro">
<rdf:type rdf:resource="&diag;ignition_electrode"/>
<rdfs:label xml:lang="pl">Elektroda zapłonowa</rdfs:label>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;expansion_vessel">
<rdf:type rdf:resource="&diag;expansion_vessel"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;faulty_constructed_chimney">
<rdf:type rdf:resource="&diag;state"/>
<rdfs:label xml:lang="pl">Wadliwie wykonany komin</rdfs:label>
<fuzzyDegree rdf:datatype="&xsd;float">0.8</fuzzyDegree>
<leads_to rdf:resource="&diag;Kolektor_spalin"/>
<leads_to rdf:resource="&diag;Kryza_komina"/>
<leads_to rdf:resource="&diag;Przerywacz_ciagu"/>
<leads_to rdf:resource="&diag;chimney"/>
<leads_to rdf:resource="&diag;podłączenie_powietrza"/>
<leads_to rdf:resource="&diag;złącze_spalin"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;filtr_palnika_Polidoro">
<rdf:type rdf:resource="&diag;burner_filter"/>
<rdfs:label xml:lang="pl">Filtr palnika</rdfs:label>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;filtr_wody_CO">
<rdf:type rdf:resource="&diag;water_filter_CO"/>
<rdfs:label xml:lang="pl">Filtr wody na CO</rdfs:label>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;gas_meter">
<rdf:type rdf:resource="&diag;gas_meter"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;gas_reducer">
<rdf:type rdf:resource="&diag;gas_reducer"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;generator_zaplonu">
<rdf:type rdf:resource="&diag;ignition_generator"/>
<rdfs:label xml:lang="pl">Generator zapłonu</rdfs:label>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;go_out_after_long_time">

```

```

<rdf:type rdf:resource="&diag;symptom_dzialanie_palnika"/>
<rdfs:label xml:lang="pl">Palnik gaśnie po dłuższym czasie</rdfs:label>
<indicates rdf:resource="&diag;damaged_circulation_pump"/>
<indicates rdf:resource="&diag;damaged_dirty_water_system"/>
<indicates rdf:resource="&diag;damaged_gas_system"/>
<indicates rdf:resource="&diag;damaged_temperature_sensor"/>
<indicates rdf:resource="&diag;protection_activated"/>
<indicates rdf:resource="&diag;superheat_boiler"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;hot_housing">
<rdf:type rdf:resource="&diag;general_symptom"/>
<rdfs:label xml:lang="pl">Gorąca obudowa </rdfs:label>
<indicates rdf:resource="&diag;damaged_gas_pressure_regulator"/>
<indicates rdf:resource="&diag;dirty_damaged_heater"/>
<indicates rdf:resource="&diag;leak_on_boiler"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;is_spark_but_do_not_light">
<rdf:type rdf:resource="&diag;symptom_dzialanie_palnika"/>
<rdfs:label xml:lang="pl">Słychać lub widać iskrę ale palnik się nie zapala</rdfs:label>
<indicates rdf:resource="&diag;broken_electricity"/>
<indicates rdf:resource="&diag;damaged_gas_system"/>
<indicates rdf:resource="&diag;damaged_ignition_electrode"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;leak_on_boiler">
<rdf:type rdf:resource="&diag;state"/>
<rdfs:label xml:lang="pl">Nieszczelność na kotle</rdfs:label>
<fuzzyDegree rdf:datatype="&xsd;float">0.4</fuzzyDegree>
<leads_to rdf:resource="&diag;Kolektor_spalin"/>
<leads_to rdf:resource="&diag;Kryza_komina"/>
<leads_to rdf:resource="&diag;Przerywacz_ciagu"/>
<leads_to rdf:resource="&diag;Punkt_analazy_spalin"/>
<leads_to rdf:resource="&diag;combustion_chamber_isolation"/>
<leads_to rdf:resource="&diag;czujnik_temperatury_spalin"/>
<leads_to rdf:resource="&diag;osłona_palnika_Polidoro"/>
<leads_to rdf:resource="&diag;podłączenie_powietrza"/>
<leads_to rdf:resource="&diag;presostat"/>
<leads_to rdf:resource="&diag;złącze_spalin"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;light_up_and_go_down_out_few_seconds">
<rdf:type rdf:resource="&diag;symptom_dzialanie_palnika"/>
<rdfs:label xml:lang="pl">Zapala się palnik, po kilku sekundach gaśnie</rdfs:label>
<indicates rdf:resource="&diag;damaged_dirty_water_system"/>
<indicates rdf:resource="&diag;damaged_gas_system"/>
<indicates rdf:resource="&diag;damaged_ionization_electrode"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;no_gas_or_low_pressure">
<rdf:type rdf:resource="&diag;state"/>
<rdfs:label xml:lang="pl">Brak dopływu gazu (lub niskie ciśnienie gazu) </rdfs:label>
<fuzzyDegree rdf:datatype="&xsd;float">0.4</fuzzyDegree>
<leads_to rdf:resource="&diag;Awaria_reduktora_gazu_na_zasilaniu_do_kotła"/>
<leads_to rdf:resource="&diag;Uszkodzony_zawor_gazowy_duzej_mocy"/>
<leads_to rdf:resource="&diag;Zawor_gazowy_SIT_845_SIGMA"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;no_power">
<rdf:type rdf:resource="&diag;state"/>
<rdfs:label xml:lang="pl">Brak zasilania (problem z bezpiecznikami, zwarcie)</rdfs:label>
<fuzzyDegree rdf:datatype="&xsd;float">0.4</fuzzyDegree>
<leads_to rdf:resource="&diag;power"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;no_water_or_low_pressure">

```

```

<rdf:type rdf:resource="&diag;state"/>
<rdfs:label xml:lang="pl">Brak wody w instalacji lub kotle (lub niskie ciśnienie)</rdfs:label>
<fuzzyDegree rdf:datatype="&xsd;float">0.6</fuzzyDegree>
<leads_to rdf:resource="&diag;Filtr_wody_CWU"/>
<leads_to rdf:resource="&diag;Niewystarczajace_cisnienie_wody_w_sieci_wodociagowej"/>
<leads_to rdf:resource="&diag;Przylgowa_sonda_NTC_powrot_CO"/>
<leads_to rdf:resource="&diag;Przylgowa_sonda_NTC_wyjście_CWU"/>
<leads_to rdf:resource="&diag;Przylgowa_sonda_NTC_zasilanie_CO"/>
<leads_to rdf:resource="&diag;Zawor_3_drogowy"/>
<leads_to rdf:resource="&diag;czujnik_cisnienia_minimalnego"/>
<leads_to rdf:resource="&diag;czujnik_przepływu_CWU"/>
<leads_to rdf:resource="&diag;expansion_vessel"/>
<leads_to rdf:resource="&diag;filtr_wody_CO"/>
<leads_to rdf:resource="&diag;zawor_napełniania"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;not_hot_water_or_low_temperature">
<rdf:type rdf:resource="&diag;symptom_wydajnosć_kotla"/>
<rdfs:label xml:lang="pl">Brak ciepłej wody lub zbyt niska temperatura CWU</rdfs:label>
<indicates rdf:resource="&diag;clogged_dirty_water_filter_CWU"/>
<indicates rdf:resource="&diag;damaged_dirty_second_heat_exchanger"/>
<indicates rdf:resource="&diag;dirty_damaged_heater"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;not_turn_up">
<rdf:type rdf:resource="&diag;general_symptom"/>
<rdfs:label xml:lang="pl">Kocioł nie włącza się</rdfs:label>
<indicates rdf:resource="&diag;air_in_the_heaters"/>
<indicates rdf:resource="&diag;broken_electricity"/>
<indicates rdf:resource="&diag;damaged_gas_meter"/>
<indicates rdf:resource="&diag;damaged_gas_pressure_regulator"/>
<indicates rdf:resource="&diag;faulty_constructed_chimney"/>
<indicates rdf:resource="&diag;no_gas_or_low_pressure"/>
<indicates rdf:resource="&diag;no_power"/>
<indicates rdf:resource="&diag;no_water_or_low_pressure"/>
<indicates rdf:resource="&diag;protection_activated"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;odpowietrznik_pompy">
<rdf:type rdf:resource="&diag;vent"/>
<rdfs:label xml:lang="pl">Odpowietrznik pompy cyrkulacyjnej</rdfs:label>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;osłona_palnika_Polidoro">
<rdf:type rdf:resource="&diag;burner_cover"/>
<rdfs:label xml:lang="pl">Osłona palnika</rdfs:label>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;podłączenie_powietrza">
<rdf:type rdf:resource="&diag;air_connection"/>
<rdfs:label xml:lang="pl">Podłączenie powietrza</rdfs:label>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;poor_efficiency">
<rdf:type rdf:resource="&diag;symptom_wydajnosć_kotla"/>
<rdfs:label xml:lang="pl">Słaba wydajność kotła (CO)</rdfs:label>
<indicates rdf:resource="&diag;clogged_dirty_water_filter_CO"/>
<indicates rdf:resource="&diag;damaged_temperature_sensor"/>
<indicates rdf:resource="&diag;dirty_damaged_heater"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;power">
<rdf:type rdf:resource="&diag;power"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;powrot_CO">
<rdf:type rdf:resource="&diag;CO_return"/>
<rdfs:label xml:lang="pl">Powrót CO</rdfs:label>

```



```

</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;presostat">
<rdf:type rdf:resource="&diag;electric_elements"/>
<rdf:type rdf:resource="&diag;flue_gas_pressure_switch"/>
<rdfs:label xml:lang="pl">Presostat spalin</rdfs:label>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;primary_heat_exchanger">
<rdf:type rdf:resource="&diag;primary_heat_exchanger"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;protection_activated">
<rdf:type rdf:resource="&diag;state"/>
<rdfs:label xml:lang="pl">Zadziałanie zabezpieczenia kotła </rdfs:label>
<fuzzyDegree rdf:datatype="&xsd;float">0.8</fuzzyDegree>
<leads_to rdf:resource="&diag;Modul_elektroniczny_CMP3-MCU"/>
<leads_to rdf:resource="&diag;Niewystarczajace_cisnienie_wody_w_sieci_wodociagowej"/>
<leads_to rdf:resource="&diag;Zawor_gazowy_SIT_845_SIGMA"/>
<leads_to rdf:resource="&diag;dysze_palnika_Polidoro"/>
<leads_to rdf:resource="&diag;elektroda_jonizacyjna_Polidoro"/>
<leads_to rdf:resource="&diag;elektrody_zaplonowe_Polidoro"/>
<leads_to rdf:resource="&diag;filtr_palnika_Polidoro"/>
<leads_to rdf:resource="&diag;presostat"/>
<leads_to rdf:resource="&diag;wentylator"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;smell_gas">
<rdf:type rdf:resource="&diag;general_symptom"/>
<rdfs:label xml:lang="pl">Czuć zapach gazu </rdfs:label>
<indicates rdf:resource="&diag;damaged_gas_system"/>
<indicates rdf:resource="&diag;dirty_damaged_heater"/>
<indicates rdf:resource="&diag;leak_on_boiler"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;sonda_temperatury_zewnetrzna">
<rdf:type rdf:resource="&diag;electric_elements"/>
<rdf:type rdf:resource="&diag;outside_sensor"/>
<rdfs:label xml:lang="pl">Zewnętrzna sonda temperatury</rdfs:label>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;strange_sounds">
<rdf:type rdf:resource="&diag;general_symptom"/>
<rdfs:label xml:lang="pl">Szumy, trzaski, dziwne dźwięki dobiegają z kotła</rdfs:label>
<indicates rdf:resource="&diag;air_in_the_heaters"/>
<indicates rdf:resource="&diag;clogged_dirty_water_filter_CO"/>
<indicates rdf:resource="&diag;clogged_dirty_water_filter_CWU"/>
<indicates rdf:resource="&diag;damaged_circulation_pump"/>
<indicates rdf:resource="&diag;dirty_damaged_heater"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;superheat_boiler">
<rdf:type rdf:resource="&diag;state"/>
<rdfs:label xml:lang="pl">Przegrzew kotła</rdfs:label>
<fuzzyDegree rdf:datatype="&xsd;float">0.4</fuzzyDegree>
<leads_to rdf:resource="&diag;Czujnik_Hall&apos;a_wentylatora"/>
<leads_to rdf:resource="&diag;Punkt_analazy_spalin"/>
<leads_to rdf:resource="&diag;czujnik_temperatury_spalin"/>
<leads_to rdf:resource="&diag;czujnik_temperatury_wody"/>
<leads_to rdf:resource="&diag;wentylator"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;turns_itself_off">
<rdf:type rdf:resource="&diag;general_symptom"/>
<rdfs:label xml:lang="pl">Kocioł sam się wyłącza</rdfs:label>
<indicates rdf:resource="&diag;air_in_the_heaters"/>
<indicates rdf:resource="&diag;faulty_constructed_chimney"/>
</owl:NamedIndividual>

```

```

<owl:NamedIndividual rdf:about="&diag;wejście_CWU">
<rdf:type rdf:resource="&diag;CWU_input"/>
<rdfs:label xml:lang="pl">Wejście CWU</rdfs:label>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;wejście_gazu">
<rdf:type rdf:resource="&diag;gas_input"/>
<rdfs:label xml:lang="pl">Wejście gazu</rdfs:label>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;wentylator">
<rdf:type rdf:resource="&diag;fan"/>
<rdfs:label xml:lang="pl">Wentylator modulowany</rdfs:label>
<is_part_of rdf:resource="&diag;Kociol_Microgenus_Plus"/>
<is_part_of rdf:resource="&diag;Kociol_Microgenus_Plus_zamknieta_komora"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;work_CO_not_work_CWU">
<rdf:type rdf:resource="&diag;general_symptom"/>
<rdfs:label xml:lang="pl">Kocioł działa w trybie CO, ale nie działa w trybie CWU</rdfs:label>
<indicates rdf:resource="&diag;air_in_boiler"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;work_but_flame_is_small">
<rdf:type rdf:resource="&diag;symptom_dzialanie_palnika"/>
<rdfs:label xml:lang="pl">Palnik działa, ale jest mały płomień</rdfs:label>
<indicates rdf:resource="&diag;damaged_dirty_burner_parts"/>
<indicates rdf:resource="&diag;damaged_gas_system"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;wyjście_CWU">
<rdf:type rdf:resource="&diag;CWU_output"/>
<rdfs:label xml:lang="pl">Wyjście CWU</rdfs:label>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;wymyennik_wtórny">
<rdf:type rdf:resource="&diag;second_heat_exchanger"/>
<rdfs:label xml:lang="pl">Wymyennik wtórny</rdfs:label>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;zasilanie_CO">
<rdf:type rdf:resource="&diag;CO_supply"/>
<rdfs:label xml:lang="pl">Zasilanie CO</rdfs:label>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;zawor_bezpieczeństwa">
<rdf:type rdf:resource="&diag;safety_valve"/>
<rdfs:label xml:lang="pl">Zawór bezpieczeństwa</rdfs:label>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;zawor_napełniania">
<rdf:type rdf:resource="&diag;filling_valve"/>
<rdfs:label xml:lang="pl">Zawór napełniania</rdfs:label>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&diag;złącze_spalin">
<rdf:type rdf:resource="&diag;flue_connector"/>
<rdfs:label xml:lang="pl">Złącze spalin</rdfs:label>
</owl:NamedIndividual>
</rdf:RDF><!-- Generated by the OWL API (version 3.4.2) http://owlapi.sourceforge.net -->

```

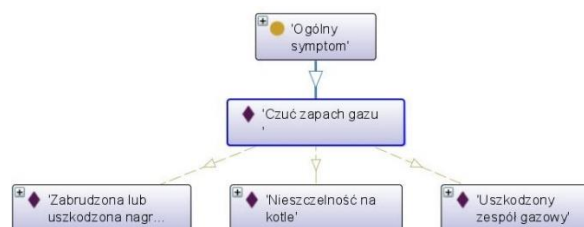
DODATEK B

REPREZENTACJA GRAFICZNA SIECI SEMANTYCZNEJ TEZURUSA

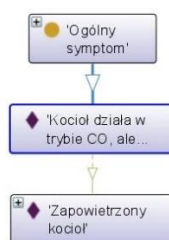
Poniżej przedstawiono fragmenty sieci semantycznej tezaurusu utworzonego do systemu diagnostyki kotłów gazowych. Rysunki B.1. do B.12. przedstawiają stany odpowiadające poszczególnym symptomom.



Rysunek B.1. Stany odpowiadające symptomowi „Brak ciepłej wody lub zbyt niska temperatura CWU”.



Rysunek B.2. Stany odpowiadające symptomowi „Czuć zapach gazu”.



Rysunek B.3. Stany odpowiadające symptomowi „Kocioł działa w trybie CO, ale nie działa w trybie CWU”.



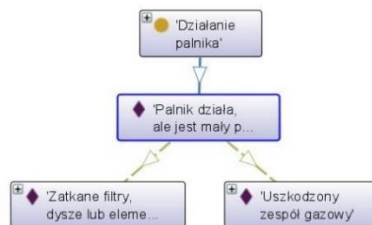
Rysunek B.4. Stany odpowiadające symptomowi „Palnik gaśnie po dłuższym czasie”.



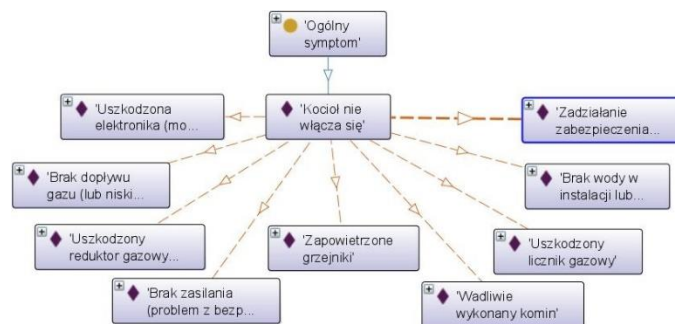
Rysunek B.5. Stany odpowiadające symptomowi „Zapala się palnik, po kilku sekundach gaśnie”.



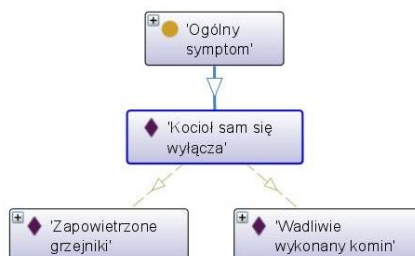
Rysunek B.6. Stany odpowiadające symptomowi „Gorąca obudowa”.



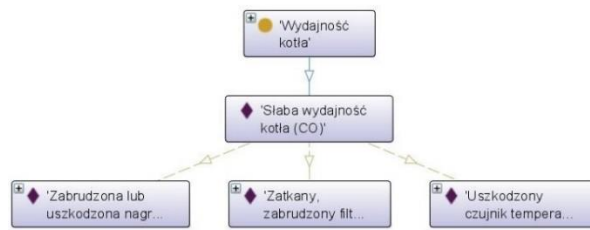
Rysunek B.7. Stany odpowiadające symptomowi „Palnik działa, ale jest mały płomień”.



Rysunek B.8. Stany odpowiadające symptomowi „Kocioł nie włącza się”.



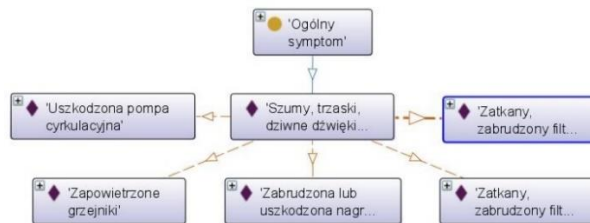
Rysunek B.9. Stany odpowiadające symptomowi „Kocioł sam się wyłącza”.



Rysunek B.10. Stany odpowiadające symptomowi „Słaba wydajność kotła (CO)”.



Rysunek B.11. Stany odpowiadające symptomowi „Słychać lub widać iskrę, ale palnik się nie zapala”.



Rysunek B.12. Stany odpowiadające symptomowi „Szumy, trzaski, dziwne dźwięki dobiegają z kotła”.

Na rysunkach B.13. do B.35. przedstawiono w sposób graficzny powiązania pomiędzy stanami, a częściami kotła gazowego, zapisane w tezaurusie.



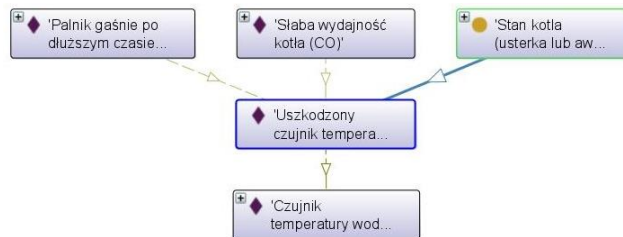
Rysunek B.13. Stan „Brak dopływu gazu (lub niskie ciśnienie gazu)” i odpowiadające mu części kotła, które mogły ulec awarii.



Rysunek B.14. Stan „Brak wody w instalacji lub kotle (lub niskie ciśnienie)” i odpowiadające mu części kotła, które mogły ulec awarii.



Rysunek B.15. Stan „Brak zasilania (problem z bezpiecznikami, zwarcie)” i odpowiadające mu części kotła, które mogły ulec awarii.



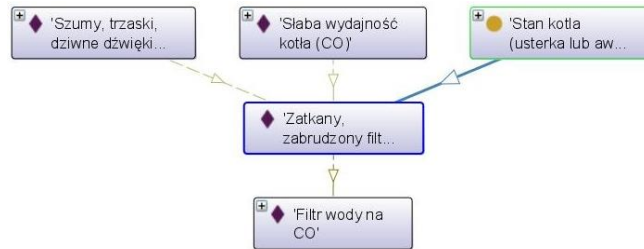
Rysunek B.16. Stan „Uszkodzony czujnik temperatury wody grzewczej” i odpowiadające mu części kotła, które mogły ulec awarii.



Rysunek B.17. Stan „Uszkodzona elektroda jonizacyjna” i odpowiadające mu części kotła, które mogły ulec awarii.



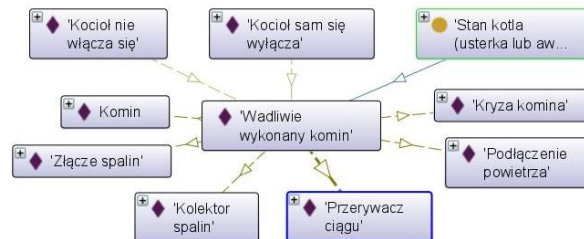
Rysunek B.18. Stan „Uszkodzona elektroda zapłonowa” i odpowiadające mu części kotła, które mogły ulec awarii.



Rysunek B.19. Stan „Zatkany, zabrudzony filtr wody CO” i odpowiadające mu części kotła, które mogły ulec awarii.



Rysunek B.20. Stan „Zatkany, zabrudzony filtr wody CWU” i odpowiadające mu części kotła, które mogły ulec awarii.



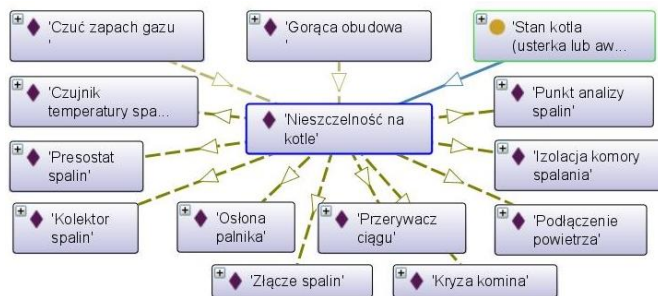
Rysunek B.21. Stan „Wadliwie wykonany komin” i odpowiadające mu części kotła, które mogły ulec awarii.



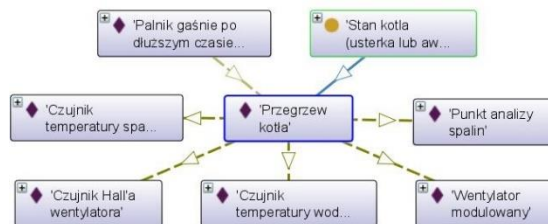
Rysunek B.22. Stan „Uszkodzony licznik gazowy” i odpowiadające mu części kotła, które mogły ulec awarii.



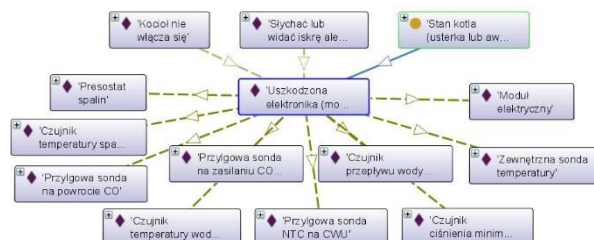
Rysunek B.23. Stan „Zabrudzona lub uszkodzona nagrzewnica” i odpowiadające mu części kotła, które mogły ulec awarii.



Rysunek B.24. Stan „Nieszczelność na kotle” i odpowiadające mu części kotła, które mogły ulec awarii.



Rysunek B.25. Stan „Przegrzew kotła” i odpowiadające mu części kotła, które mogły ulec awarii.



Rysunek B.26. Stan „Uszkodzona elektronika (moduł elektroniczny, części elektroniczne)” i odpowiadające mu części kotła, które mogły ulec awarii.



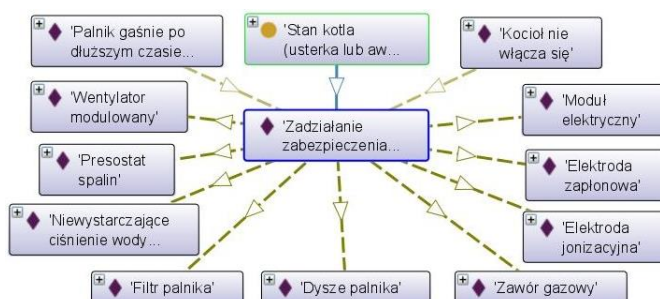
Rysunek B.27. Stan „Uszkodzona pompa cyrkulacyjna” i odpowiadające mu części kotła, które mogły ulec awarii.



Rysunek B.28. Stan „Uszkodzony reduktor gazowy” i odpowiadające mu części kotła, które mogły ulec awarii.



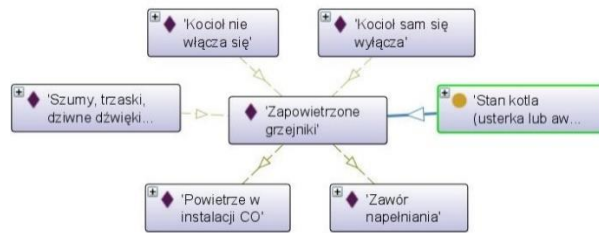
Rysunek B.29. Stan „Zatkane filtry, dysze lub elementy palnika” i odpowiadające mu części kotła, które mogły ulec awarii.



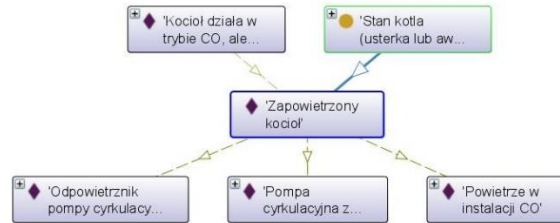
Rysunek B.30. Stan „Zadziałanie zabezpieczenia kotła” i odpowiadające mu części kotła, które mogły ulec awarii.



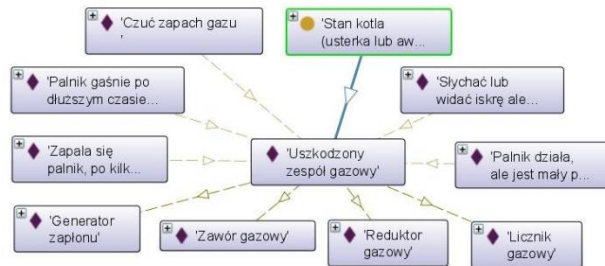
Rysunek B.31. Stan „Zanieczyszczony wtórny wymiennik ciepła” i odpowiadające mu części kotła, które mogły ulec awarii.



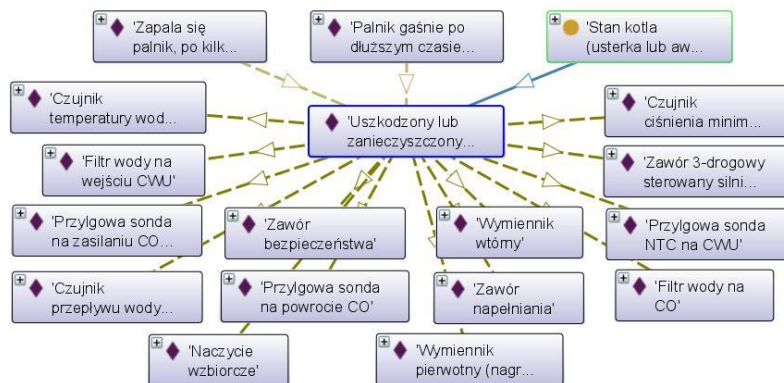
Rysunek B.32. Stan „Zapowietrzone grzejniki” i odpowiadające mu części kotła, które mogły ulec awarii.



Rysunek B.33. Stan „Zapowietrzony kocioł” i odpowiadające mu części kotła, które mogły ulec awarii.



Rysunek B.34. Stan „Uszkodzony zespół gazowy” i odpowiadające mu części kotła, które mogły ulec awarii.



Rysunek B.35. Stan „Uszkodzony lub zanieczyszczony zespół wodny” i odpowiadające mu części kotła, które mogły ulec awarii.