Politechnika Gdańska
Wydział Elektroniki, Telekomunikacji
i Informatyki

Rozprawa doktorska

Wyszukiwanie kontekstowe w pamięci semantycznej

Julian Szymański

promotor: prof. Włodzisław Duch
Uniwersytet Mikołaja Kopernika w Toruniu

Gdańsk, luty 2009

Podziękowania

Chciałbym gorąco podziękować wszystkim, którzy przyczynili się do tego, że praca ta powstała. W szczególności wdzięczny jestem:

prof. Włodzisławowi Duchowi, za opiekę nad pracą i czas, który mi poświęcił,

prof. Henrykowi Krawczykowi, za umożliwienie realizacji pracy w ramach badań Katedry Architektury Systemów Komputerowych wydziału Elektroniki Telekomunikacji i Informatyki Politechniki Gdańskiej,

całej mojej rodzinie, a zwłaszcza żonie Paulinie za cierpliwość i zrozumienie.

Spis treści

Motywacja, cel i zakres pracy

Istniejące obecnie systemy informatyczne używające interfejsów w języku naturalnym w dużej części oparte są na imitacji rozumienia przedmiotu dialogu, w którym identyfikowane są słowa kluczowe, do których dopasowywane są szablony interakcji. Częstym efektem takiego podejścia są odpowiedzi wymijające, bądź zupełnie nieadekwatne do przedmiotu dialogu. Typowym przykładem może być tu sławny program Eliza [99].

Jednym z kierunków badań nad możliwością użycia języka naturalnego do komunikacji z systemem informatycznym jest implementacja w nim struktur analogicznych funkcjonalnie do tych, jakie występują w mózgu człowieka i umożliwiają rozumienie języka. Jest to założenie paradygmatu neurokognitywnego [26], którego celem jest implementacja w maszynie procesu poznawczego. Badania nad sztuczną inteligencją kierujące się w stronę komputerowego modelowania aktywności poznawczej [95] mają na celu poprawę komunikacji człowieka z maszyną. Elementem procesu poznawczego człowieka jest *pamięć semantyczna*[1] odpowiedzialna za przechowywanie znaczenia pojęć i stanowiąca podstawę do rozumienia znaczenia symboli językowych. Dociekania nad tym rodzajem pamięci są przedmiotem niniejszej pracy.

By można było zrealizować repozytorium powiązań między pojęciami w systemie informatycznym, które jest podstawą do rozumienia ich znaczenia przez komputer, konieczne jest postawienie pytania o sposób organizacji wiedzy w jego biologicznym odpowiedniku i sposobu w jaki powstaje w nim znaczenie.

W pracy dokonany został przegląd psycholingwistycznych teorii dotyczących pamięci semantycznej oraz metod reprezentacji wiedzy, które użyte zostały do stworzenia systemu będącego implementacją podstawowych funkcjonalności pamięci semantycznej człowieka. Dla utworzonego semantycznego

[1] pamięć semantyczna – termin ten w pracy odnosi się do komputerowej implementacji repozytorium umożliwiającego składowanie powiązań pomiędzy pojęciami, szczegółowo opisanego w rozdziale 3. Motywacją do zrealizowania takiej bazy wiedzy są modele psycholingwistyczne pamięci semantycznej człowieka opisane w rozdziale 1, w którym termin ten odnosi się do elementu procesu poznawczego.

repozytorium pojęć opracowany został algorytm *wyszukiwania kontekstowego*[2] pozwalający wynajdywać obiekty składowane w pamięci semantycznej poprzez odwoływanie się do ich cech. Na bazie tego algorytmu zademonstrowane zostały elementarne kompetencje lingwistyczne komputerowej pamięci semantycznej przejawiające się w grze językowej.

Dane w pamięci semantycznej człowieka pokrywają cały język. Istniejące obecnie cyfrowe zasoby lingwistyczne nie zawierają pełnej informacji o strukturze powiązań między pojęciami. Dlatego badania ograniczone zostały do wybranej dziedziny, w obrębie której przeprowadzono eksperymenty: weryfikujące algorytm wyszukiwania opartego na kontekście i metody akwizycji wiedzy semantycznej. Stanowić one będą podstawę do zastosowania proponowanego podejścia na szerszą skalę.

Zweryfikowanie podejścia implementacji pamięci semantycznej i jej zastosowań w ograniczonej dziedzinie pozwala na szersze postawienie problemu, w którym zwycięstwo maszyny w grze językowej, w obrębie całego języka może być przyjęte jako cel pośredniego etapu w zdaniu przez komputer testu Turinga [91].

Użycie wyszukiwania kontekstowego powiązane z uczeniem na ograniczonym obszarze wiedzy stanowi przykład zastosowania zaproponowanej reprezentacji wiedzy. Jest ona relatywnie prosta, w stosunku od innych, dużo bardziej złożonych formalizmów (np.: logiki, ram), w których taki proces nie został zademonstrowany. Panuje powszechne przekonanie o konieczności wykorzystywania wyrafinowanych form reprezentacji wiedzy w analizie języka naturalnego; w szeregu zastosowań, takich jak wyszukiwanie kontekstowe czy gry słowne, prostsze formy reprezentacji wiedzy są bardziej przydatne i umożliwiają sprawne znajdywanie rozwiązań, czy też odpowiednich skojarzeń. Pozwala to przezwyciężyć problem „kruchości reprezentacji" (którą ramę należy użyć w danym kontekście?) podniesiony przez krytyków sztucznej inteligencji.

Zaprezentowana w pracy wektorowa reprezentacja pojęć może znaleźć szereg zastosowań, np.: do poprawy jakości klasyfikacji tekstów, czy organizacji dużych baz dokumentów lub grafiki. Wyszukiwanie kontekstowe przedstawia alternatywę dla tradycyjnego wyszukiwania informacji w tekście opartego na identyfikacji i dopasowywaniu słów kluczowych.

[2]algorytm wyszukiwania kontekstowego – zadanie wyszukiwania obiektów w pamięci semantycznej poprzez odwołanie się do cech je opisujących. Szczegóły implementacji przedstawione zostały w rozdziale 3.2.

Tezy pracy

- **Reprezentacja wiedzy semantycznej stanowi podstawę dla algorytmu wyszukiwania kontekstowego**

  W oparciu o psycholingwistyczne modele pamięci semantycznej i reprezentację wiedzy wykorzystującą powiązania określonego typu między pojęciami, można zrealizować wyszukiwanie obiektów poprzez określanie ich cech.

  Baza wiedzy organizująca pojęcia językowe, nazwana *pamięcią semantyczną*, jest komputerową implementacją podstawowych funkcjonalności, jakie posiada jej biologiczny pierwowzór.

  Numeryczna reprezentacja pojęć pozwala zrealizować funkcjonalności, które nie zostały zademonstrowane przez inne formalizmy.

- **Algorytm wyszukiwania może zostać użyty do weryfikacji i akwizycji wiedzy semantycznej**

  Algorytm wyszukiwania kontekstowego pozwala na ocenę jakości wiedzy pamięci semantycznej poprzez miarę opartą na liczbie poprawnie zrealizowanych wyszukań.

  Dzięki wprowadzeniu w procesie odnajdywania obiektów dodatkowej interakcji z użytkownikiem, algorytm wyszukiwania kontekstowego może również zostać użyty do wzbogacania wiedzy pamięci semantycznej.

Rozdział 1

Pamięć semantyczna

W rozdziale tym przedstawiono informacje dotyczące pamięci semantycznej, będącej elementem procesu poznawczego człowieka. Psycholingwistyczne modele tej pamięci posłużyły jako motywacja i wzór do przeprowadzenia jej komputerowej implementacji opisanej w rozdziale 3.

Pamięć jest mechanizmem istot żywych pozwalającym zdobywać doświadczenie poprzez gromadzenie informacji. Zjawisko to umożliwia podejmowanie decyzji dzięki zgromadzonym doświadczeniom. Pamięć stanowi odzwierciedlenie doznań lub wrażeń pochodzących zarówno z otoczenia, jak i z doświadczeń wewnętrznych. W ujęciu psychologii poznawczej, która traktuje pamięć jako proces przetwarzania informacji wyróżnić można trzy podstawowe funkcje pamięci: rejestrowanie – zapamiętywanie, przechowywanie, odtwarzanie [53].

Pamięć warunkuje uczenie. Pojęcia te są ze sobą silnie związane, tak że teoretycy pamięci często włączają proces uczenia się do definicji pamięci. Anderson [4] określa uczenie jako proces, poprzez który na skutek doświadczenia zachodzą względnie trwałe zmiany w potencjale zachowaniowym, natomiast pamięć definiuje jako względnie trwały zapis doświadczenia, które znajduje się u podłoża uczenia. Pamięć w tym ujęciu jest miejscem zapisu doświadczenia, uczenie natomiast jest adaptacją widoczną w zachowaniu.

Ze względu na różne aspekty wyróżnia się szereg rodzajów pamięci, jednak dotychczas nie powstał pełen model opisujący to zjawisko całościowo. Pojęcie pamięci doczekało się wielu ujęć, wynikających głównie z różnych metodologii jej badania i funkcji jakie pełni [17]. Dominującą w psychologii teorią pamięci jest model dokonujący podziału ze względu na czas przechowywania i pojemność. Wyodrębnia on trzy podstawowe typy pamięci: sensoryczną, krótkotrwałą i długoterminową.

- Pierwszy typ odpowiedzialny jest za wychwytywanie i strukturalizacje informacji pochodzącej z receptorów. Koncepcję tą wprowadził w 1960 r. G. Sperling [85] opisując ultrakrótki typ pamięci przechowujący informacje pochodzące z bodźców bezpośrednio po ich zadziałaniu przez czas, trwający dla wrażeń wzrokowych ok. 0,5 sekundy, a dla wrażeń słuchowych ok. 4 sekund. Pamięć ta tworzy ślady oddziaływania bodźców, które utrwalane są później w dalszych strukturach pamięciowych. Stanowi ona niejako pomost między podmiotem poznającym a światem doświadczenia, gdzie następuje redukcja ogromnej ilości informacji docierającej w percepcji, która organizowana jest w pierwotne struktury reprezentujące poznanie [102].

- Magazyn pamięci krótkotrwałej traktowany jest jako obszar, w którym przechowywane są informacje przekazywane z pamięci ultrakrótkotrwałej do pamięci długotrwałej. Pamięć ta charakteryzuje się ograniczoną pojemnością a także ograniczonym czasem przechowywania zawartych w niej danych. Szacuje się, że czas utrzymywania się w niej informacji to ok. 30-40 sekund. Z tym rodzajem pamięci związany jest zaprezentowany w 1956 r. przez G. Millera [57] eksperyment pokazujący, że można w niej przechować 7 ± 2 obiektów prostych (typu: litery, liczby, elementarne figury geometryczne, dźwięki). Pamięć ta jest swego rodzaju buforem pośredniczącym między zarejestrowanym doświadczeniem a pamięcią długotrwałą. Bufor ten odpowiedzialny jest za proces zapamiętywania – kodowania informacji w pamięci długotrwałej, jak również związany jest z procesami wydobywania, czyli odtwarzania (przypominania i rozpoznawania) przechowywanych na trwałe informacji. Pamięć krótkoterminowa pośredniczy w przechowywaniu nowych spostrzeżeń (percептów), jak również przetwarza obiekty doświadczenia wewnętrznego (nazywane receptami), wydobywane z pamięci długoterminowej. Przy pojawianiu się nowych informacji dane w pamięci krótkotrwałej zostają utracone lub też ulegają procesowi konsolidacji, w wyniku którego przechodzą do pamięci trwałej. Za proces konsolidacji pamięci odpowiedzialna jest głównie struktura mózgu zwana hipokampem.

- Magazyn pamięci długotrwałej charakteryzuje się bardzo dużą pojemnością, podobnie jak i czas przechowywania zawartych w niej treści. Pamięć długotrwała ma charakter syntetyczny. Zapamiętywanie w tym obszarze jest powolniejsze, wymaga skupienia uwagi i powtórzeń, chyba że zawiera bardzo ważne treści dla jednostki, wtedy przebiega szybciej. Dzieje się tak prawdopodobnie przez zwiększenie się plastyczności mózgu pod wpływem emocji.

Teorie dotyczące organizacji i struktury pamięci długotrwałej dokonują podziałów w oparciu o analizę sposobu kodowania, przetwarzania, funkcji oraz materiału, który jest zapamiętywany. Podział ten skutkuje różnymi modelami opisującymi sposoby przetwarzania i funkcjonalnościami realizowanymi przez pamięć długoterminową. W mózgu ssaków istnieją trzy komplementarne, przenikające się podsystemy pamięci długoterminowej, których centrum stanowią hipokamp, prążkowie oraz zespół jąder migdałowatych, sterujące odpowiednio pamięcią przestrzenną i deklaratywną, motoryczną oraz emocjonalną [94].

W zależności od charakteru organizowanej informacji używane są różnego typu reprezentacje, które przechowywane mogą być jako świadoma i łatwa do przekazywania, ale mogąca ulegać zatarciu pamięć deklaratywna. Przedmiotami tej pamięci są informacje w formie wiedzy o rzeczach. Wiedza składowana w tych strukturach jest typu „wiem że ...". Stanowi ona swego rodzaju osobistą encyklopedię zawierającą podstawowe informacje o świecie. Wiedza deklaratywna przechowywana jest w kodzie skojarzeniowym – zawiaduje nią obszar pamięci semantycznej związany z hipokampem i pewnymi obszarami kory mózgowej.

Innym typem jest pamięć proceduralna mająca charakter dotyczący działania. Jest to rodzaj pamięci działający poza świadomością, w którym przechowywane są umiejętności, sposoby postępowania, strategie. Wiedza proceduralna jest trudna do zwerbalizowania, ale bardzo trwała. Ten rodzaj pamięci składuje wiedzę typu „wiem jak ..." określającą procedury, reakcje, czynności jakie wykonujemy. Zapisywana jest w mapach sensomotorycznych i planach sekwencji ruchów, przede wszystkim w jądrach podstawy mózgu, prążkowiu, głównie w jądrze ogoniastym [52].

Trzecim typem pamięci występującym na tym poziomie podziału jest pamięć robocza. Jest to pamięć operacyjna łącząca pamięć proceduralną i deklaratywną, między którymi zapewnia komunikację poprzez kontrolę funkcjonowania połączeń. Sterowana jest przez korę przedczołową. Realizuje ona „system przerwań" umożliwiający dostęp do aktywacji pamięci długotrwałej i przechowuje tymczasowe rezultaty [75].

Poszukiwanie struktur w jakie organizowana jest informacja przez człowieka oraz obszarów w mózgu, w których określony rodzaj wiedzy jest składowany, były przedmiotem badań m.in. Endela Tulvinga. W *„Episodic and semantic memory"* [90] ze względu na sposoby przechowywania i mechanizmy wydobywania informacji w obrębie pamięci długoterminowej, dokonuje on rozróżnienia na pamięć semantyczną i epizodyczną.

Pamięć epizodyczna ma za swój przedmiot czyste doznania zmysłowe i odnosi się do przeżyć jakich człowiek doświadczył – odpowiedzialna jest za kodowanie, przechowywanie i przypominanie epizodów, których doświadczy-

ło się osobiście w określonym czasie i miejscu. Pamięć ta związana jest ze zdarzeniami oraz kontekstem ich wystąpienia, dzięki czemu pamiętamy przeszłe wydarzenia. Epizodyczna część zdarzeń życia jednostki zaklasyfikowana została jako podsystem pamięci epizodycznej, nazwany pamięcią autobiograficzną.

Pamięć semantyczna związana jest z systemem słownym i jego funkcją znaczeniowo-wyjaśniającą, dla której jest swoistą bazą wiedzy zawierającą informację o symbolach werbalnych, ich znaczeniu oraz wzajemnych powiązaniach opartych na określonych skojarzeniach. Postrzegana może być jako złożenie osobistego słownika i encyklopedii. W tym długoterminowym kontenerze pamięciowym, składowana jest osobista wiedza ogólna o świecie stanowiąca podstawę rozumienia języka. W odróżnieniu od pamięci epizodycznej, większość ludzi informacje w tym rodzaju pamięci posiada podobne, dzięki czemu możliwy jest proces komunikacji.

Pamięć semantyczna działa prawdopodobnie w ten sposób, że w mózgu tworzone są zbiory informacji o podobnym charakterze, opatrzone pewnymi etykietami, które są symbolami reprezentującymi znaczenie. Pomiędzy nimi, w miarę wzrostu wiedzy tworzone są asocjacje, nie będące jedynie prostym połączeniem, a zawierającym dodatkowo informacje pozwalające określić kontekst takiego powiązania. Pamięć semantyczna przypuszczalnie powstała na drodze ewolucji: w zamierzchłych czasach symbole w pamięci nie były wyrazami, lecz raczej posiadały reprezentacje piktograficzną. Wraz z upowszechnieniem się języka mówionego słowa stały się swego rodzaju indeksami w pamięci semantycznej [61].

Pamięci semantyczna i epizodyczna ściśle ze sobą współpracują: początkowo informacje dotyczące wspomnień i postrzeżeń związanych z wydarzeniem trzymane są w pamięci epizodycznej, będącym miejscem składowania doświadczenia. Następnie, w procesie konsolidacji i dalszego używania reprezentacje z tej pamięci przechodzą w reprezentacje sensu wydarzenia i w myśleniu funkcjonują już jako obiekty pamięci semantycznej, stanowiące tło pojęciowe dla kolejnych doświadczeń. Przechowywane w pamięci semantycznej informacje typu deklaratywnego są uogólnieniem, bądź też wynikiem zgromadzonych doświadczeń. Poprzez konsolidację pamięć semantyczna ulega ciągłemu procesowi reorganizacji zawartej w niej wiedzy.

Pamięć semantyczna warunkuje posługiwanie się językiem, którego rozumienie oparte jest na składowanych w niej typowanych skojarzeniach. Dzięki takiej organizacji pojęcia i treści postrzeżeń nie występują w izolacji od siebie a poprzez istniejące między nimi zależności powstaje w umyśle wyobrażenie będące pojęciową reprezentacją zjawisk zachodzących w rzeczywistości. Pamięć semantyczna umożliwia w zdarzeniach zmysłowych (np. postrzeżeniu przedmiotu) wyodrębnienie i nazwanie charakterystycznych struktur i związ-

ków oraz organizowanie ich na podstawie relacji, wzorców lub formuł. Proces ten możliwy jest poprzez wiązanie struktur percepcyjnych z symbolami językowymi, dzięki którym uwidoczniają się abstrakcyjne relacje.

Istnienie pamięci semantycznej potwierdzone zostało w badaniach wykorzystujących obrazowanie mózgu, wykazujące wzmożoną aktywność lewych płatów kory czołowej podczas przywoływania, poprzez skojarzenia, wspomnień. Drugim miejscem związanym z deklaratywną pamięcią długoterminową jest hipokamp. Potwierdzenie istnienia semantycznych struktur funkcjonalnych w mózgu odbyły się między innymi na przypadku pacjenta H.M. [79], któremu ze względu na padaczkę usunięty został ten fragment mózgu. Po przeprowadzonym zabiegu wykazywał on amnezję pamięci deklaratywnej – nie mógł nabywać wiedzy ani semantycznej, ani epizodycznej. Natomiast uwaga i inne zdolności poznawcze pozostały nietknięte.

Czasy dostępu do informacji składowanej w pamięci deklaratywnej zależne są od kontekstu oraz częstotliwości dostępu do niej. Parametr czasu reakcji semantycznej jest podstawą większości badań przeprowadzanych nad tym rodzajem pamięci.

Wprowadzenie pamięci semantycznej do badań lingwistycznych umożliwia ujęcie języka jako systemu wyrażeń wyposażonych w znaczenie. Konieczność takiego podejścia wynika z faktu, że słowa i znaczenia nie są w umyśle jednoznacznie powiązane. Rozdźwięk między słowami i znaczeniami unaocznia się w zjawisku wieloznaczeniowości (polisemii), jak również w istnieniu wielu słów odpowiadających jednemu znaczeniu (synonimia). Brak wzajemnego idealnego odwzorowania znaczenia i słowa skutkuje wieloma trudnościami podczas automatycznego przetwarzania języka naturalnego np. koniecznością ujednoznaczniania pojęć w przetwarzanym tekście [89]. Problem ten jest jedną z przyczyn ogromnej trudności w automatycznej realizacji przekładów języków, podczas których unaocznia się fakt, że każdy język zawiera słowa, które używane są jedynie w określonym kontekście i których znaczenie jest specyficzne tylko dla niego. Wyciągnąć stąd można wniosek, że nie ma uniwersalnych znaczeń a powstają one pod wpływem doświadczenia i są niejako wypadkową słów funkcjonujących w określonym języku oraz osobistych doświadczeń użytkownika języka.

W kontekście myśli o języku, jako systemie znaków mających przekazywać zamierzone, abstrakcyjne informacje nieodłącznie towarzyszy lingwistyce Semantyka (grec. $\sigma\eta\mu\alpha\nu\tau\iota\kappa\acute{o}\varsigma$ – istota znaczenia), poruszająca problem istoty znaczenia i relacji między językiem a rzeczywistością. W ogólnym ujęciu pytanie o znaczenie jest pytaniem o relacje między słowem a przedmiotem oznaczonym.

W leksykografii trudności z formalnym zdefiniowaniem znaczenia sprowadziły przypisywanie znaczeń pojęć do równoznacznych im wyrażeń bardziej

analitycznych, będących opisem znaczenia np. uproszczona eksplikacja znaczenia wyrazu radość (*definiendum*) w języku polskim może brzmieć (*definiens*) „uczucie zadowolenia i satysfakcji przejawiające się śmiechem, optymistycznym spojrzeniem na rzeczywistość czy poczuciem beztroski" [33]. Treść charakterystyczna dla danego pojęcia nazywana jest konotacją, czyli zbiorem cech właściwych desygnatom[1].

Używana w podejściach leksykograficznych definicja jest tylko pewnym zawężeniem znaczenia, które w kognitywistyce jest badane szerzej, jako część procesu poznawczego. Analiza sposobu powstawania znaczenia w umyśle ma bardzo praktyczny sens. Znając sposób formułowania się znaczenia potencjalnie jesteśmy w stanie zbudować jego komputerowy model stanowiący podstawę rozumienia. Proces powstawania znaczenia w umyśle nie daje się jednak w tak prosty sposób wyjaśnić i zrealizować, jak ma to miejsce w tradycyjnym podejściu słownikowym. Użycie definicji pozwala jedynie dookreślić pojęcie w kontekście innych znaczeń i zakłada wiedzę o znaczeniach występujących w definicji. Klasyczna definicja składa się z dwóch części: rodzaju najbliższego, czyli nazwy podającej klasę, w której zawiera się zakres definiowanego wyrazu oraz różnicy gatunkowej wskazującej na to, co wyróżnia definiowane pojęcie spośród elementów klasy nadrzędnej. W kognitywnym ujęciu semantyki wprowadza się pojęcie słownika umysłowego [61], w którym składowane są pojęcia o obiektach świata i relacjach między nimi. Znaczenie jest tu wynikiem zachodzących powiązań miedzy pojęciami.

Wytworzony przez człowieka system znaków przekazujących abstrakcyjne informacje pociąga za sobą pytanie o ich naturę, a więc o to, co jest znaczeniem znaku. Semantyka stara się odpowiedzieć na pytanie „co znaczy" zadane słowo i jak to się dzieje, że potrafimy określić pewne obiekty mianem np. krzeseł, mimo znaczących różnic w ich konstrukcji [61].

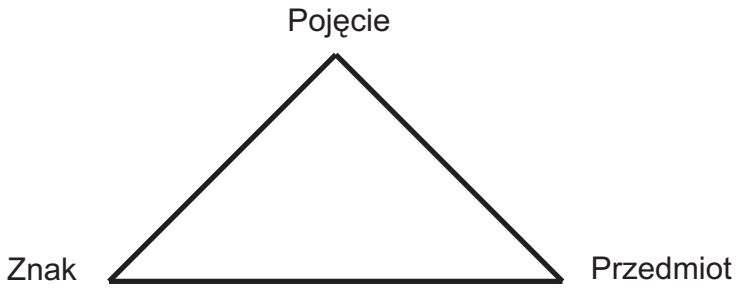
By system informatyczny mógł posiadać kompetencje językowe, musi posiadać możliwość rozumienia pojęć. Pytanie o istotę rozumienia pociąga za sobą konieczność określenia jego istoty, implikującą stawianie generalnych pytań teoretycznych o naturę znaczenia.

Wersję obiektywistyczną[2] konceptualizmu[3] znaczenia opisuje trójkąt oznaczania Ogdena i Richardsa [64]. To klasyczne ujęcie natury symboli lingwistycznych przedstawione zostało na rysunku 1.1. Wyróżnione są na nim trzy elementy konstytuujące ten model:

[1] łac. *designatus* – oznaczony, każdy przedmiot oznaczany przez daną nazwę

[2] istnieje fundamentalna rzeczywistość, niezależna od naszego poznania i świadomości

[3] nazwy ogólne nie posiadają realnego bytu, lecz są jedynie wytworami ludzkiego umysłu



Rysunek 1.1: Trójkąt semantyczny

znak – dowolny element zdolny do reprezentowania czegoś innego. Słowa są tu znakami szczególnego rodzaju – symbolami. Większość symboli nie wiąże się w żaden naturalny sposób z przedmiotami lub zjawiskami, do których opisu służą. Słowa przyjmują znaczenia w zależności od kontekstu, w którym występują. Pogląd ten funkcjonuje pod nazwą „przesądu prawdziwego znaczenia" – błędnego przekonania, że słowa posiadają swoją precyzyjną definicję.

pojęcie – jest mentalną reprezentacją, myślą, na którą wskazuje znak. Idea będąca wyobrażeniem związanym z obiektem doświadczenia – znaczenie powstaje po stronie podmiotu poznającego i jest przedmiotem jego mentalnej aktywności.

przedmiot – referent, jest odniesieniem do obiektu naturalnego istniejącego w rzeczywistości, mogącego być przedmiotem postrzeżenia.

Model ten jako znaczenie umieszcza pojęcie rozumiane jako typ myśli, odróżnione od przedmiotu oznaczonego (referenta): słowa są tu arbitralnymi znakami, które same w sobie nic nie znaczą a przyjmują znaczenie poprzez pojęcia, w zależności od przedmiotu doświadczenia. Trójkąt w intencji Ogdena i Richardsa miał przedstawić funkcjonowanie sensu, symbolu i obiektu w języku. Istotą modelu jest wskazanie pewnej wewnętrznej struktury systemu językowego, w którym pojęcie (wyobrażenie) poprzez znak wiąże się z obiektem świata rzeczywistego (przedmiot). Wartym zaznaczenia jest fakt oderwania w tym modelu znaczenia od gramatyki.

Przyjęcie określonego modelu tłumaczącego, czym jest znaczenie, będzie miało zasadniczy wpływ na realizację rozumienia w komputerze. Znaczenie i rozumienie są ze sobą ściśle powiązane, ponieważ to, jak zachodzi proces powstawania znaczenia wpływa na to jak zachodzi rozumienie. Pytania o istotę znaczenia, stawiane są od wielu lat przez logików, filozofów i językoznawców. Wskazują oni, że pytanie o związek zachodzący miedzy słowem a znaczeniem

jest kluczowe z punktu widzenia filozofii języka, i niezbędne do realizacji systemów informatycznych mających posługiwać się językiem naturalnym [1].

W lingwistyce, w najogólniejszym ujęciu, odrywającym znaczenie od gramatyki, sens formułuje się poprzez zbiór cech semantycznych związanych z danym pojęciem. Cechy semantyczne są tu najmniejszym, niepodzielnym nośnikiem znaczenia. Za Brzozowską [11] mogą zostać one pogrupowane ze względu na ich rangę w odniesieniu do pojęcia w tzw. *strefy konotacyjne*:

- Kryterialne cechy semantyczne stanowią zbiór cech pozwalających odróżnić grupę przedmiotów, do których stosuje się dana nazwa. Identyfikacja danego obiektu, jako wchodzącego w zakres znaczenia danej nazwy odbywa się na podstawie pewnej liczby dostrzegalnych cech wspólnych. Do przeprowadzenia rozróżnienia nie są wykorzystywane wszystkie cechy, ponieważ klasyfikacja obiektu, jako desygnatu określonej nazwy jest formą przybliżenia opartego na podobieństwie do danej kategorii semantycznej.

- Konotacje semantyczne. Grupa ta zawiera asocjacje, stereotypy powstałe poprzez doświadczenie w obrębie określonego kręgu kulturowego. Na podstawie tych cech nie rozpoznajemy przedmiotu, ale możemy dokonać rozróżnień i uzyskiwać szerszy kontekst znaczeniowy.

- Cechy indywidualne obiektów. Nadawane są one przez indywidualnych użytkowników języka, będące wynikiem subiektywnej, antropocentrycznej interpretacji rzeczywistości.

Trzy sfery konotacyjne tworzą zbiór cech semantycznych znaczenia danego znaku językowego, będących jego definicją kognitywną. W ujęciu tym znaczenie odnosi się do modeli struktur poznawczych a nie bezpośrednio do świata. Treścią definicji jest zespół cech charakterystycznych, koniecznych i wystarczających.

Alternatywną dla cech semantycznych propozycją ujęcia znaczenia jest teoria radialnej struktury znaczenia George Lakoffa [45]. Opiera się ona na prototypach będących najbardziej typowymi przedstawicielami danej kategorii semantycznej, do której klasyfikacja odbywa się poprzez przyjętą miarę podobieństwa. Obiekty uznawane za typowe przykłady danej kategorii tworzą jej prototyp. Pozostałe obiekty tworzą podkategorie peryferyjne, mogące być kategoriami centralnymi dla znaczeń podrzędnych kategorii głównej.

Obie te teorie znaczenia mają swoje odzwierciedlenie w modelach pamięci semantycznej (paragraf 1.1) co obrazuje, jak ujęcie zagadnienia rozumienia wpływa na kształt teorii psycholingwistycznych.

Rozumienie języka jest procesem złożonym. Przetwarzanie komunikatu nadanego z użyciem języka naturalnego odbywa się na kilku płaszczyznach:

zarówno semantycznej, składniowej, jak i emocjonalnej, i tak zarówno użyte wyrazy, jak ich szyk nadają znaczenie wypowiedzi. Na poziomie fonologicznym na rozumienie ma również wpływ intonacja. W zależności od położenia akcentów wyrażenie nabiera znaczenia pytania, oznajmienia bądź wykrzyknika. Do ujednoznacznienia wieloznacznych słów przekazywanych treści wykorzystywane są zarówno informacje leksykalne, syntaktyczne, jak i kontekstowe. Pozwalają one na antycypację np.: niewyraźnie usłyszanych sygnałów lub też na dopełnienie końca zdania [61].

Pamięć semantyczna ma wpływ na jeden z aspektów rozumienia – jako słownik umysłowy nadaje kontekst znaczeniowy pojęciom komunikatu wynikający z wzajemnych asocjacji słów. Odbywa się to poprzez pobudzanie skojarzonych ze sobą elementów będących aktualnym przedmiotem doświadczania, w wyniku czego w pamięci roboczej powstaje pojęciowy obraz postrzeganej wypowiedzi. Pojęcia nie powiązane z kontekstem nie zostają aktywowane lub zostają wytłumione. Zjawisko to znane jest pod nazwą torowania semantycznego polegającego na tym, że jeśli w zdaniu pojawia się jakiś wyraz wieloznaczny, to znaczenie jego będzie określone dzięki pojęciom występującym w jego otoczeniu. Rozpoznanie jednego pojęcia ze zbioru słów o podobnym znaczeniu wytłumia aktywację pozostałych znaczeń. Zjawisko to może zostać użyte do ujednoznaczniania pojęć występujących w tekście [89].

Zasadniczym elementem procesu poznawczego, który jest konieczny by mogło zajść rozumienie wyrażeń języka naturalnego jest przestrzeń odwzorowań doświadczenia – miejsce, w którym percepty nabierają znaczenia w odniesieniu do innych znanych pojęć. Powstaje tu zdarzenie umysłowe określane mianem rozumienia słowa, będące wynikiem rezonansu między stanami zapamiętanymi a stanem bieżącym – doświadczeniem. Pamięć semantyczna, jako element procesu poznawczego człowieka aktywnie uczestniczy w tym zdarzeniu. Stanowi ona oparte na nazwanych skojarzeniach z obiektem postrzeżenia tło pojęciowe, na którym powstają wrażenia pozwalające rozumieć język.

1.1 Psycholingwistyczne modele pamięci semantycznej

Sposób, w jaki pojęcia są reprezentowane i zorganizowane w systemie przetwarzania doświadczenia u człowieka, jest przedmiotem badań psycholingwistyki. Do tej pory nie został wypracowany jeden spójny model opisujący to zagadnienie całościowo, jednak istnieje szereg teorii odnośnie sposobu organizacji pojęć w umyśle, których szczegóły związane są z budową i wyko-

rzystaniem umysłowego słownika [61]. Modele te weryfikowane są w testach psychologicznych (paragraf 1.2).

W tym rozdziale zaprezentowane zostały główne teorie opisujące teoretyczne modele pamięci semantycznej. Większość z nich dotyczy cech desygnatów i podstawowych składników semantycznych. Najbardziej typowym ujęciem pamięci semantycznej jest sieć złożona z węzłów (słów, pojęć) i połączeń między nimi w postaci prostych asocjacji, bądź typowanych relacji. Pojęcia reprezentowane poprzez węzły tworzą sieć nawzajem połączonych obiektów językowych. Odległości między nimi odzwierciedlają podobieństwo semantyczne a typy powiązań opisują zależności między nimi, takie jak relacje przynależności do klasy nadrzędnej, relacje o charakterze implikacyjnym itp.

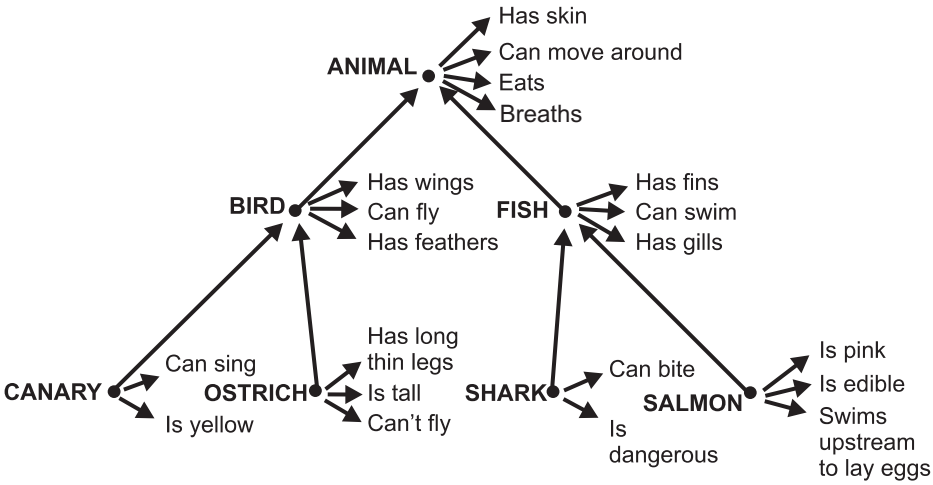
Podejście oparte na sieci powiązań między słowami wprowadza znaczenie jako wynik interakcji z innymi węzłami. Znaczenie określonego pojęcia wynika z jego do innych pojęć. W umyśle powiązania mają nie tylko naturę czysto skojarzeniową. Asocjacje między pojęciami w pamięci semantycznej są również jakościowe, jak i mają inklinacje celowości. Wartość semantyczna – znaczenie jest wypadkową wszystkich tych czynników, co zostało ujęte w opracowanej reprezentacji wiedzy (paragraf 3.1) użytej do implementacji pamięci semantycznej.

1.1.1 Model hierarchiczny

Podstawowym modelem pamięci semantycznej jest opracowana w 1968 roku koncepcja Collinsa i Quilliana [20]. Organizuje ona pojęcia w sieć hierarchiczną za pomocą predefiniowanej relacji *is_a* realizującej tu zarówno instancjonowanie, jak i przynależność do nadklasy[4]. Dla każdego węzła wyodrębnione są cechy, z którymi się on wiąże za pomocą relacji *has_a* i *can_a*. W modelu hierarchicznym relacja *is_a* organizuje pojęcia w taksonomię. Podejście takie zapewnia tzw. ekonomię poznawczą: cechy przypisane węzłom nadrzędnym nie powtarzają się na poziomie podrzędnym. Fragment taksonomii kategorii naturalnych ujętej w modelu hierarchicznym przedstawiono na rysunku 1.2.

Model hierarchiczny pozwala na odnajdywanie połączeń między węzłami i przez to udzielanie odpowiedzi tak/nie odnośnie cech posiadanych przez obiekty, które zapisane są w takiej strukturze. Założenia na bazie których powstał ten model pamięci semantycznej zakładają, że czas odpowiedzi (RT –

[4]Połączenie w jedną relację matematycznych operacji należenia $\in$ oraz zawierania (inkluzji) $\subset$ ukrywane pod nazwą is_a jest uproszczeniem, które może prowadzić do sprzeczności [95]

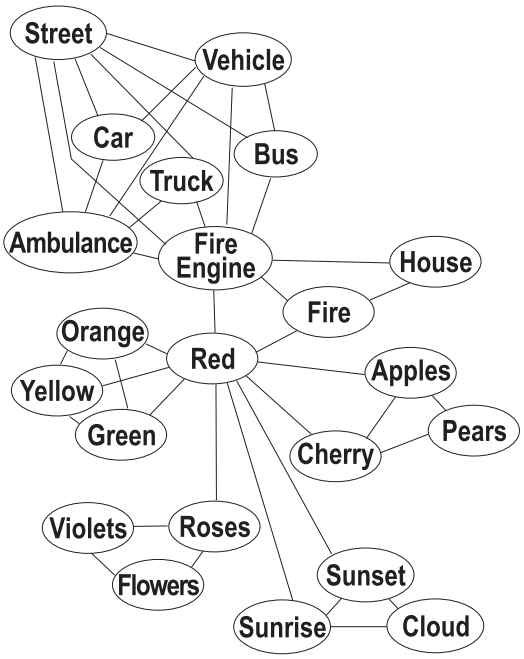


Rysunek 1.2: Model hierarchiczny

response time) jest silniej związany z powiązaniami opisującymi cechy obiektów, niż z cechami wynikającymi z organizacji hierarchicznej. Czas odpowiedzi, dla cech typowych, dla zadanego pojęcia będzie krótszy niż uzyskanie odpowiedzi poprzez analizę powiązań w relacjach nadrzędnych. Uzyskane wyniki badań eksperymentalnych, polegające na badaniu prawdziwości zdań wskazują na korelację tej teorii z jej weryfikacją empiryczną, a odchylenia tłumaczone są powstawaniem dodatkowych szlaków asocjacyjnych w umyśle między częściej używanymi pojęciami. Dzięki takiej redundancji wzrasta efektywność przetwarzania informacji.

Badanie efektu odległości semantycznej opiera się na założeniu, że czas weryfikacji prawdziwości zdania zawierającego dwa pojęcia odpowiada odległości w hierarchicznym modelu między dwoma badanymi termami. Weryfikacja teorii poprzez efekt typowości lub podobieństwa w przykładowych zdaniach „kanarek jest ptakiem", „struś jest ssakiem" ma na celu wytłumaczenie krótszego czasu reakcji (odpowiedzi) dla pierwszego pytania. Wnioski wynikłe z badań przeprowadzonych w testach weryfikacji zdań wskazują na istnienie związku między hierarchicznymi strukturami modelu, a sposobem organizacji danych w umyśle. Opisywana przez model organizacja danych nie potwierdzona została jej fizyczną realizacją w mózgu.

Krytycy tego podejścia zgłaszają wątpliwości związane przede wszystkim z brakiem odniesienia do problemu wagi poszczególnych cech dla węzła i reprezentatywności egzemplarzy dla kategorii oraz braku efektu typowości, np. reakcje na pojęcia szpak i kura powinny się znacząco różnić. Odrębnym problemem związanym z przyjęciem tej teorii jest kwestia ograniczenia liczby typów relacji między węzłami.



Rysunek 1.3: Model rozprzestrzeniającej się aktywacji

Godnym wspomnienia jest fakt istnienia niejawnego założenia przyjętego przez twórców modelu hierarchicznego o uporządkowaniu pamięci ludzkiej. Nie ma faktów przemawiających za tym, że dane w pamięci są właśnie w ten sposób organizowane, aczkolwiek potwierdzone w eksperymentach zjawiska RT wskazują na pewną odpowiedniość między takim modelem a tym co zachodzi w mózgu. Model dobrze reprezentuje kategorie pojęć odnoszących się do obiektów naturalnych, jednak jego aplikacja do kategorii abstrakcyjnych nastręcza wielu kłopotów.

1.1.2 Model rozprzestrzeniającej się aktywacji

Struktura, którą zaproponowali w modelu rozprzestrzeniającej się aktywacji Collins i Loftus [19] przedstawia pamięć semantyczną, jako sieć powiązań między pojęciami, których znaczenie związane jest z ich wzajemnym położeniem. Znaczenie jest w tym modelu ekstrapolowane ze skupienia wystąpień skojarzeń wokół określonego słowa. Sieć wzajemnie aktywujących się pojęć przedstawiono na rysunku 1.3.

Powiązanie między węzłami określa bliskość semantyczną pojęć. W sieci możliwe jest wprowadzenie różnych odległości między węzłami, dzięki czemu węzły reprezentujące cechy najbardziej charakterystyczne dla danej kategorii są silnie połączone z węzłem ją reprezentującym. W modelu tym nacisk

położony jest na założenie tak zwanej rozproszonej aktywacji. Polega ona na przekazywaniu impulsu aktywującego wybrane elementy sieci do kolejnych, powiązanych z nimi elementów. Wywołanie określonej nazwy pobudza węzły im odpowiadające. Następnie pobudzenie to zostaje przekazane do węzłów sąsiadujących. Jeśli aktywacja jest wystarczająco silna, zostaje przekazana jeszcze dalej. Po przekroczeniu progu wzbudzenia (aktywacji) w jakimś punkcie sieci, aktywacja ta rozprzestrzenia się na sąsiednie okolice, czyli najszybciej aktywowane są elementy najbliżej leżące. Jeśli aktywacja pochodzi z kilku źródeł, zostaje ona wzmocniona. Dodatkowo, częściej używane szlaki skojarzeń między węzłami zyskują większą siłę połączenia, przez co zapewniają łatwiejsze przekazywanie pobudzenia. Poprzez wzmacnianie połączeń między węzłami silnie skojarzonymi uzyskany został tzw. efekt typowości – podobieństwa pojęć. Aktywacja zależy od odległości (będącej siłą i liczbą pośrednich połączeń) między pojęciami. Im odległość jest dalsza, tym aktywacja jest słabsza. Duża liczba nieistotnych ścieżek osłabia aktywację, a im bliżej znajdują się sąsiednie węzły, tym silniejsze będzie ich pobudzenie.

Model ten, lepiej niż hierarchiczny tłumaczy pewne nietaksonomiczne podobieństwa oraz efekt podobieństwa semantycznego opartego na asocjacjach. Zjawisko to widoczne jest w niektórych przypadkach, zwłaszcza przy ocenie podobieństwa słów, gdzie odległość semantyczna miedzy pojęciami nie powoduje efektu zwiększania czasu odpowiedzi dla nieprawdziwych twierdzeń np.: „wszystkie owoce są warzywami", „owoce są kwiatami".

Wprowadzenie znaczenia słowa poprzez skojarzenia z nim związane, pozwala wprowadzić więcej niż jedną drogę wytłumaczenia podobieństwa semantycznego. Podstawowym problemem w tej teorii jest fakt, że definicja oparta tylko na skojarzeniach z właściwościami nie będzie adekwatna dopóki nie będzie obejmować wszystkich aspektów *definiendum*. Innym słabym punktem wskazywanym tej teorii jest brak wewnętrznej struktury oraz brak typów relacji między powiązaniami, jak również nie zachowywanie kognitywnej ekonomii.

W swoje pierwotnej wersji, teoria rozprzestrzeniającej się aktywacji nie uwzględnia również możliwości istnienia powiązań hamujących, które wytłumiają pobudzenia znaczeń związanych z określonym słowem, ale są niewłaściwe w określonym kontekście (innymi pojęciami aktywującymi sieć). Istnienie powiązań hamujących pozwala jedynie rozchodzić się impulsom wzajemnie wzmacniającym, których podobieństwo określone jest znaczeniem wypowiedzi aktywującej sieć a nie tylko wynika ze sposobu organizacji powiązań między pojęciami. Rozszerzenie koncepcji rozproszonej aktywacji o powiązania hamujące użyte zostało do ujednoznaczniania pojęć medycznych w *grafach spójnych pojęć* [54], dzięki którym słowa będące reprezentacją fonologiczną pojęć aktywowane są właściwe we właściwych znaczeniach. Aktywacja ta

odbywa się na zasadzie „zwycięzca bierze wszystko", dzięki czemu słowa aktywują się we właściwych sensach, a znaczenia nie adekwatne do zadanego kontekstu są wytłumiane. Podejście to użyte zostało również do ujednoznaczniania słów na podstawie słownika WordNet [89].

W badaniach empirycznych znalezione zostały potwierdzenia tezy, że siła aktywacji węzła jest zmniejszającą się funkcją liczby powiązań, jakie ta aktywacja przeszła [98]. W mózgu człowieka dla podstawowych skojarzeń aktywacja rozchodzi się szybko, w czasie rzędu 30 - 100 ms, co widoczne jest w EEG . Dla pojęć odległych czas ten jest wyraźnie większy ($>$ 700 ms) i potwierdzone zostało w testach z torowaniem skojarzeniowym dla par pojęć [56].

1.1.3 Teoria cech semantycznych

Teoria cech semantycznych opiera się na definiowaniu pojęcia poprzez zbiór jej właściwości. Charakteryzuje ją dużo prostsza struktura od modeli zaprezentowanych wcześniej, oparta na kolekcji list. Każde pojęcie jest zbiorem cech (podobnie jak właściwości w modelach sieciowych). Zbiór cech w modelu tym jest reprezentantem znaczenia obiektu. Wyróżnione zostały dwa rodzaje cech:

- cechy charakterystyczne (częste) – będące cechami opisowymi obiektu, powszechnymi, lecz nie będące niezbędnymi do zdefiniowania znaczenia np. drozd (lata, gnieździ się na drzewach).

- cechy definicyjne (konieczne) – istotne dla znaczenia danego pojęcia, np. drozd (pióra, czerwone podbrzusze).

Przesłankami do opracowania modelu było uwzględnianie cech wspólnych i różnicujących pojęcia przy podejmowaniu decyzji o ich wzajemnym podobieństwie. Według autorów [83] porównywanie takie odbywa się w dwóch fazach: szybkiej ocenie ogólnych cech, w której nastąpić ma szybkie rozstrzygnięcie. Jeśli w tej fazie nie udało się dokonać oceny podobieństwa, następuje przejście do drugiego etapu, wolniejszego, bardziej szczegółowego porównania listy cech definicyjnych. Np. odpowiedź na pytane „czy kanarek jest ptakiem?" daje silną odpowiedź twierdzącą opartą na podobieństwie cech charakterystycznych, przez co pozwala przeprowadzać szybko weryfikację zdania. Pojęcie *pingwin* posiada pewne pokrycie cech charakterystycznych ptaka. W celu oceny prawdziwości zdania „pingwin jest ptakiem", w związku z istnieniem tylko kilku cech charakterystycznych następuje przejście do fazy drugiej – porównania cech definicyjnych.

Model cech semantycznych dobrze tłumaczy efekt typowości: dużego podobieństwa między obiektami np. marchewka jest warzywem. Proces decyzyjny zachodzi szybciej, gdy porównywany jest typowy przedstawiciel kategorii, innej niż jego przypadek szczególny. Potwierdzeniem tych obserwacji ma być wspomniany dwufazowy sposób porównywania.

Do wad tego podejścia zaliczyć należy dużą złożoność pamięciową wynikającą z rezygnacji z ekonomii kognitywnej.

Badania doświadczalne wykazały jednak pewien efekt wskazujący na luki teorii cech semantycznych: nazwany został on efektem rozmiaru kategorii (*category size effect*) [34]. Rozpatrując zdania typu „pudel jest psem", „wiewiórka jest ssakiem" wykazano, że weryfikacja zdań w wypadku, gdy obiekt jest elementem małej kategorii przebiega szybciej, pomimo że małe kategorie zawierają więcej cech. W sytuacji takiej model powinien przeprowadzać więcej wyszukiwań opartych na drugiej fazie, co wiąże się z dodatkowym czasem.

1.1.4 Organizacja poprzez prototypy

Teoria prototypów jest modelem stopniowej kategoryzacji pojęć, gdzie wybrane elementy zbioru są bardziej centralne niż inne. Dla przykładu, typowym egzemplarzem semantycznej kategorii *meble* jest *krzesło*, które jest bardziej typowe niż *lampa*.

Wywodząca się z nurtu badań nad kategoriami naturalnymi teoria prototypów jest odejściem od tradycyjnych warunków koniecznych i wystarczających, które jest podejściem opartym na teorii zbiorów. W odróżnieniu od teorii opartej na cechach semantycznych, gdzie np. ptak jest definiowany poprzez zbiór jego cech (np.: dziób, lata, pióra), teoria prototypów rozważa kategorię taką jak *ptak* jako złożenie różnych obiektów, które mają nierówny status – poszczególne egzemplarze kategorii mogą przynależeć do niej w różnym stopniu np.: drozd jest bardziej prototypem ptaka niż np. pingwin, jabłko jest bardziej owocem niż figa, przez co jest lepszym reprezentantem tej kategorii.

Model organizacji pojęć poprzez prototypy zdefiniowany został przez Eleanor Rosch [74]. W teorii tej zakłada się istnienie archetypów będących reprezentacjami klas semantycznych. Przynależność obiektów do nich określana jest poprzez podobieństwo realizowane na różnych poziomach przetwarzania. Prototypy powstają jako najbardziej istotne pobudzenia biorące udział w formułowaniu się wyobrażenia, pochodzącego od najbliżej skojarzonych z kategorią pojęć. Prototypem staje najbardziej centralny członek grupy – dany egzemplarz będący najlepszym reprezentantem kategorii.

Badania nad prototypową kategoryzacją pojęć wskazują na istnienie dwóch

podstawowych zasad je formułujących: ekonomii kognitywnej (redukującej nieskończoną liczbę różnic pomiędzy pobudzeniami pojęć do proporcji mogących być ujmowanymi poznawczo) i spostrzeżeniu, iż przy analizie współwystępowania słów określone pojęcia łączą się z innymi, np. skrzydła częściej występują w otoczeniu piór niż futra.

W obrębie teorii prototypów dokonane zostały podziały na poziomy kategorii wertykalnych i horyzontalnych:

- Wymiar wertykalny skupia się na zawieraniu się kategorii wewnątrz siebie, dzięki którym pojęcia coli, pies, ssak się różnią.
- Wymiar horyzontalny skupia się na segmentacji kategorii na tym samym poziomie wertykalnym i jest wymiarem, w którym można wprowadzać rozróżnienia między pojęciami typu samochód, pies, krzesło.

Rosch wskazuje na fakt, że użycie prototypów zawierających najbardziej reprezentatywne atrybuty wewnątrz kategorii zwiększyłoby elastyczność użycia modelu i poprawiło rozróżnialność kategorii w obrębie podziału horyzontalnego. Funkcjonowanie prototypów w procesie poznawczym człowieka zostało potwierdzone w badaniach nad tworzeniem sztucznych kategorii pojęciowych, które to badania przeprowadzili Posner i Keele [67].

Problemem w podejściu opartym na prototypach jest trudność z jednoznacznym stwierdzeniem, które desygnaty należą w języku do centralnej podkategorii zakresu nazwy. Zwłaszcza, gdy część z cech pochodzi z abstrahowania i uogólniania, co często jest wynikiem subiektywnych odczuć i doświadczeń podmiotu poznającego.

Osobną kwestią związaną z tym modelem jest sposób wyznaczania prototypów, jako że niektóre pojęcia, egzemplarze kategorii są bardziej reprezentatywne np.: drozd, wróbel są prototypem kategorii ptaki bardziej niż np. struś. W związku z tym powstaje zagadnienie, na jakiej zasadzie określone obiekty stają się reprezentantami kategorii [6]. Szacuje się, że w każdej kategorii jest około 5 prototypowych przykładów mających więcej cech wspólnych niż peryferyjne przykłady. Istnienie obiektów występujących na obrzeżach jednostek semantycznych, prowadzi do rozmycia granic miedzy kategoriami.

1.1.5 Model koneksyjny

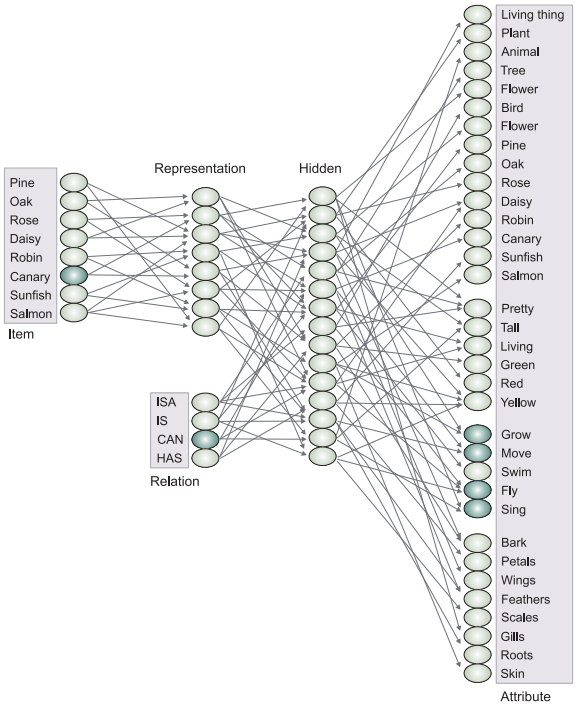
Alternatywnym podejściem do przedstawionych wcześniej ujęć znaczenia poprzez powiązania między pojęciami jest podejście oparte na przetwarzaniu równoległym i rozproszonym (*Parallel Distributed Processing – PDP*). W podejściu tym podstawą do reprezentacji pojęcia są węzły nazywane umownie neuronami (w istocie reprezentujące grupy wielu neuronów) a nośnikiem

znaczenia połączenia między tymi jednostkami. Podejście to tworzy model organizacji danych oparty o doświadczenia związane z przetwarzaniem neuronowym. Zakłada, że przetwarzanie informacji odbywa się przez interakcję dużej liczby prostych komponentów, posiadających połączenia hamujące i pobudzające do innych elementów.

Reprezentacja pojęć realizowana jest poprzez sieć złożoną z elementarnych części będących matematycznymi modelami neuronów. Podejście takie ma zbliżać model pamięci semantycznej do jego biologicznego pierwowzoru. Struktura utworzona przez sieć ważonych połączeń, których zadaniem jest wzmacnianie bądź osłabianie wybranych pobudzeń realizuje sposób przetwarzania zachodzący w mózgu: równoległy i rozproszony.

Podstawą, na której oparty został model, jest realizacja empirycznych spostrzeżeń dotyczących działania pamięci, która jest elastyczna i potrafi sobie poradzić z informacją niekompletną, bądź nawet niewłaściwą a adresowanie w niej oparte jest na kontekście. Rozproszone reprezentacje pozwalają na automatyczne generalizacje znaczenia, w którym podobne wzorce produkują podobne odpowiedzi. Umożliwiają one użycie różnych typów atrybutów np. koloru, w celu uzyskania dostępu do informacji zapisanych w strukturach pamięciowych. Atrybuty różnią się efektywnością określania pojęć np. pingwin będzie lepiej charakteryzowany cechą „nie lata" niż „ptak".

Wydobywanie informacji odbywa się poprzez pobudzenia jednostki będącej reprezentantem interesującego nas pojęcia a przez to aktywowanie jej własności w formie innych węzłów sieci. Pamięć informacji nie jest zlokalizowana w określonym miejscu, czy w konkretnym węźle lub powiązaniu, tak jak miało to miejsce we wcześniej przedstawionych podejściach. Zapamiętane wzorce są kodowane jako szablony aktywacji pomiędzy elementarnymi jednostkami. Wzajemna aktywacja neuronów jest funkcją wagi ich powiązań. Używane w tym modelu warstwy ukryte mają na celu dodatkową modyfikację sygnału przesyłanego pomiędzy elementem wejściowym a wyjściem, która jest również modyfikowana siłą powiązania. Wiedza w tego typu strukturach kodowana jest w postaci wag powiązań pomiędzy węzłami, które są jedynie reprezentantami obiektów. Powstaje w ten sposób pewien wzorzec pobudzeniowy reprezentujący znaczenie, będący rozproszonym sub-symbolicznym obrazem pobudzeń między węzłami. Użycie struktur, opartych na sieci neuronów wzajemnie ze sobą połączonych, do modelowania pamięci semantycznej wykazało ciekawe własności nie prezentowane przez inne modele: spontaniczne generalizacje – sądy wyrażane na temat ogólnych informacji, które nie były jawnie zakodowane oraz zdolności do tworzenia nowych powiązań między kategoriami, które nie były podane podczas uczenia sieci. Zasadniczą zaletą takiego podejścia jest możliwość odtwarzania pełnej informacji dostępnej w sieci na podstawie częściowych pobudzeń.



Rysunek 1.4: Model pamięci semantycznej oparty o sieć neuronową

Przykładową aplikacją podejścia koneksyjego do modelowania pamięci semantycznej jest sieć neuronowa, którą McClelland i Rogers użyli do opisu kategorii naturalnych z dziedziny zwierząt [55]. Przedstawiona na rysunku 1.4 sieć neuronowa koduje powiązania określonego typu pomiędzy obiektami a ich cechami. Opracowany model opiera się na jednokierunkowej sieci neuronowej opisującej obiekty reprezentowane przez warstwę wejściową, za pomocą określonych typów relacji, z ich cechami kodowanymi przez warstwę wyjściową. Przedstawiona architektura w powiązaniach sieci zapisuje zdania w formie trójek obiekt – typ relacji – cecha. Jako zbiór uczący użyte zostały informacje, opisane na modelu hierarchicznym, na które składa się wiedza o obiektach zorganizowanych w taksonomię i powiązanych z nimi cechami. Obraz uczący sieć neuronową złożony został z obiektu i typu relacji, a odpowiedzią sieci była odpowiadająca im cecha. W modelu tym informacje o powiązaniach kodowane są w warstwie ukrytej. Dodatkowa ukryta warstwa nazwana „*representation*" ma na celu utworzenie wewnętrznego przedstawienia zapisanych w sieci neuronowej obiektów.

Wyniki eksperymentów pokazały, że w procesie uczenia podobne obiekty reprezentowane poprzez podobne złożenia cech i typów relacji uzyskały zbliżone reprezentacje w warstwie „*representation*" tak, że powierzchnie de-

cyzyjne separowały semantyczne grupy pojęć w sposób zbliżony do tego, jak dokonałby tego człowiek, oddzielały one np. ptaki od ssaków. Sieć poprawnie odtwarzała taksonomiczne podziały wprowadzone w modelu hierarchicznym, jednak dokonane zostało to w odmienny sposób. Nie opierały się one na bezpośrednio zdefiniowanych relacjach *is_a* a powstały poprzez utworzenie się wzorców opartych na podobieństwach i różnicach między wewnętrznymi reprezentacjami różnych pojęć. Sieć wykazała również dobre własności generalizacyjne. Uczona niepełnymi informacjami o obiekcie wykazała jego dodatkowe cechy, które pokryły się z oczekiwaniami np. sieć uczona pojęć *sparrow is_a animal*, *bird*, *living thing* wykazała podobne cechy do *robin* i *canary* przejawiające się w aktywacji powiązań *can_a fly* i *has_a wings*.

1.2 Badania pamięci semantycznej człowieka

Celem badań nad pamięcią semantyczną jest odnalezienie struktur organizacji wiedzy skojarzeniowej w umyśle, sposobu jej reprezentacji i metod wydobywania z nich informacji. Zagadnienie to angażuje wiele aspektów dotyczących poznania ludzkiego: percepcji, uwagi, przypominania, przetwarzania języka. Stąd teorie pamięci semantycznej dotyczą wszystkich tych aspektów epistemologicznych. W szczególności, badania nad pamięcią semantyczną skupiają się nad opracowaniem modelu struktury pojęć: ich organizacji, nazywania egzemplarzy, kategorii oraz sposobu ich przetwarzania. Podstawową techniką badawczą jest analiza decyzji leksykalnych [61]. Znajomość sposobów organizacji danych kontekstowych stanowi przyczynek do implementacji systemów w paradygmacie neurokognitywnym, gdzie architektura systemu informatycznego inspirowana jest wynikami badań nauk humanistyczno – biologicznych.

Istnienie szeregu teorii dotyczących sposobów organizacji i przetwarzania danych w umyśle wskazuje na brak jednoznacznej odpowiedzi na podstawowe kwestie związane z pamięcią: w jaki sposób nasza wiedza pojęciowa jest w niej przechowywana: czy są to reprezentacje oparte o cechy, prototypy, wyobrażenia oraz w jakich strukturach wiedza ta jest organizowana. Nie jasne są również sposoby użycia danych w nich zapisanych: w jaki sposób są one wydobywane, jak są wyszukiwane, w jaki sposób biorą udział w procesach decyzyjnych. Pomimo tych niejasności na bazie fenomenologicznego opisu zjawisk biologicznych, psychologicznych i neurologicznych oraz eksperymentów statystycznych powstał szereg modeli opisujących zjawiska zachodzące przy przetwarzaniu informacji przez człowieka.

Wykorzystywane w psychologii metody badania pamięci semantycznej skupiają się głównie na przeprowadzaniu testów i poprzez ich statystyczną

analizę odkrywają reguły i weryfikują teorie dotyczące organizacji wiedzy przez człowieka. Ponieważ wiedza o świecie integruje wiele obszarów psychologowie ograniczają swoje badania pamięci semantycznej do wybranych dziedzin, w szczególności do kategorii naturalnych. Wyróżnić można kilka podstawowych testów używanych w psychologii, badających pamięć semantyczną:

- Zadania skojarzeń. Swobodne skojarzenia używane były przez Freuda do studiowania osobowości. Ich układ stanowi przesłanki do wyciągania wniosków o strukturze wiedzy i jej organizacji przez człowieka. W zadaniu skojarzeń kategorialnych osoby biorące udział w eksperymencie proszone są o podanie skojarzenia odnośnie zadanej nazwy kategorii np. owoc – ?. Pewną modyfikacją tego podejścia jest metoda swobodnych skojarzeń Galtona, w której podaje się słowo – hasło prosząc o swobodne skojarzenia, które następnie są wykorzystywane do wydobycia specyficznego wspomnienia autobiograficznego. Podejście takie wykorzystywane jest do gromadzenia danych kontekstowych w kooperacyjnym przedsięwzięciu zbierającym skojarzenia on-line[5]. W ciągu 1253 dni działania zgromadził 54.988 słów powiązanych 4.950.758 powiązaniami skojarzeniowymi.

- Odtwarzanie za pomocą bodźców wywołujących. Są to testy polegające na badaniu okoliczności towarzyszących zjawisku paramnezji. Przejawia się ono we wrażeniu, które mamy, gdy usiłujemy sobie przypomnieć słowo, które wiemy, że znamy, ale nie potrafimy go przywołać. Zjawisko to polega na tym, że mamy poczucie znajomości jakiegoś faktu, ale nie potrafimy go sobie w pełni przypomnieć. Nazywane jest ono również TOT (*Tip of the tongue*) – „mam na końcu języka". Badania eksperymentalne nad tym zjawiskiem wykazały, że ludzie przypominają sobie ogólny wygląd poszukiwanego słowa, liczbę sylab czy pojedyncze litery, i czasami wystarczy niewielka pomoc, aby człowiek przypomniał sobie poszukiwaną informację [14].

  Badania nad paramnezją polegają na jej wywołaniu poprzez np. czytanie definicji rzadkich słów. W momencie, gdy zajdzie to zjawisko odnajdywaniu słowa poprzez podanie częściowej informacji: typu skojarzenia semantyczne, brzmieniowe, bądź też podpowiedziami odnośnie wyglądu lub układu liter. Na podstawie otrzymanych wyników wyciągane są wnioski odnośnie konstrukcji pamięci człowieka.

[5]http://www.wordassociation.org ,VII 2007

- Pomiar czasu decyzji semantycznej. Test ten polega na zadawaniu pytań dotyczących właściwości obiektów z różnych poziomów testowej hierarchii np. dla dziedziny zwierząt: czy kanarek je?, czy kanarek lata?, czy kanarek jest żółty?, czy kanarek jest ptakiem? itp. a następnie badaniu czasu reakcji i poszukiwaniu jego korelacji z poziomami hierarchii, na których występują określone cechy, w celu potwierdzenia efektu odległości semantycznej.

  Pomiar decyzji semantycznej przeprowadzany jest w wielu odmianach np. w weryfikacji prawdziwości zdań i czasu koniecznego na udzielenie odpowiedzi np.: „Kot jest ssakiem", lub też analizie czasu odpowiedzi koniecznego do stwierdzenia przynależności do kategorii np.: *ptak – drozd („tak") ptak – drzewo („nie").*

- Pomiar szybkości działania pamięci rozpoznawczej. Osobną klasą testów czasów reakcji jest pomiar dotyczący decyzji leksykalnych, gdzie dokonywany jest pomiar czasu odpowiedzi do oceny słów i nie-słów np.: *kot* i *ktook*. We wszystkich tych eksperymentach czas reakcji ma wskazywać wartości, jakie daje przeszukanie słownika umysłowego i przez to pozwala wnioskować o sposobie jego organizacji.

Istniejące komputerowe realizacje wymienionych testów, ukierunkowane są na badanie, za ich pomocą sposobu organizacji informacji przez człowieka a nie na modelowanie określonej teorii w systemach informatycznych. Wyróżnić tu należy system MemCog dostępny w sieci[6] przeprowadzający 10 klasycznych eksperymentów psychologii kognitywnej.

[6]http://www.du.edu/psychology/methods/experiments/ ,VII 2007

Rozdział 2

Reprezentacja wiedzy

2.1 Zagadnienie reprezentacji wiedzy

Fundamentalną koncepcją sztucznej inteligencji jest pojęcie reprezentacji wiedzy. Określa ono sposób, w jaki informacja jest przez komputer strukturalizowana, jak jest składowana i przetwarzana. Formalizacja wiedzy do formy, która może być interpretowana maszynowo wraz ze sposobami jej przetwarzania umożliwia realizację wnioskowania, co jest podstawą inteligentnego działania programu [36]. W odróżnieniu od technik programowania (języków, środowisk wytwarzania oprogramowania) metody reprezentacji wiedzy nie zmieniają się szybko. Istnieje szereg już opracowanych paradygmatów stanowiących podstawę do tworzenia aplikacji informatycznych. Reprezentacja wiedzy jest pierwotna w stosunku do sposobu jej realizacji w systemie informatycznym. Jest metodą, za pomocą której zagadnienie uzyskuje swoją formalną implementację w określonym języku programowania [36].

Człowiek swoją wiedzę wyrażać może na wiele sposobów: może do tego celu użyć gestów, rysunków, wzorów. Najbardziej powszechną metodą jest język naturalny, który pomimo, że cześć jego została sformalizowana, wciąż jest dalece poza zasięgiem maszyn cyfrowych.

Zagadnienie reprezentacji wiedzy, zwłaszcza w odniesieniu do języka naturalnego pociąga za sobą szereg pytań: czym jest wiedza, jak ją reprezentować, jak z niej korzystać (przetwarzać i podejmować decyzje). Problem reprezentacji wiedzy wyrasta z pytań o sposób poznawania człowieka. Rezultaty tych rozważań znajdują implementację w systemach informatycznych, a najlepiej widoczne są w zastosowaniach sztucznej inteligencji. W kognitywistce, reprezentacja wiedzy skupia się na pytaniu wyrosłym z epistemologii: jak ludzie organizują i przetwarzają informację pochodzącą z doświadczenia. W sztucznej inteligencji podstawowym celem jest przyjęcie takiego modelu

reprezentacji, który umożliwiałby programowi przetwarzać informację tak, by wyniki uzyskane były podobne ludzkiemu działaniu. Te dwie dziedziny są ze sobą ściśle związane – sztuczna inteligencja czerpie inspiracje z rozważań dotyczących ludzkiego poznania, zakorzeniając się tym samym w naukach o poznaniu. Na zagadnienie reprezentacji wiedzy można zatem patrzeć jako na epistemologię stosowaną.

2.1.1 Wiedza

Wiedza oznacza zbiór wiadomości z określonej dziedziny, będącymi wszelkimi zobiektyzowanymi i utrwalonymi formami kultury umysłowej i świadomości społecznej powstałymi w wyniku kumulowania się doświadczeń i uczenia. Z tego punktu widzenia można powiedzieć, że wiedza jest symbolicznym opisem otaczającego nas świata rzeczywistego. Na wiedzę składa się zbiór faktów, stanowiących opis otoczenia, a także relacji, jakie zachodzą między jego elementami, jak również procedur, które mają za zadanie manipulować tymi relacjami [59]. Podział ten odpowiada sposobowi składowania danych w pamięci proceduralnej i deklaratywnej w procesie poznawczym człowieka.

Encyklopedia PWN [33] określa wiedzę jako ogół wiarygodnych informacji o rzeczywistości, wraz z umiejętnością ich wykorzystania. W szerokim znaczeniu, jest to wszelki zbiór informacji, poglądów, wierzeń, którym przypisuje się wartość poznawczą lub (i) praktyczną.

Wiedza przejawia się w działaniu w świecie. Poprzez prawdziwość prognoz przez nią tworzonych odbywa się jej weryfikacja. Podstawowym celem działania obrazu świata reprezentowanego w wiedzy jest rozwiązywanie postawionych pytań lub interpretacja faktów.

Istotnym aspektem wiedzy jest uczenie i wnioskowanie nowych faktów, które odbywa się na zasadzie kojarzenia wedle określonych reguł. Podczas pozyskiwania nowych danych i po ich analizie do bazy wiedzy trafiają kolejne składowe, które integrowane są z już istniejącą w niej wiedzą. Służyć będą one w przyszłości do analizowania kolejnych wiadomości.

Jedną z form organizowania wiedzy stosowanych przez ludzi jest asocjacja empiryczna. Sformułowanie to określa sposób nabywania i użycia wiedzy specjalistycznej w określonej dziedzinie. Opiera się ona na szeregu skojarzeń dotyczących przyczyn obserwowanych danych i faktów. Specjaliści czerpią swą wiedzę w trakcie praktyki zawodowej – uczenie odbywa się na zasadzie uogólniania faktów, z których ekstrahowane są następnie reguły stanowiące wiedzę. Reguły te są reprezentowane różnorodnie: mogą być to np.: powiązania logiczne, jak również wyekstrahowane prototypy. Ekspert posiadający takie reguły używa metod heurystycznych do powiązania niepewnych i często błędnych danych, w celu postawienia diagnozy lub interpretacji faktów.

Wysnute za pomocą tej metody wnioski stają się dla podmiotu poznającego kolejnym faktem w jego bazie wiedzy.

Formalizacja sposobów zapisu wiedzy w dziedzinie sztucznej inteligencji jest problemem, który nie znalazł jeszcze pełnego jednolitego rozwiązania – do różnych zastosowań używane są różne podejścia. Teoria reprezentacji wiedzy jest słabo rozwinięta, a jej biologiczny wzór – sposób reprezentacji wiedzy w mózgu nie jest do końca jasny. Stanowi on wciąż niedościgły wzór i inspirację do dalszego rozwoju metod reprezentowania wiedzy w systemach informatycznych.

Rozważania nad reprezentacją wiedzy rozpoczynają się przedstawieniem zagadnienia poprzez wrócenie do pytania o istotę reprezentacji wiedzy. W dalszej części rozdziału przedstawione zostaną wybrane metody reprezentowania wiedzy z użyciem notacji opartej na powiązaniach, która może zostać zastosowana do modelowania semantyki, opartej na asocjacjach między pojęciami. Dla reprezentacji wiedzy, z których korzystano do realizacji pamięci semantycznej opisane zostały podstawowe o modele obliczeniowe.

2.1.2 Reprezentacja wiedzy

Rozważając pojęcie reprezentacji wiedzy szereg prac opisuje, jakie powinna mieć ona właściwości, jednak z rzadka odpowiadają, czym ona jest. By uzyskać odpowiedź na to pytanie należy cofnąć się do źródłowego znaczenia tego pojęcia i zastanowić się, czym jest ono w swojej istocie. Wyszczególnić można pięć głównych cech reprezentacji wiedzy [22]:

1. Najbardziej fundamentalny substytut dla rzeczy używany do określania świata zewnętrznego.

   Każdy rozumny byt zdolny wnioskować[1] staje przed faktem, że obiekty, o których zachodzi wnioskowanie znajdują się w świecie zewnętrznym, natomiast sam proces myślenia odbywa się wewnątrz podmiotu. Ta dychotomia wymaga po stronie podmiotu substytutu do interakcji ze światem. Powstają tu dwa zasadnicze pytania: w jaki sposób zachodzi odpowiedniość między surogatem a rzeczywistością oraz jaka jest dokładność: jak reprezentacja mentalna jest bliska rzeczywistej rzeczy. W tym aspekcie reprezentacji wiedzy pojawia się również pytanie o sposób w jaki surogat funkcjonuje zarówno jako reprezentacja kategorii naturalnych, jak i abstraktów, procesów i innych reprezentacji wewnętrznych. Konsekwencją tego jest, że myśląc o świecie przy użyciu jego

[1] termin wnioskowanie użyty jest tu w szerszym znaczeniu, jako zdolność wytwarzania nowych wyrażeń ze starych, a nie tylko w odniesieniu do wnioskowania logicznego

reprezentacji dokonujemy redukcji i operujemy na jego modelu. Dalszą konsekwencją używania reprezentacji jako aproksymacji rzeczywistości jest nieuniknioność powstawania błędnych wniosków, nawet przy założeniu poprawności używanych reguł, ponieważ każda aproksymacja skupia się na pewnych aspektach, pomijając inne.

2. Reprezentacja wiedzy jest zbiorem ontologicznych założeń.

   Ponieważ o świecie myślimy z użyciem pojęć, używając reprezentacji świata dokonujemy pewnych założeń odnośnie tego co i jak postrzegamy. Jest to pewne potwierdzenie koncepcji znanej jako hipoteza Sapira-Whorfa [100], postulującej, iż przyjęcie określonego sposobu widzenia świata determinuje, co w tym świecie jesteśmy w stanie zobaczyć, i co więcej, w jaki sposób będziemy to doświadczenie przetwarzać. Tak jak dla człowieka język jest pryzmatem, przez który postrzegamy rzeczywistość, tak reprezentacja wiedzy jest niejako okularami, które nakładamy do realizacji określonego zadania. Staranna analiza natury tych okularów dostarcza obszaru, który może zostać z ich użyciem zamodelowany, czyli właściwego użycia reprezentacji wiedzy. Spojrzenie na reprezentację wiedzy z epistemologicznego punktu widzenia pozwala na ogólne określenie, obszarów jej zastosowań. Założenia ontologiczne są sposobem, w jaki patrzymy na świat i w nie mniejszym stopniu zniewalają, ograniczają nasze poznanie, co kantowskie formy aprioryczne (czas i przestrzeń, w których postrzeżenie istnieje z konieczności). Pozwalają one nam wyraźnie zobaczyć jedynie pewien wycinek świata, co do którego mamy intencję, że jest istotny. Intencjonalność, skupienie się na określonej dziedzinie jest tu kluczowe: ze względu na złożoność świata nasz aparat poznawczy musi dokonywać selekcji tego, co jest istotne, a co nie.

   Przyjęcie założenia odnośnie technologii reprezentacji wiedzy określa punkt widzenia determinujący, jakiego rodzaju rzeczy nas będą interesować: logika skupiać będzie nas na bytach i relacjach między nimi, systemy regułowe sprowadzają świat faktów ujętych w pary implikacyjne, trójki OAV (paragraf 2.1) w formy związków wiążące obiekty z atrybutami, ramy narzucają myślenie w formie struktur organizujących dane. Każda z metod dostarcza więc własnego punktu widzenia na to, co jest ważne, a co za tym idzie własnych założeń, co do tego, jaki jest świat opisywany poprzez określony sposób reprezentacji.

3. Częściowa teoria rozumowania.

   Reprezentacja wiedzy jest zbiorem pojęć, danych i reguł wnioskowania. Należy odróżnić ją jednak od struktur danych, w rozumieniu tym np.

sieć semantyczna jest reprezentacją, a graf jest tu strukturą danych lingwistycznych. Semantyka jest w tym przypadku przejawem przyjętego sposobu reprezentacji, wyrażanego w zaimplementowanych strukturach danych.

Pierwotnym założeniem reprezentacji wiedzy jest możliwość rozumowania z jej użyciem. Zakłada ona wiedzę o tym, co znaczy rozumowanie, jakie ograniczenia są na nie nałożone: co można, co należy wywieść z wiedzy posiadanej. Przyjęte założenia odnośnie rozumowania wskazywać będą na to, czym jest inteligencja: z punktu widzenia logiki będzie to zdolność myślenia dedukcyjnego z użyciem określonych formalnych reguł. W wypadku tym, dostępne operacje np. implikacja wskazywać będą, jakiego rodzaju konkluzji możemy dokonać. Przyjęcie określonego sposobu reprezentacji pociągać za sobą będzie możliwe sposoby wnioskowania.

4. Nośnik do wykonywania obliczeń.

   Reprezentacja wiedzy pozwala stworzyć środowisko do realizacji obliczeń. Z czysto mechanistycznego punktu widzenia na rozumowanie popatrzeć można jako na proces obliczeniowy. By używać reprezentacji wiedzy musimy ją przetwarzać. Tak więc, z reprezentacją wiedzy nierozerwalnie związane będzie pytanie o wydajność obliczeniową. Przeliczanie reprezentacji ma na celu realizację rozumowania przejawiającego się we wnioskowaniu, a rodzaje inferencji, jakie będą dostępne zależne są od sposobu organizacji informacji. Istniejące sposoby reprezentacji oferują możliwość przechowywania danych oraz metody przeprowadzania określonych rodzajów wnioskowań. Np.: organizacja obiektów przy użyciu np. ram w taksonomie sugeruje użycie wnioskowania hierarchicznego i realizacji dziedziczenia strukturalnego. Systemy regułowe wspomagają wnioskowanie dostarczając mechanizmy inferencji z danych zorganizowanych w grupy implikacji.

   Konieczność przetwarzania reprezentowanej wiedzy nierozerwalnie wiąże ją z wydajnością obliczeniową. W kontekście tym reprezentacja wiedzy jest pewnym kompromisem między szybkością przetwarzania, a elastycznością wyrazu języka reprezentacji. Systemy sztucznej inteligencji muszą więc kłaść duży nacisk na efektywne przetwarzanie dużych zasobów informacyjnych, jako że celem posiadania przez system wiedzy jest jego inteligentne zachowanie, a podstawowym celem reprezentacji wiedzy jest umożliwienie wnioskowania: wyciągania konkluzji z wiedzy, które nie są jawnie w nich zapisane. Wiedza musi być efektywnie reprezentowana w znaczący dla niej sposób. Dzięki wydajności

w składowaniu faktów potrafić będziemy wywieść nową wiedzę z już istniejącej, która pozwalać będzie na wsteczne powiązanie ze światem zewnętrznym. Jedną z metod poprawiania wydajności będzie np. dokonywanie generalizacji, tak by otrzymywać abstrakcje, które opisują ogólne cechy dla podzbioru obiektów.

5. Nośnik ludzkiego wyrazu w postaci np.: języka, gestów, dźwięków (muzyki), wzorów etc.

   Reprezentacja wiedzy jest sposobem, w jaki ludzie wyrażają swoje odniesienie do świata i sposobem, w jaki są w nim. Pozwala ona zarówno na myślenie, wyrażanie, jak i komunikację z innymi osobami, czy też środowiskami przetwarzającymi informację. Reprezentacja wiedzy jest więc sposobem wyrazu pewnego aspektu dotyczącego świata, w kontekście tworzenia komunikatywnego wyrażenia (czyli wyobrażenia, które może zostać przekazane). Spojrzenie na reprezentację wiedzy, jako na medium przekazu wyrażenia implikuje pytanie o jakość, zarówno przekazu, jak i prezentacji: na ile jest ona ogólna, jak precyzyjna, ale również jak łatwo można przy tak przyjętym medium dokonać aktu komunikacyjnego. Sama możliwość komunikacji nie jest jedynym czynnikiem określającym wymianę informacji, równie ważna jest też ilość energii jaka musi zostać zużyta na jej dokonanie. W aspekcie tym reprezentacja wiedzy jest językiem, w którym się komunikujemy, więc konieczna jest łatwość jej użycia.

2.2 Reprezentacja wiedzy oparta na powiązaniach

Reprezentacja wiedzy za pomocą sieci powiązanych ze sobą pojęć jest bardzo wygodnym z punktu widzenia człowieka sposobem reprezentacji. Siłą tego podejścia jest możliwość jej graficznego przedstawienia, co jest czytelną prezentacją zagadnienia dotyczącego określonej dziedziny. Drugą zaletą, jest możliwość algorytmicznego przetwarzania informacji zapisanej w takich strukturach wiedzy. Wynikająca z tego możliwość implementacji inferencji umożliwia wydajną obróbkę i składowanie tego typu danych przy użyciu maszyny cyfrowej.

Istnieje duża liczba metod wykorzystujących podejście oparte na powiązaniach między obiektami w obrębie modelowanego zagadnienia. Rozdział ten opisuje najbardziej typowe sposoby reprezentacji wiedzy z użyciem tej notacji, które mogą zostać użyte do opisu pewnych aspektów języka naturalnego oraz wybrane, związane z nimi modele obliczeniowe.

2.2.1 Strukturalizowane obiekty

Jednym z najprostszych sposobów reprezentacji wiedzy jest notacja A – V (*attribute – value*). Pozwala ona dokonywać opisu jednego (domyślnego) obiektu poprzez zestaw wartości jego atrybutów.

Przykładem zastosowania takiej reprezentacji są reguły produkcji, będące połączeniem notacji A – V z szablonem IF – THEN. Podejście to pozwala na modelowanie wiedzy poprzez definiowanie reguł budujących sieć powiązań, typowo służącej do opisu ciągów przyczynowo skutkowych.

Metoda opisu za pomocą par *warunek – działanie* pochodzi z badań Newell i Simona [62] nad modelowaniem sposobu działania percepcji i aparatu poznawczego człowieka. Ponieważ sam zbiór twierdzeń nie jest wystarczający do opisania bardziej złożonej dziedziny wiedzy, potrzebne są jeszcze reguły operujące na tych twierdzeniach. Zapis wiedzy w tej metodzie dokonywany jest przez pary atrybut – wartość i łączeniu ich z użyciem reguł o konstrukcji jeśli – to. Pomimo swej prostoty metoda ta umożliwia opis dość złożonych problemów, jak również pozwala przeprowadzić zaawansowane wnioskowanie. Reprezentacja regułowa często jest wykorzystywana w systemach klasyfikacji, gdzie przynależność do określonej klasy jest wypadkową reguł zapisanych w systemie [29]. Ogólna postać reguły produkcji ma postać:

IF [warunek1 AND warunek2 ...] THEN akcja

Lewa część formuły określa warunki jej stosowalności, które mogą być złożone za pomocą funktorów (np.: koniunkcji, alternatywy). Prawa część określa jej działanie. Przy formalnym zapisie reguł opuszcza się czasem symbol IF a zamiast instrukcji THEN używa się symbolu implikacji. Przesłanka jest wyrażana wtedy jako połączenie za pomocą funktorów logicznych pewnej liczby twierdzeń, np. IF (A = u AND F = w) OR F = v THEN G = y. Korzystając z symbolu implikacji regułę tę można zapisać w postaci ((A, u) i (F, w)) lub (F,v) $\Rightarrow$ (G,y), która jest bardziej zwarta.

Elementarne zdania if – then opisują w jednolity sposób związki i relacje między faktami. Podejście to jest naturalne dla człowieka, który w ten sposób zwykł wyrażać swoją wiedzę o zależnościach przyczynowo – skutkowych. Połączenie faktów i reguł można traktować jako mechanizm będący sprzężeniem pomiędzy wiedzą deklaratywną i proceduralną. W podstawowej realizacji tej reprezentacji wiedzy reguły, w odróżnieniu od np. procedur, nie mają możliwości odwoływania się do siebie. Minimalizuje to oddziaływanie ich pomiędzy sobą, pozwalając hermetyzować obszary wiedzy programu, z drugiej strony utrudnia to znacznie napisanie bardziej złożonego algorytmu.

Czytelność wiedzy zapisanej z użyciem reguł produkcji jest zależna od ich liczby. Dla niewielkiego rozmiaru bazy wiedzy łatwo przewidzieć skut-

Rysunek 2.1: Przykład trójek OAV opisujących lokatę

ki wykonywania instrukcji warunkowych, jednak wraz ze wzrostem ich liczby czytelność maleje, tak że przy osiągnięciu pewnej liczby reguł system staje się nieczytelny, a działanie programu nieprzewidywalne. Fakt ten utrudnia analizę, aktualizację i wprowadzanie nowej wiedzy. W wypadku braku wymogu śledzenia wnioskowania i zasad, poprzez które podjęta została określona decyzja reprezentacja wiedzy oparta na tym podejściu nadaje się do zastosowania w stosunkowo dużych systemach informatycznych. W wypadku zaistnienia konieczności analizy wnioskowania reprezentacja z użyciem reguł produkcji jest efektywnym sposobem składowania wiedzy umożliwiającym przejrzyste i czytelne dla człowieka zapisanie wiedzy dotyczącej jedynie małego obszaru [101].

Jednym z rozszerzeń reprezentacji z użyciem reguł produkcji jest wprowadzenie stopni pewności, liczb zazwyczaj z przedziału (-1, 1) lub (0,1), za pomocą których określa się pewności konkluzji występującej w danej regule. Zbiór reguł można rozpatrywać jako szczególny sposób zapisu pewnej sieci twierdzeń, ponieważ z prawdziwości jednego twierdzenia mogą wynikać inne.

Naturalnym rozszerzeniem notacji A – V są twierdzenia pozwalające opisać fakty za pomocą uporządkowanych trójek w postaci:

$$\langle \mathrm{O}, \mathrm{A}, \mathrm{V} \rangle = \langle \text{obiekt} \rangle, \langle \text{nazwa atrybutu} \rangle, \langle \text{wartość atrybutu} \rangle$$

Zapis ten pozwala wyrazić twierdzenia wyrażające przysługiwanie obiektowi O, atrybutu o nazwie A i jego wartości V. Trójki O-A-V (*object, attribute, value*) [21] również mogą być przedstawiane w postaci graficznej. Przykład grafu opisujący strukturę lokaty poprzez cechy będące wartościami składającymi się na nie atrybutów przedstawiony został na rysunku 2.1.

Metoda ta jest podstawą, na bazie której działają współczesne systemy informatyczne: wykorzystywana jest ona w budowie ram (stanowiących podstawę paradygmatu obiektowego). Jest to reprezentacja, w której obiekty są strukturalizowane w klasy zawierające określone pola będące cechami. Notacja OAV stanowi również podstawę RDF [48].

W ogólnym podejściu do modelowania wiedzy z użyciem notacji trójkowej OAV zamiast obiektu może się znajdywać konkretna rzecz lub abstrakt – pojęcie. W sytuacji tej atrybut jest cechą posiadaną przez obiekt, wyróżniającą go spośród innych. Podczas modelowania złożonych struktur z użyciem notacji OAV wykorzystuje się słowniki nazw obiektów i atrybutów oraz wartości – zakresów, jakie mogą one przyjmować. Dzięki takiemu podejściu zapis jest oszczędniejszy i pozwala wydajnie gospodarować zasobami pamięciowymi. Uporządkowana trójka w niektórych systemach modelujących powiązania rozmyte rozszerzana jest do postaci czwórki – *quadra*. Zapis twierdzeń składa się w takiej sytuacji z tych samych elementów co trójka OAV oraz współczynnika określającego stopień pewności powiązania CF (*Certainty Factor*).

$$\langle \mathrm{O}, \mathrm{A}, \mathrm{V}, \mathrm{CF} \rangle = \langle \text{obiekt} \rangle, \langle \text{nazwa atrybutu} \rangle, \langle \text{wartość atrybutu} \rangle, \langle \text{stopień pewności} \rangle.$$

Notacja ta pozwala na wprowadzenie twierdzeń przybliżonych poprzez przypisanie współczynnika określającego stopień pewności, typowo będącego wartością z przedziału <0,1>. Stopień pewności jest cechą przypisywaną określonemu powiązaniu, w określonym kontekście, w związku z tym brak jest formalnych metod jego określania. Przypisywanie stopnia pewności twierdzeniom jest ogólnym mechanizmem rozszerzenia siły wyrazu, którego interpretacja i sposób realizacji w dużym stopniu zależą od zastosowania. Wspomnieć w tym miejscu należy również na potencjalną możliwość dalszego rozszerzenia modelu OAV. Można ją uzyskać np. wprowadzając dodatkowe współczynniki dla wartości atrybutów, jak również obok pewności modelować siłę deskrypcyjną określonego powiązania, co szczególnie przydatne jest przy dużej liczbie cech związanych z obiektem. Na uwadze należy mieć jednak fakt, że rozszerzenia rozmywające wiedzę OAV mogą w szczególności prowadzić do sprzeczności w obrębie modelowanej dziedziny.

Poprzez łączenie ze sobą trójek notacja OAV pozwala na budowanie wyrażeń pozwalających opisywać złożone zagadnienia. To ogólne podejście do modelowania informacji stało się podstawą reprezentowania wiedzy, która może być przedstawiona w postaci grafów, będącej obok strukturalizowanych obiektów (w paradygmacie obiektowym) jedną z bardziej ekspresywnych metod wyrazu informacji.

2.2.2 Sieci semantyczne

Idea łączenia razem pojęć w sieć powiązanych ze sobą wyrażeń jest bardzo stara, prawdopodobnie może być datowana na czasy dużo przed Arystotelesem. W nowoczesnym podejściu terminu *sieci semantyczne* jako pierwszy

użył Richard H. Richens (Cambridge Language Research Unit w 1956) w projekcie *Interlingua*, którego celem było maszynowe tłumaczenie języków [73]. Kolejnym etapem rozwoju sieci semantycznych był system M. Rossa Quilliana (*The teachable language comprehender*), pozwalający modelować znaczenie słów w sposób umożliwiający przeprowadzanie na nich obliczeń [70].

Na przełomie lat 1970 i 1980 idea powiązań między słowami została rozwinięta w powszechnie znany *hipertext*, będący podstawą funkcjonowania sieci WWW. Obecnie, kolejnym jego rozwinięciem jest umożliwienie określania typu powiązania, co ma być podstawą semantycznej sieci WWW.

Sieci semantyczne są jednym ze sposobów reprezentacji wiedzy, który poprzez tworzenie powiązań między pojęciami używany jest do składowania semantyki. Podejście to opiera się na łączeniu określonym typem powiązania pojęć, które są przeważnie reprezentowane przez słowo. U podłoża takiego wyrażania semantyki leży teoriopoznawcze założenie, że znaczenie znajduje się w powiązaniach między pojęciami. Znaczenie określonego pojęcia wynika z jego powiązań z innymi pojęciami. Przy przyjęciu takiego założenia rozumienie, jest zdolnością do umiejscowiania doświadczenia w sieci pojęciowej.

Siłą opisu sieciowego za pomocą wyrażeń w postaci trójek OAV (lub analogicznych do nich) jest czytelna dla człowieka reprezentacja w postaci grafu, którego wierzchołki reprezentują pojęcia a krawędzie relacje określonego typu pomiędzy nimi. W szczególności mogą być to relacje semantyczne, nadające powiązaniu znaczenie, dzięki czemu podczas przetwarzania może być ono interpretowane w określony sposób. To ogólne podejście do reprezentacji wiedzy jest odzwierciedleniem jednej z metod jakiej człowiek używa do organizacji informacji – opartej o typowane skojarzenia.

Sposób reprezentacji wiedzy oparty na typowanych asocjacjach między pojęciami powszechnie używany jest w słownikach maszynowych operujących semantyką. Najbardziej powszechnymi typami relacji semantycznych w sieciach leksykalnych są:

- meronimia – opisująca bycie częścią
- hypernimia i hyponimia - opisujące strukturę taksonomii
- synonimiczność – opisująca bliskość znaczeniową słów
- antynomia – opisująca przeciwieństwo dwóch pojęć

Wymienione typy relacji pozwalają na opisanie niektórych zależności lingwistycznych między słowami lub pojęciami występującymi w języku. Istnieje cały szereg innych, nie wspomnianych tutaj typów relacji, które w konkretnych zastosowaniach wprowadzane są w celu zwieszenia siły wyrazu tego

podejścia użytego do modelowania określonego obszaru wiedzy. Liczba użytych typów relacji jest istotna z punktu widzenia siły ekspresji wiedzy, która za ich pomocą może zostać wyrażona. Jednak nadmierna ich liczba zmniejsza czytelność zapisu a brak wprowadzenia interpretacji typów powiązań (np. proceduralnego) właściwie nie zmienia funkcjonalności bazy wiedzy pozostawiając ją do realizacji osobie z niej korzystającej.

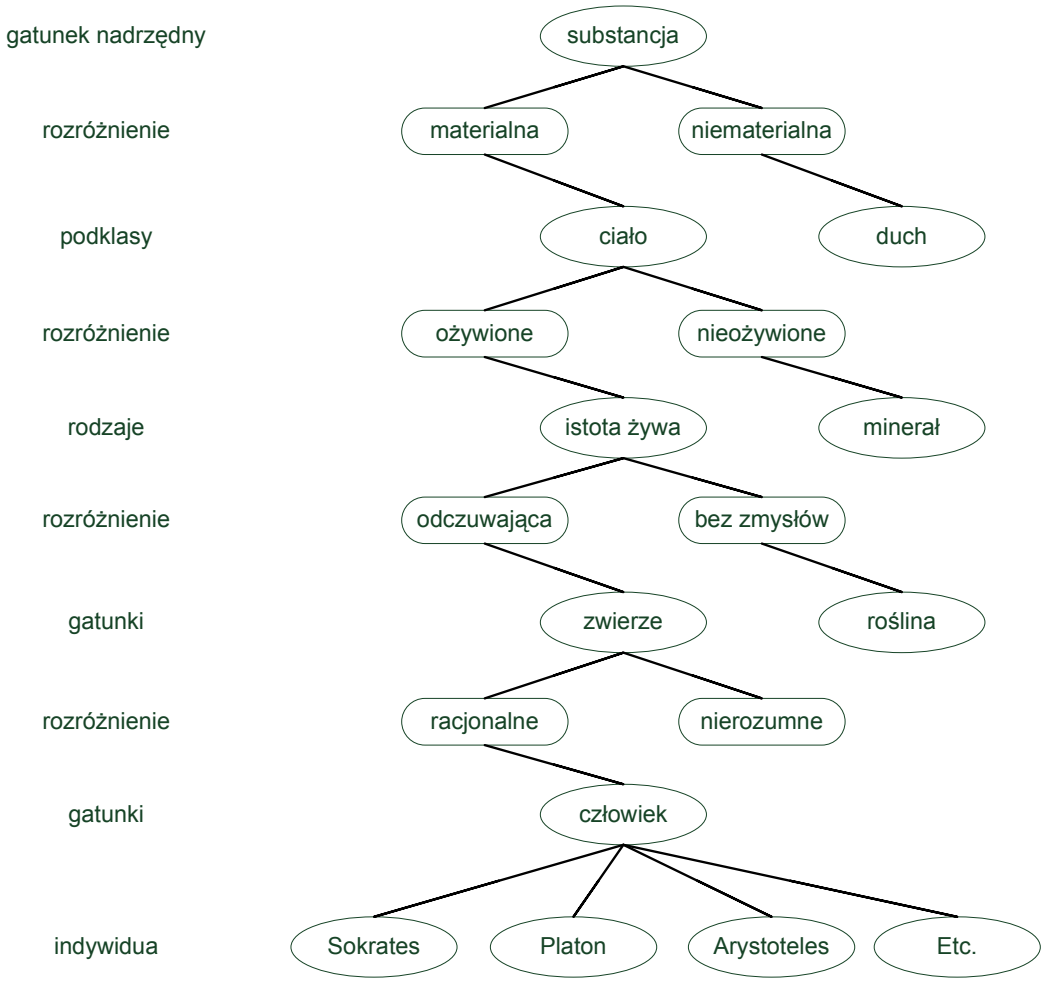
Obok wymienionych typowych relacji semantycznych często wprowadza się dodatkową relację instancjonującą indywidua istniejące w sieci powiązań. Ten typ relacji pozwala wskazać byty jednostkowe, nie jest jednak w szczególny sposób odróżniany od innych, co jest głównym zarzutem dla tego sposobu reprezentacji wiedzy i podstawowym problemem związanym z sieciami semantycznymi. Brak rozdzielenia klas i instancji wynikający z przyjętej reprezentacji kompensowany jest odwołaniem do proceduralnej interpretacji powiązania, dzięki czemu możliwe jest określenie czy opisywane jest wystąpienie jakiejś klasy, czy jej własności.

Typowe podejście do wnioskowania w sieciach semantycznych opiera się na własnościach grafowych przetwarzanych sieci. Pozwalają one wywieść nową wiedzę nie będącą jawnie zapisaną w strukturze sieci a wynikającą z powiązań między węzłami. Podejście to nie dokonuje rozróżnień między typami powiązań, co jest jego istotnym ograniczeniem. Użycie innych metod wnioskowania (np. logiki) czy też szukania, wymaga konwersji tej metody reprezentacji do innej, za którą stoi określony model obliczeniowy pozwalający realizować określone inferencje. W związku ze swoją ogólnością wynikająca z elementarnej jednostki wiedzy konstruującej sieci semantyczne (trójki w postaci obiekt – relacja – cecha) możliwe jest dokonanie takiej transformacji do dużej liczby modeli obliczeniowych bez straty, bądź konieczności wprowadzania nowej wiedzy.

Za Sowa [84] można wskazać najpopularniejsze typy reprezentacji wiedzy, u podłoża których leży idea wiązania pojęć ze sobą w sieć, która może być przedstawiona w postaci graficznej, za pomocą grafu powiązań. Metoda wizualizacji z użyciem grafu jest punktem spinającym klasę metod reprezentacji wiedzy występujących pod wspólna nazwą sieci semantycznych, używanych do opisu zagadnień występujących w języku naturalnym.

2.2.2.1 Sieci definicyjne

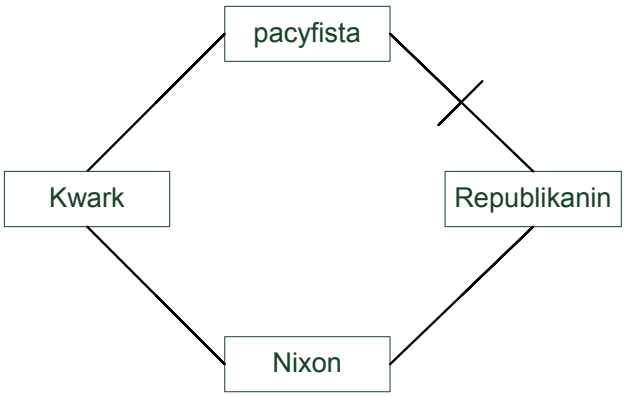
Podstawą na bazie której zorganizowany jest ten typ sieci jest relacja *is_a*. Pozwala ona przedstawiać drzewo taksonomii opisywanej dziedziny. Utworzona w ten sposób hierarchia umożliwia wprowadzenie dziedziczenia cech poprzez propagację właściwości z pojęć bardziej ogólnych do szczegółowych obiektów.



Rysunek 2.2: Drzewo Porfirusza

Podstawą współczesnej nauki o taksonomiach jest metoda drzewa Porfirusza *(arbor Porphyrii)*, oparta na przedstawieniu zależności rodzajów i gatunków w formie drzewa. Podejście to jest ilustracją metody Arystotelesa definicji kategorii poprzez wyspecyfikowanie klasy (typu ogólnego) *genus* i rozróżnień *differentiae*, które pozwalają odróżnić podtypy od naddtypów. Rozróżnienie uniwersalii i indywiduów w podejściu hierarchicznym zrealizowane jest poprzez umieszczenie obiektów w liściach drzewa. System klasyfikacji hierarchicznej Porfirusza przedstawiony został na rysunku 2.2.

Współcześnie realizowane reguły dziedziczenia są specjalnym przypadkiem arystotelejskich sylogizmów, w których dwoma przesłankami są nadklasa oraz rozróżnienie, i jeżeli są one prawdziwe to spełnione są warunki konieczne do dziedziczenia własności z nadrzędnych klas do podrzędnych,



Rysunek 2.3: Trójkąt Nixona

np. na rysunku 2.2 klasa „substancja" z rozróżnieniem na „materialną" i „niematerialną" wydziela podklasy „ciało" i „duch". „istota żywa" dziedziczy właściwość „substancja materialna" z węzła „ciało" i dodaje właściwość „ożywiona" wynikłą z położenia w taksonomii.

Drzewo Porfirusza jest klasą logik monotonicznych, w których nowa informacja monotonicznie zwiększa ilość twierdzeń, które mogą zostać udowodnione. Żadna z zapisanych w drzewie hierarchii informacji nie może zostać usunięta, bądź modyfikowana, za pomocą dodania nowych informacji.

Osobnym typem sieci hierarchicznych są grafy umożliwiające składowanie wiedzy należącej do klasy logik niemonotonicznych pozwalających do istniejących reguł dodawać nową informację, która dokonuje zmiany reguł, bądź też ich anulowanie np. poprzez blokowanie dziedziczenia. Podejście takie może być użyteczne w wielu zastosowaniach, jednak powodować może problemy wymagające rozwiązywania konfliktów. Typowym przykładem jest tzw. diament Nixona (rysunek 2.3) obrazujący konflikty spowodowane dziedziczeniem z dwóch różnych naddypów. Na rysunku 2.3 przedstawiony został elementarny problem wynikający z dopuszczenia wielodziedziczenia. Wiedza zapisana dla pojęcia *kwakier*[2] określa, że jest on pacyfistą, natomiast dla pojęcia *republikanie* określone zostaje, że nie są oni pacyfistami. Poprzez dopuszczenie wielodziedziczenia powstaje konflikt: czy Ryszard Nixon, będący zarówno republikaninem jak i kwakierem, i dziedziczący własność „pacyfista" poprzez dwie ścieżki, jest nim czy nie? Rozwiązanie tego typu konfliktów wymaga wprowadzenia dodatkowych metod obsługujących wyjątki. By uniknąć tego typu niejednoznaczności wiele implementacji rezygnuje z wielodziedziczenia, co zmniejsza siłę ekspresji metody na rzecz bardziej systematycznego podejścia gwarantującego globalną spójność.

[2]członek protestanckiej sekty „*stowarzyszenie przyjaciół*"

Metoda hierarchicznego dziedziczenia stanowi podstawę dla języków programowania obiektowego. Pomimo swojego wieku, podejście to wciąż jest powszechnie używane do organizowania wiedzy w systemach informatycznych. Zaznaczyć należy tu fakt, że ogranicza się tylko do jednego aspektu – hierarchii, co w zastosowaniu dla opisu pojęć występujących w języku naturalnym jest znacznym uproszczeniem.

2.2.2.2 Grafy egzystencjalne

Sanders Peirce w latach 1880 – 1885 utworzył notację algebraiczną nazwaną *grafami konceptualnymi*. Metoda ta służąca do modelowania wiedzy symbolicznej została rozwinięta przez prekursora teorii mnogości Giuseppe Peano we współcześnie używaną notację rachunku predykatów.

W pierwszej wersji grafy egzystencjalne opisywane były poprzez graf relacji między pojęciami (podobny do sieci hierarchicznych). Zawierał on tylko dwa operatory logiczne: koniunkcję i kwantyfikator egzystencjalny. Pozostałe operatory występujące w logice takie jak negacja, implikacja, alternatywa wymagały rozszerzenia tej reprezentacji. Brakiem była tu również niemożność zapisu uniwersalnego kwantyfikatora *każdy* pozwalającego na wprowadzenie zakresu stosowania operatora określającego jaka część formuły jemu podlega. Problem reprezentacji zakresu został rozwiązany poprzez wprowadzenie dodatkowych reprezentacji graficznych: zrealizowane zostały one na bazie owalu wyznaczającego podgraf sieci konceptualnej, do którego stosują się operatory. Połączenie owali z koniunkcją i kwantyfikatorem egzystencjonalności (reprezentowanym przez linię) umożliwiło wyrazić wszystkie operatory logiczne używane w notacji algebraicznej.

Przykładem zdania zapisanego z użyciem grafu egzystencjalnego jest twierdzenie *Jeśli student posiada indeks, to studiuje*. Zdanie to może zostać zapisane w postaci graficznej z użyciem grafu egzystencjalnego przedstawionego na rysunku 2.4. Przedstawione zdanie zawiera zewnętrzny owal, który reprezentuje szablon dla wyrażenia *jeśli*, zawierającego predykat *student* połączony linią reprezentującą kwantyfikator egzystencjalny z *posiada*, który połączony jest linią z *indeks*. Zdanie to może zostać zapisane w postaci algebraicznej:

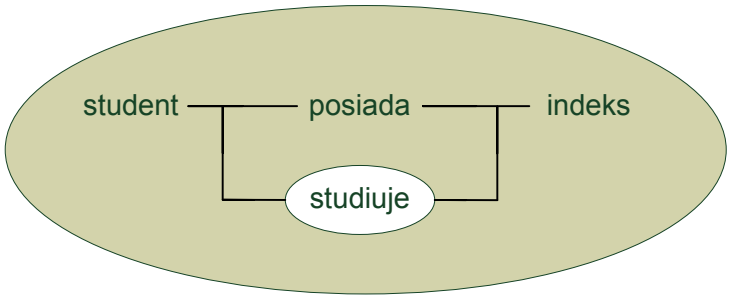
$$\neg(\exists x)(\exists y)(student(x) \wedge indeks(y) \wedge posiada(x,y) \wedge \neg studiuje(x))$$

, co jest równoważne[3]:

[3]korzystając z:

$\neg(\exists_x p(x)) \equiv \forall_x \neg p(x)$ oraz z

$\neg p(x) \vee \neg q(x) \vee ... \vee z(x) \equiv p(x) \wedge q(x) \wedge ... \Rightarrow z(x)$



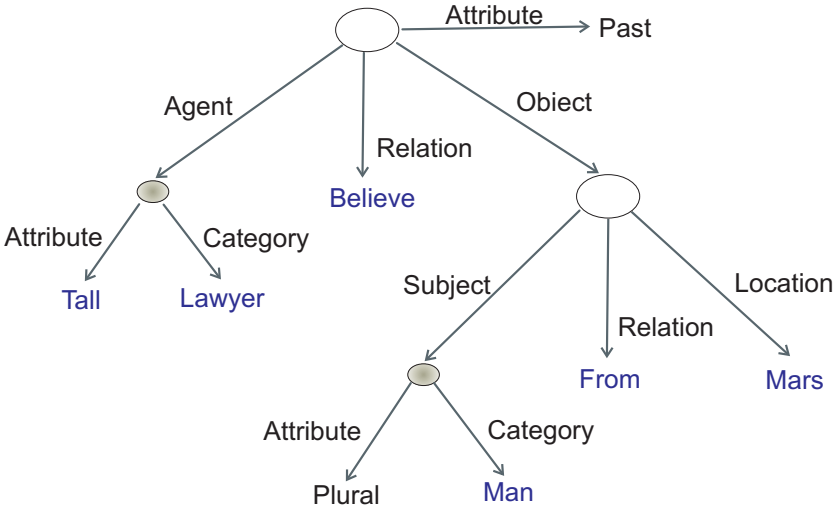
Rysunek 2.4: Graf egzystencjalny dla zdania: „Jeśli student posiada indeks, to studiuje"

$$(\forall x)(\forall y)((student(x) \wedge indeks(y) \wedge posiada(x,y) \Rightarrow studiuje(x))$$

Notacja z użyciem grafów egzystencjalnych ma siłę ekspresji powszechnie stosowanego rachunku predykatów i jest do niego transformowalna, co zapewnia użycie mechanizmów wnioskowania logiki pierwszego rzędu dla tej reprezentacji grafowej. Użycie mechanizmów logiki pierwszego rzędu do zastosowań języka naturalnego [8] [95], niesie za sobą szereg problemów wynikających zarówno z konieczności zapewnienia rozstrzygalności, skończoności wnioskowania, jak również obsługi wyjątków i sprzeczności [7] [9], które powodują ich ograniczone zastosowanie w tym obszarze.

2.2.2.3 Grafy lingwistyczne

W lingwistyce Lucien Tesniér rozwinął notację umożliwiająca graficzną prezentację powiązań między składowymi zdania. Współcześnie, podejście to używane jest do przedstawiania rozbioru logicznego i gramatycznego zdania opartego odpowiednio o części mowy i części zdania. Rozbiór taki w systemach informatycznych przetwarzających język naturalny (NLP) wspierany jest przez aplikacje automatycznie wprowadzające znaczniki, określające przynależność według funkcji jaką pełnią w zdaniu wyrazy lub związki wyrazowe. Dzięki nim dla analizowanego tekstu powstaje graf zależności między wybranymi składowymi, umożliwiający analizę gramatyczną zdania. Metoda przedstawiania zależności między składowymi zdania w postaci graficznej rozwinięta została w dwóch zaprezentowanych dalej podejściach proponowanych przez Andersona i Shanka.



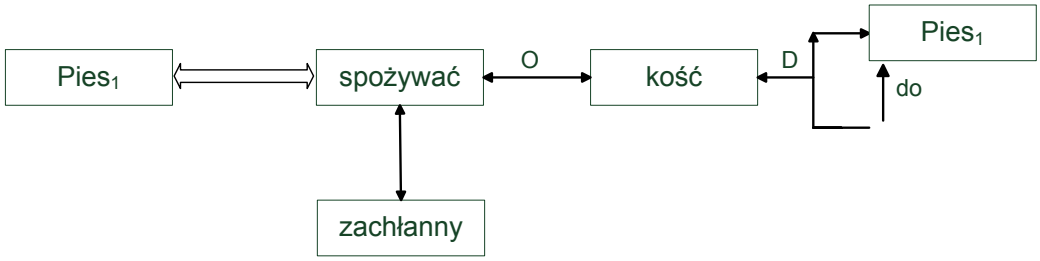
Rysunek 2.5: Przykład zdania zapisanego z użyciem sieci twierdzeń

Sieci twierdzeń

Propositional Network Model zaproponował John Anderson jako model organizacji wiedzy będący rozszerzeniem dotychczas przedstawionych systemów sieciowych, gdzie powiązania były tworzone tylko między termami [2]. Wiedza w modelu Andersona zapamiętywana jest również w postaci sieci, jednak połączenia przyjmują formę bardziej złożoną, składającą się z twierdzeń lub zdań opisujących określone powiązania. Przykład zdania „*The tall lawyer believed the men were from Mars*" reprezentowanego przez sieć twierdzeń przedstawiony został na rysunku 2.5.

Model propozycjonalny Andersona jest sposobem reprezentacji opartym o sądy: podstawową jednostką konstruującą znaczenie jest element twierdzenia, będący relacją i jej argumentami. Z tak określonych atomowych nośników znaczenia formułowana jest sieć opisująca zdanie, bądź nawet całą określoną dziedzinę wiedzy. Twierdzenie jest najmniejszą jednostką znaczenia złożoną z pojęć, które może być nośnikiem wartości prawdziwości. Rozpatrywane może być ono jako pojedyncza asercja dotycząca wiedzy – przyjmująca wartość prawdziwości *true* lub *false*.

Sieć propozycjonalna reprezentuje rodzaj abstrakcyjnego języka, w którym elementarne zdanie budowane jest poprzez predykat i jego argumenty. Argumenty, dzięki którym wyrażane są w tym modelu predykaty, wiążą się ze sobą i poprzez grupy połączonych ze sobą twierdzeń formułują sieć tworzącą obraz znaczeniowy opisywanego zdania. Zdania sieci twierdzeń nie są naturalne w znaczeniu tego słowa odnoszącego się do języka, jaki używa człowiek



Rysunek 2.6: Graf zależności konceptualnych Shanka

a raczej są nośnikiem sensu zdania. Notacja Andersona nie zachowuje struktury zdaniowej, która tracona jest w momencie dekompozycji wyrażenia na język tego modelu zachowującego jego treść semantyczną.

Koncepcja sieci sądów została rozwinięta w teorię ACT-R [3]. Jest ona kognitywną architekturą modelującą proces poznawczy w oparciu o deklaratywną pamięć długotrwałą, gdzie składowane są doświadczenia oparte o sieć wzajemnie powiązanych elementów. Współpracująca z tą pamięcią pamięć proceduralna zawiera reguły produkcji, w które organizowane są reguły wyekstrahowane z doświadczenia. W ACT-R aktywną częścią pamięci deklaratywnej jest pamięć robocza. Za jej pomocą nabywana wiedza jest przetwarzana do bardziej złożonych elementarnych jednostek informacji (*chunks*) w inkrementalnym procesie transformacji wiedzy deklaratywnej w proceduralną. Jest to model odpowiadający procesowi konsolidacji, w którym każda reguła poprzez powtarzanie się jej w procesie przepływu informacji w modelu ACT-R otrzymuje wagę określającą jej przydatność i ważność. Model ACT-R zastosowano do wyjaśnienia własności pamięci, kolejności odpowiedzi oraz przypominania, uczenia się nowych słów, uczenia się elementów programowania oraz rozumowania. Architektura wykazała umiejętności konstrukcji nowych reguł produkcji w czasie działania systemu rozwijając bardziej złożone umiejętności na bazie przeszłych doświadczeń.

Grafy zależności konceptualnych

Notacja graficznej reprezentacji zdań rozwinięta została przez Rogera Schanka. W tym podejściu do modelowania zależności pomiędzy pojęciami [76] użycie różnego rodzaju strzałek umożliwia opis różnego typów relacji, np. $\Leftrightarrow$ opisuje powiązanie agent – akcja lub strzałka ze znacznikiem *o* opisująca instancję obiektu klasy. Przykład grafu opisującego zdanie „Pies zachłannie je kość" przedstawiony został na rysunku 2.6. Występujący w zdaniu czasownik *jeść* zamieniony został w grafie akt podstawowy *spożywać*, jak również przysłówek *zachłannie* zamieniony został na przymiotnik *zachłanny*. Strzałka

ze znacznikiem d opisuje kierunek akcji, w której pojęcie *kość* jest kierowana z nieznanego bliżej miejsca do rzeczownika *pies*. Graf zależności konceptualnych rozwinięty został w tak zwane skrypty poznawcze [78] opisujące szablony elementarnych akcji – umożliwiające modelowanie czasowników i przez to opis złożonych z nich procesów. Skrypty poznawcze zawierały struktury semantyczne występujące na poziomie zdania i umożliwiały łączenie ich w większe układy. W celu uczenia lub odkrywania tych struktur automatycznie używane było wnioskowanie w oparciu o przykłady pozwalające wychwytywać szablony powiązań pomiędzy pojęciami [77].

2.2.2.4 Sieci wynikań

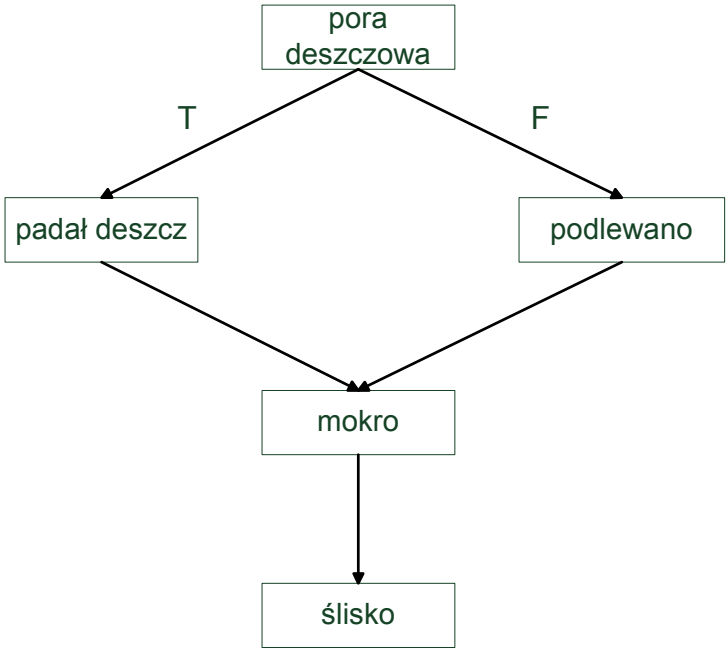
Węzły w tej klasie sieci wiązane są relacją implikacji. Pozwalają one wyrażać przekonania i związki przyczynowo–skutkowe. Inne relacje mogą być zagnieżdżone w węzłach, jednak przeważnie podczas używania mechanizmów wnioskowania są one pomijane.

Typowym zastosowaniem sieci wynikań jest opisywanie zależności przyczynowo – skutkowych. Opisywać mogą one zarówno rozgałęzienia złożone lub struktury binarne, jak również oparte na prawdopodobieństwie, w zależności od interpretacji typów powiązań i wag przypisywanych do krawędzi i węzłów.

Zamieszczona na rysunku 2.7 sieć wynikań pokazuje przykładowe przyczyny tego, że trawa jest mokra, i co za tym idzie, że jest ślisko. Każdy z węzłów reprezentuje określone twierdzenie a powiązania między nimi tworzą sieć zależności przyczynowo – skutkowych między twierdzeniami. W najprostszym wypadku krawędzie sieci obciążone są wagami reprezentującymi binarne wartości prawdziwości *true* i *false*. W rozwinięciu mogą być to liczby rzeczywiste opisujące prawdopodobieństwa przejść między poszczególnymi węzłami.

Sieci wynikań kładą w dużej mierze nacisk na operator implikacji, jednak możliwe jest modelowanie z ich użyciem wszystkich operatorów algebry Boole'a poprzez dopuszczenie koniunkcji wejść do węzła i sumy logicznej dla wyjść, których użycie pozwala na realizacje wszystkich operatorów logiki.

W zależności od przyjętego modelu obliczeniowego oraz sposobu interpretacji powiązań i wag dla takiego samego grafu, może być stosowane różnego rodzaju wnioskowanie. Wyróżnić tu można następujące podstawowe podejścia:



Rysunek 2.7: Graf wynikań opisujących ciąg przyczyn prowadzących do zjawiska „ślisko”

1. Grafowe.

   Metody wnioskowania tego typu używane są w systemach składujących asercje prawdziwości w postaci powiązań między węzłami i używające właściwości wynikających z modeli obliczeniowych stosowanych dla grafów. Zapisane w takiej sieci twierdzenia są prawdziwe *a priori* i poprzez powiązania (krawędzie) propagują się na całą sieć, w zależności od odległości od określonego węzła. Jest to typowy przypadek wnioskowania w przód – odnajdywana jest ścieżka prawdziwości pomiędzy zaistniałą przyczyną a skutkiem, który nastąpił. Sieci takie w równie prosty sposób pozwalają na wnioskowanie wstecz. Propagując wartości prawdziwościowe do węzłów, dla których taka wartość nie jest znana można uzyskać wiedzę dodatkową wynikłą, z tak realizowanego procesu dedukcji. Obok wnioskowania z posiadanej wiedzy, reprezentacja taka używana jest do weryfikacji spójności logicznej, wyszukiwania sprzeczności i miejsc, w których oczekiwane powiązania implikacyjne nie są spełnione. W sytuacji wykrycia takich niespójności struktura sieci może zostać przebudowywana, dając w rezultacie rodzaj niemotonicznego rozumowania zwanego rewizją przekonań.

2. Logiczne.

   Graf zostaje przetworzony do reprezentacji w postaci klauzul zapisanych z użyciem zredukowanej liczby operatorów rachunku predykatów (kwantyfikatora egzystencjalności i spójnika koniunkcji), a następnie użyte zostają metody wnioskowania właściwe temu sposobowi reprezentacji.

3. Oparte na prawdopodobieństwie.

   Zastępując wartości prawdy wartością jeden a fałszu zero, sieć może zostać zaadaptowana do interpretacji probabilistycznej. Dodając wartości pośrednie idea ta rozwinięta została w sieci bayesowskie pozwalająca wnioskować w oparciu o prawdopodobieństwo warunkowe [66]. Umożliwiają one dokonywanie bardziej złożonej (niż tylko binarna) analizy modelowanego problemu: dla zaprezentowanej na rysunku sieci interpretacja zero-jedynkowa jest jedynie zgrubnym przybliżeniem, nie biorącym np. pod uwagę tego czy częściej pada, czy jest susza. Zastosowanie statystyki Bayesa pozwala wprowadzić do sieci powiązań przyczynowo – skutkowych dodatkowej wiedzy, na której może być oparte wnioskowanie.

4. Indukcyjne na przykładach uczących.

   Sieci uczące są konstruowane od podstawy, bądź też rozszerzają swoją wiedzę poprzez przykłady. Nowa wiedza może zmieniać stan całej sieci poprzez dodawanie lub usuwanie węzłów i krawędzi oraz skojarzonych z nimi wag. Systemy uczące w odpowiedzi na nową dla nich informację zmieniają swoje wewnętrzne reprezentacje w taki sposób, który pozwala im efektywniej dostosować się do środowiska, w którym mają funkcjonować. Najprostszy sposób uczenia polega na zamianie nowej informacji do reprezentacji używanej przez system i dołączenia jej, bez umożliwiania żadnych jej przyszłych zmian, do aktualnej bazy wiedzy systemu.

   Najbardziej powszechnym sposobem uczenia jest wprowadzenie wag przypisywanych do węzłów lub/i relacji, które w zależności od typów sieci są różnie interpretowane np.: w sieciach Bayesa jest to prawdopodobieństwo, dla modeli symulujących struktury neuronowe są to wartości określające siłę oddziaływania pomiędzy poszczególnymi podstawowymi składowymi modelu. Najbardziej złożoną formą uczenia jest zmiana struktury całej sieci, dokonywana przez nią samą pod wpływem prezentowanych przykładów.

Istnieje szereg modeli obliczeniowych używanych do uczenia i wnioskowania w maszynach cyfrowych. Popularnym podejściem dla sieci wynikań są drzewa decyzyjne, opisane w kolejnym paragrafie bardziej szczegółowo, ze względu na użycie metody leżącej u podstawy tego modelu do zrealizowanego w pracy wyszukiwania w pamięci semantycznej.

2.2.2.5 Drzewa decyzyjne

Zasadniczą cechą systemów mających wykazywać zachowanie inteligentne jest możliwość uczenia – zdolność do poprawy zachowania na podstawie doświadczenia. W przypadku problemu klasyfikacji, do którego typowo używane są drzewa decyzyjne, uczenie jest poprawą jakości oceny przynależności obiektu do klasy dokonaną na podstawie zaprezentowanych przykładów będących zbiorem uczącym, bądź też poprawą działania systemu poprzez dokonane przez niego klasyfikacje, które zostały ocenione. Dzięki ocenie – informacji zwrotnej, możliwa jest poprawa dotychczas podejmowanych decyzji.

Jedną z metod sztucznej inteligencji posiadającej możliwość uczenia na podstawie zbioru przykładów są drzewa decyzyjne. Podobnie jak przy użyciu reguł produkcji, wiedza zawarta w drzewie decyzyjnym może zostać opisana klauzulami *if – then* [72], a proces decyzji może zostać przedstawiony w czytelnej dla człowieka postaci graficznej obrazującej kolejne etapy prowadzące do konkluzji. Ogromną zaletą drzew decyzyjnych jest sposób tworzenia reguł. Nie są one jawnie definiowane przez użytkownika, a powstają z danych: na podstawie zbioru przykładów ekstrahowane są najbardziej deskryptywne cechy dla określonego przypadku. Z dostarczonego zbioru faktów opisujących określone zagadnienie konstruowane jest drzewo, w węzłach którego podejmowane są decyzje, pozwalające nowym przypadkom przypisać etykiety określające nazwy klas kategorii.

Drzewa decyzyjne charakteryzują się dużą efektywnością programowej realizacji wynikającą z pamięciowej reprezentacji drzewiastej. Liczba możliwych testów konieczna do dokonania klasyfikacji ograniczona jest przez liczbę atrybutów. Maksymalna liczba testów konieczna do dokonania klasyfikacji wyznaczana jest głębokością drzewa decyzji – liczbą węzłów w najdłuższej jego gałęzi.

Drzewo decyzyjne jest klasyfikatorem w formie struktury drzewiastej złożonej z węzłów, z których wychodzą gałęzie prowadzące do innych węzłów, które są albo:

- liściem – wskazującym na poszukiwany element – klasę.
- węzłem decyzyjnym – wskazującym test, który musi zostać przeprowadzony. Testem określona jest funkcja na wartościach atrybutu opisanego węzłem. Typowo testy realizowane są na pojedynczym atrybucie, którego wynik pozwala przejść do kolejnego, podrzędnego węzła. Przeprowadzanie testów tylko na pojedynczych atrybutach, nie wykorzystuje możliwości użycia wiedzy o zależnościach pomiędzy nimi.

  Możliwość taką mają wielowymiarowe drzewa decyzyjne, które opierają się na testach większej liczby zmiennych w poszczególnych węzłach. Pozwalają one w sytuacjach, gdy poszczególne atrybuty zależą od siebie, na ograniczenie nadmiernego rozrastania się drzewa. Przykładem podejścia przeprowadzającego testy na więcej niż jednym atrybucie są drzewa skośne (*oblique*), gdzie testowana jest kombinacja liniowa atrybutów. W geometrycznym ujęciu separacja kolejnych podzbiorów przy użyciu testów jednej cechy rozdziela klasy za pomocą płaszczyzn równoległych do osi. Użycie do rozdzielania klas kombinacji liniowej cech pozwala na skonstruowanie praktycznie dowolnej hiperpłaszczyzny [60].

Dla testów atrybutów dyskretnych wyniki, jakie mogą przyjmować, są jednoznacznie określone przez zbiór wszystkich wartości, jakie posiadają w zbiorze testowym. Test tożsamościowy pozwala na wprowadzenie dużej liczby rozgałęzień węźle. Polega na porównaniu wyniku testu t z wartością atrybutu A:

$$t(A) = A(v) \tag{2.1}$$

gdzie:

v jest zbiorem wartości atrybutu A.

Test przynależnościowy określany jest jako:

$$t(A) = \begin{cases} 0 & \text{, jeżeli t(A)} \notin A(v) \\ 1 & \text{, jeżeli t(A)} \in A(v) \end{cases} \tag{2.2}$$

Testy atrybutów ciągłych nie mogą być przeprowadzone, w tak prosty sposób, jak dla wartości dyskretnych. Rozwiązaniem jest dokonywanie dyskretyzacji, podziału na interwały i dalej traktowanie atrybutów ciągłych, jak nominalnych. Możliwe jest również dokonywanie testów dla wartości ciągłych na wartościach progowych, z którymi porównywane

są ciągłe wartości atrybutów (2.3). Jako próg decyzji testu przeważnie rozważa się sprawdzanie środków każdego z wyznaczonych przez wartości ciągłe przedziałów.

$$t(A) = \begin{cases} 0 & \text{, jeżeli t(A)} < A(v) \\ 1 & \text{, jeżeli t(A)} \geqslant A(v) \end{cases} \tag{2.3}$$

Struktura drzewa decyzyjnego nie zawiera cykli i jest spójnym grafem skierowanym, w którym wyróżniony jest jeden wierzchołek nie posiadający krawędzi do niego wchodzących, zwany korzeniem. Gałęzie reprezentują tu poszczególne wartości atrybutów, których testy definiowane są w węzłach. Wychodząc od korzenia można się poruszać po kolejnych węzłach poprzez test atrybutu i w ten sposób przechodzić wzdłuż gałęzi odpowiadającej wynikom testów do kolejnego poddrzewa, aż do momentu dojścia do liścia będącego wartością klasyfikacji – klasą reprezentującą decyzję. Dzięki temu, można dla każdego liścia drzewa decyzyjnego wskazać ścieżkę łączącą go z korzeniem, która stanowi regułę, na podstawie której określana jest przynależność obiektu do kategorii. Reguła ta może zostać wyrażona w postaci klauzul logicznych, będących koniunkcjami kolejnych testów. Zbiór wszystkich ścieżek opisuje wszystkie reguły, które użyte są do dokonania oceny przynależności obiektu do klasy. Zamiana na reprezentację regułową pozwala przedstawić wiedzę zapisaną w drzewie w formie, która może być bezpośrednio rozumiana przez ludzi. Szczególnie przydatne jest to w sytuacji, gdy konieczne jest wytłumaczenie, dlaczego w procesie klasyfikacji zadane zostało określone pytanie. Zamiana taka zmniejszać może jednak zrozumiałość wiedzy całościowej zapisanej za pomocą drzewa, której graficzny obraz jest bardziej czytelny dla człowieka.

Drzewo decyzyjne jest indukcyjnym podejściem do uczenia klasyfikacji, wnioskowanie odbywa się tu od ogółu do szczegółu (*top – down*). By mogło zostać użyte muszą zostać spełnione podstawowe wymagania:

- Opis danych musi być za pomocą pary atrybut – wartość. Każdy z przykładów opisany jest zestawem atrybutów o określonych wartościach.
- Kategorie, do których przykłady są przypisywane, muszą być jawnie ustalone.
- Klasy muszą być rozdzielne, nie dopuszcza się wypadków, w których klasyfikowany obiekt może przynależeć do więcej niż do jednej kategorii, lub nie przynależy do żadnej.

- Zapewniona musi być wystarczająca ilość danych uczących. W uczeniu drzew decyzyjnych dopuszcza się możliwość wystąpienia w części przykładów błędów, jak również zbiór uczący może zawierać atrybuty nie posiadające określonej wartości. Kompensacja braków zostaje zrealizowana poprzez dostarczenie odpowiednio dużej liczby przykładów, co zabezpiecza przed niekorzystnym wpływem szumu.

Większość algorytmów uczenia drzew decyzyjnych jest wariantem szukania zstępującego, będącego rozwinięciem podstawowego algorytmu ID3 [71]. Do testu wykorzystane zostaje zachłanne wyszukiwanie atrybutu – algorytm wybiera najlepszy z atrybutów i nie bierze pod uwagę wcześniejszych decyzji.

Algorytm konstrukcji drzewa decyzyjnego przedstawić można w postaci pseudokodu:

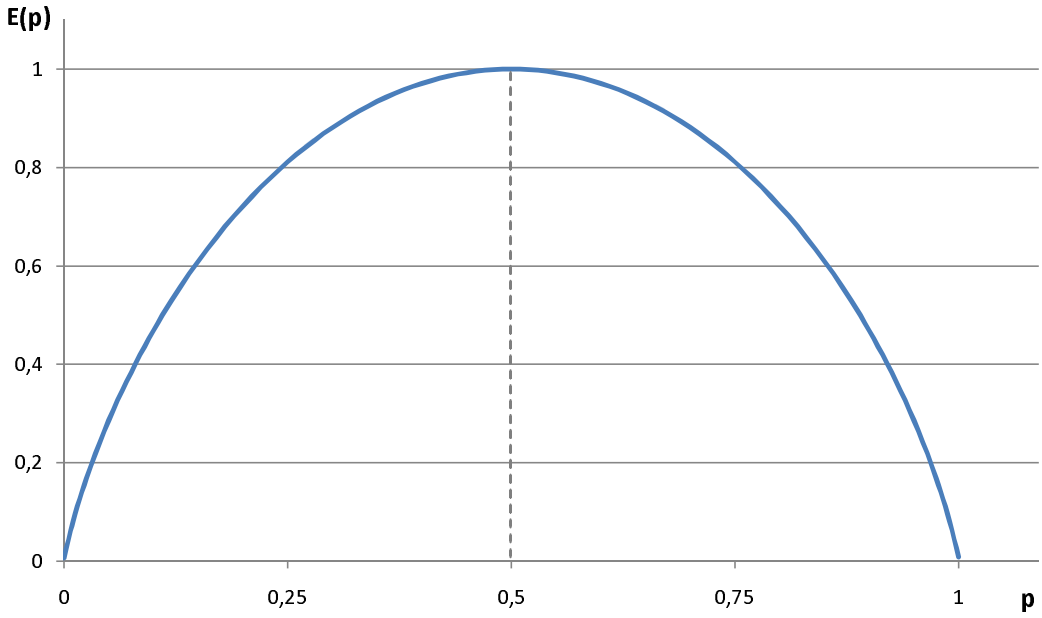
```
function buduj_drzewo (zbior_uczacy: [klasy][atrybuty][wartosci])
returns : atrybut, klasa
begin
    if spelnione_kryterium_stopu( zbior_uczacy )
        return klasa;
    else
        atrybut = wybierz_test( [atrybuty] )
        foreach wartosci[ atrybut ]
          zbior_uczacy* = reorganizuj( zbior_uczacy )
          buduj_drzewo ( zbior_uczacy* )
end
```

Kryterium stopu (funkcja `spelnione_kryterium_stopu()`) określa, czy utworzony węzeł powinien być traktowany jako końcowy liść drzewa, zawierający w swoim opisie etykietę klasy – decyzję. Najprostszym kryterium określającym, czy dany węzeł drzewa powinien być traktowany jako końcowy jest otrzymanie węzła zawierającego tylko obiekty jednej klasy. Kryterium to zapewnia poprawną klasyfikację zbioru uczącego (o ile nie występują w nim sprzeczne dane), prowadzić może jednak do nadmiernego dopasowania się klasyfikatora do danych uczących i zatracenia przez to własności generalizacyjnych. W wypadku niewystarczającego, bądź błędnego (sprzecznego) zbioru obrazów testowych zaistnieć może sytuacja, gdy kryterium to może nie być spełnione. Brak

możliwości jednoznacznego określenia, do której z klas należy dokonać przypisania, wymaga przyjęcia pewnego przybliżenia klasyfikacji. Strategią może być zwrócenie np. klasy najbardziej prawdopodobnej. Należy mieć na uwadze fakt, że zwracanie najbardziej prawdopodobnej klasy może powodować pogorszenie się jakości klasyfikacji. Częstym podejściem do określania warunku stopu jest ocena jakości podziału oparta na miarach niejednorodności węzłów. Dla węzła zawierającego te same klasy wartość takiego kryterium $I(W) = 0$, natomiast dla węzłów zawierających równoliczne klasy kryterium to przyjmuje wartość maksymalną. Przesłanką do zatrzymania algorytmu jest wtedy brak przyrostu czystości kolejnych węzłów ΔI.

Podstawową składową każdego algorytmu budującego drzewo decyzji ze zbioru danych uczących jest metoda, jaka zostaje użyta do przypisania atrybutów poszczególnym węzłom (w schemacie algorytmu ogólnie nazwana funkcją `wybierz_test`). Potrzeba zastosowania metody wyznaczającej atrybut wynika z faktu, że niektóre cechy będą dzieliły zbiór klas lepiej niż inne. Algorytm konstrukcji drzewa decyzyjnego przeszukuje zbiór wszystkich atrybutów opisujących treningowe obiekty i wybiera taki, który najlepiej rozdziela klasy od siebie. Rezultat testu atrybutu wyznacza podzbiór zbioru uczącego zbior_uczacy*, który użyty zostaje do konstrukcji kolejnego węzła (*reorganizuj*(zbior_uczacy)).

Najczęściej używane kryteria wyboru atrybutu do testu, tworzące węzeł drzewa decyzyjnego mają charakter teorioinformacyjny. Oparta na nich miara jakości atrybutu wskazuje, jak dobrze wybrana cecha dzieli dane uczące w odniesieniu do zagadnienia ich klasyfikacji. Wybór atrybutu do utworzenia węzła decyzyjnego drzewa określony jest funkcją, która przyjmuje maksimum dla najbardziej użytecznej cechy – takiej która najlepiej rozdziela klasy z obszaru wyszukiwań. W algorytmie ID3 oparty jest on na zysku informacyjnym (*IG – information gain*), jaki daje wybranie każdego z atrybutów. Zysk informacyjny wyliczany jest na podstawie entropii E w zbiorze S cech. Przyjmując, że w zbiorze są tylko przykłady pozytywne i negatywne (binarne) docelowej przynależności do klasy, entropia w zbiorze zdefiniowana jest jako 2.4.



Rysunek 2.8: Wykres entropii dla dwóch kategorii

$$E(S) = -p_p \log_2 p_p - p_n \log_2 p_n \tag{2.4}$$

gdzie:

p_p jest proporcją pozytywnych przykładów w zbiorze S,

p_n jest proporcją negatywnych przykładów cechy.

Przyjęcie takiej miary oceny przydatności cechy do separacji zbioru klas tworzy funkcję osiągającą maksimum = 1, dla cechy zawierającej równą liczbę pozytywnych, jak i negatywnych przykładów. Na rysunku 2.8 przedstawiony został wykres entropii
$E(p) = -p\ \log_2\ p - (1-p)\ \log_2\ (1-p)$. Przyjmuje on wartość maksymalną w momencie, gdy wybrana cecha dzieli zbiór klas na dwie równe części.

Uogólniając przedstawiony wzór dla binarnego przypadku cech, na dowolne dyskretne wartości, jakie może przyjmować cecha wzór 2.4 można zapisać w postaci 2.5.

$$E(S) = -\sum_{i=1}^{c} p_i\ \log_2\ p_i \tag{2.5}$$

gdzie:

p_i jest proporcją kolejnych wartości cechy, jakie może przyjmować w zbiorze S (maksymalnie c).

Używając entropii związanej z każdą z cech, wyznaczyć można efektywność każdego z atrybutów w użyciu go do podziału zbioru klas i w efekcie końcowym do klasyfikacji.

Użycie określonego atrybutu do podziału zbioru określone jest zyskiem informacyjnym $G(S, A)$ w zbiorze przykładów S dla atrybutu A, który zdefiniowany jest jako 2.6. Zysk informacyjny związany z atrybutem jest różnicą entropii układu przed dokonaniem podziału i po rozdzieleniu wybraną cechą.

$$G(S, A) = E(S) - \sum_{v \in V(A)} \frac{|S_v|}{S} E(S_v) \tag{2.6}$$

gdzie:

$v \in V(A)$ jest zbiorem możliwych wartości jakie może przyjąć atrybut A,

S_v jest podzbiorem S, dla którego atrybut A posiada wartość.

Wzór 2.6 opisuje zysk, będący różnicą entropii w oryginalnym zbiorze S i oczekiwaną wartością entropii, po podziale zbioru S z użyciem atrybutu A. Oczekiwana entropia jest ważoną sumą entropii dla każdego elementu w podzbiorze S_v. Zysk informacyjny jest w tej sytuacji oczekiwaną redukcją entropii poprzez użycie znanej wartości atrybutu A.

Proces wyboru nowego atrybutu i dzielenia zbioru przykładów treningowych jest powtarzany, aż do osiągnięcia warunku stopu na podzbiorze utworzonym z zawężenia zbioru treningowego poprzez użyte atrybuty, i z wyłączeniem już wcześniej użytych. Maksymalizacja przyrostu zysku informacyjnego niesie jednak ze sobą niebezpieczeństwo preferowania atrybutów o wielu możliwych wartościach. Problem ten wyeliminowany został w algorytmie C4.5, w którym do wyznaczenia cechy używa się współczynnika przyrostu informacji (*GR – gain ratio*), który jest rozszerzeniem przyrostu informacyjnego o czynnik normalizujący. Przyrost informacji mierzony jest tu jako różnica między poszczególnymi zbiorami przykładów w węzłach potomnych, ze względu na rozkład częstości klas. Zdefiniowany jest on następująco:

$$GR(S, A) = \frac{G(S, A)}{Iv(S, A)} \tag{2.7}$$

gdzie:

> Iv określa wartość informacyjną atrybutu A dla zbioru przykładów S, dla których ten atrybut posiada wartość (S_v).

Wielkość przyrostu informacji określa równomierność podziału zbioru S przy użyciu atrybutu A, która określona jest następująco:

$$Iv(S, A) = -\sum_{v \in V(A)} \frac{|S_v|}{S} \log_2 \frac{|S_v|}{S} \tag{2.8}$$

Inną miarą mogącą posłużyć do wyznaczenia najlepszego podziału drzewa binarnego jest indeks Giniego, zaproponowany przez Breimana [10], opisany formułą 2.9.

$$GI(S) = 1 - \sum_{i=1}^{c} p_i^2 \tag{2.9}$$

gdzie:

> p_i jest proporcją kolejnych wartości cechy, jakie może przyjmować w zbiorze S.

Indeks Giniego jest miarą zanieczyszczenia węzła przyjmującą zero, gdy w węźle są przypadki z dokładnie jednej klasy. Na bazie tej miary oparty został algorytm CART (*Classification and Regression Trees*) [10].

Obok metod wywodzących się z teorii informacji do wyboru atrybutu, który ma być testowany można użyć szeregu innych podejść. Podejścia zaprezentowane powyżej stanowią przyjęte standardy stosowane w systemach używających drzew decyzyjnych.

Jedną z alternatywnych możliwości wyboru atrybutu do testu jest metoda CHAID (*Chi-squared Automatic Interaction Detector*) [43] oparta na standardowym statystycznym teście niezależności, którego typowym przykładem jest statystyka χ^2. Wprowadza ona prawdopodobieństwo niezależności dwóch zmiennych losowych x_1 i x_2 obliczanych na podstawie obserwacji ich wartości dla losowej próby n elementów populacji. Przyjmując oznaczenia v_1 i v_2 jako liczby możliwych wartości, jakie

mogą przyjmować odpowiednio zmienne losowe x_1 i x_2, oraz f_i^1 jako liczbę wystąpień i-tej wartości cechy x_1 i analogicznie f_j^2 jako liczbę wystąpień j-tej wartości cechy x_2 (dla $j = 1 \,..\, v_2$). Oznaczajac dalej jako f_{ij} liczbę jednoczesnego wystąpienia (częstość) i-tej wartości cechy x_1 i j-tej wartości cechy x_2, a poprzez e_{ij} odpowiednią oczekiwaną liczbę wystąpień przy założeniu niezależności x_1 i x_2 (czyli *hipotezy zerowej*). Wówczas wartość statystyki χ^2 może zostać obliczona zgodnie z formułą:

$$\chi^2 = \sum_{i=1}^{v_1} \sum_{j=1}^{v_2} \frac{(f_{ij} - e_{ij})^2}{e_{ij}} \tag{2.10}$$

gdzie wartość oczekiwana e_{ij} szacowana jest następująco:

$$e_{ij} = \frac{f_i^1 f_j^2}{n} \tag{2.11}$$

χ^2 mierzy różnicę między faktycznym rozkładem poszczególnych par wartości cech x_1 i x_2, a ich rozkładem oczekiwanym przy założeniu niezależności tych cech. Większa wartość statystyki χ^2 wskazuje na większą różnicę rozkładów, przez co prawdopodobieństwo niezależności cech jest mniejsze. Prawdopodobieństwo to może zostać wyznaczone przyjmując, że wartość statystyki χ^2 jest zmienną losową o wartościach wyznaczanych różnymi próbami losowymi. Przyjmując określony poziom ufności δ można wyznaczyć posługując się tablicami statystycznymi wartość progową tej statystyki taką, że prawdopodobieństwo jej uzyskania lub przekroczenia dla niezależnych cech wynosi δ. Typowo przyjmuje się, że miedzy cechami występuje zależność jeśli wartość statystyki χ^2 przekracza próg, dla poziomu istotności 5% ($\delta = 0{,}05$) [18].

Jednym z problemów niosących użycie drzew decyzyjnych do klasyfikacji jest nadmierne dopasowanie do danych trenujących (*overfitting*). Efekt ten objawia się małym błędem klasyfikacji na zbiorze uczącym a dużym na nieznanym zbiorze, do klasyfikacji którego drzewo ma zostać użyte. Rozwiązaniem problemu nadmiernego dopasowywania jest przycinanie (*prunning*) drzew, polegające na zastąpieniu fragmentu drzewa liściem lub poddrzewem. Dzięki temu wyjściowe drzewo zostaje uproszczone. Zwiększa to szybkość klasyfikacji kosztem dokładności dla danych uczących, jak również podejście to potencjalnie może dawać lepsze efekty dla danych spoza zbioru trenującego, pomimo pogorszenia klasyfikacji na zbiorze uczącym. Przycinając drzewo dopuszcza się sytuację, że liść reprezentuje przykłady należące do różnych kategorii,

co powoduje, że klasyfikacja nie jest jednoznaczna. W sytuacji takiej naturalnym podejściem jest decyzja o przynależności do kategorii większościowej.

Podstawową przesłanką do tworzenia mniejszych drzew decyzji jest założenie, że reprezentowana przez ścieżkę w drzewie hipoteza prostsza (mająca mniej punktów decyzji – testów) jest lepsza niż długa, nawet, jeśli miałaby być mniej dokładna (z przyjętym marginesem błędu). Zasada ta funkcjonująca pod nazwą brzytwy Ockhama wynika z faktu, że długa reguła, mająca wiele stopni swobody potencjalnie jest w stanie dopasować się do dowolnych danych uczących. Jej zdolności uogólniające są jednak mniejsze, przez co bardziej prawdopodobne jest, że poprawna jest tylko dla danych uczących, na których została utworzona.

Efektowi nadmiernego dopasowania można przeciwdziałać podczas konstrukcji drzewa (przycinanie w trakcie wzrostu). Realizowane jest to poprzez odpowiednio skonstruowane kryterium stopu (w schemacie algorytmu funkcja `spelnione_kryterium_stopu`), które może być spełnione nawet, jeśli w zbiorze obiektów użytych do wyznaczenia cechy dla nowego węzła jest więcej niż jedna kategoria. Powodować to może niespójność drzewa, a w praktyce okazuje się trudne do zrealizowania ze względu na konieczność podejmowania decyzji o zatrzymaniu na podstawie rozkładu kategorii w niewielkim podzbiorze przykładów, z którego trudno pozyskać statystycznie wiarygodne wnioski [18]. Najczęściej spotykanym w praktyce podejściem jest budowanie pełnego drzewa dla zbioru trenującego, a następnie przeglądając od dołu (*bottom-up*) kolejno jego węzły zastępowanie spełniających określony warunek albo liściem, albo ich poddrzewem. Proces przycinania jest przeważnie realizacją następującego schematu:

function *przytnij_drzewo* (drzewo, zbior_przycinania)
returns : drzewo*
begin
  foreach wezel **in** drzewo
    drzewo* = drzewo - (wezel = lisc)
      `% zastąp węzeł klasą większościową`
    if *jakosc_klasyfikacji*(drzewo*, zbior_przycinania) >=
      jakosc_klasyfikacji(drzewo, zbior_przycinania)
      `% sprawdz czy nie pogorszy to klasyfikacji`
    drzewo = drzewo*
end

W zależności od ilości danych *zbior_przycinania* konstruowany jest bądź ze zbioru uczącego, bądź też na oddzielnie skonstruowanym zbiorze trenującym. Dobór tego zbioru będzie zasadniczo wpływał na końcowy kształt i jakość drzewa, jakie powstanie po przycięciu. Jeśli nie można ze zbioru uczącego wyodrębnić osobnego zbioru przycinania, decyzja o przycięciu musi zostać podjęta na jego podstawie.

Metoda przycinania znana jako *przycinanie pesymistyczne* [18] opiera się na szacowaniu błędu dla wskazanego węzła n, który ma zostać zastąpiony liściem l. Na podstawie podzbioru T zbioru trenującego wyznaczonego przez oceniany węzeł wyznaczany jest błąd klasyfikacji $e_T(n)$ poddrzewa o korzeniu w węźle n i błąd $e_T(l)$ wynikający z zastąpienia węzła liściem l. $e_T(n)$ oznacza liczbę przykładów niepoprawnie sklasyfikowanych przez poddrzewo, do liczby wszystkich przykładów T. $e_T(l)$ jest ułamkiem będącym liczbą kategorii różnej od kategorii większościowej (która jest przypisana do liścia) i liczby wszystkich przykładów w tym samym zbiorze co $e_T(n)$. Przycinanie ma miejsce w sytuacji, gdy dla przyjętego poziomu ufności μ spełniona jest nierówność 2.12.

$$e_T(l) \leqslant e_T(n) + \mu\sqrt{\frac{e_T(n)(1 - e_T(n))}{|T|}} \tag{2.12}$$

Inna metoda wyznaczenia miejsca przycięcia drzewa oparta jest na szacowaniu błędu klasyfikacji dla zmniejszonego drzewa, dopuszczająca korektę dla przyjętego progu błędu. W podejściu tym zgadzamy się na pewien całościowy błąd klasyfikacji dla zbioru uczącego i przycinamy drzewo w ten sposób, by usunąć jak najwięcej gałęzi, nie przekraczając przyjętego marginesu całościowego błędu klasyfikacji.

Często bardzo dobre efekty można uzyskać zamieniając drzewo decyzyjne na zbiór reguł i następnie przycinając te reguły. Polega to na usuwaniu warunków reguł, o ile nie pogarsza to szacowanej dokładności (określanej tak, jak miało to miejsce dla przycinania drzew).

Do bardziej zaawansowanych metod przycinania zaliczyć należy tzw. *przycinanie od środka*. Polega ono na usuwaniu nie całego poddrzewa związanego z węzłem wyznaczonym do zamiany na liść, a jedynie jego korzenia. Ponieważ zamiana testu w węźle może dla węzłów potomnych powodować, że związane z nimi testy przestają być optymalne, (a w liściach mogą ulec zmianie większościowe kategorie) podejście to wymaga reorganizacji całego poddrzewa.

Ważną kwestią podczas rzeczywistych aplikacji drzew decyzyjnych jest

postać atrybutów opisujących klasyfikowane obiekty. Wypełnione wartości dyskretnych atrybutów dla wszystkich obiektów są postulowanymi idealnymi danymi, służącymi do budowy teoretycznego modelu drzewa decyzyjnego. W rzeczywistości mamy dane zawierające przekłamania i wartości niepewne. Zdarza się również, że dla części atrybutów brak jest wartości.

W sytuacji braku części wartości dla niektórych atrybutów ze zbioru przykładów niemożliwe jest wyznaczenie optymalnego testu do dokonania podziału na podzbiory odpowiadające wynikom testów.

Podczas konstrukcji drzewa najprostszym podejściem jest pominięcie nieznanych wartości, co na ogół nie daje dobrych efektów. Stosowaną techniką jest zredukowanie wartości kryterium wyboru atrybutu do testu o współczynnik określający proporcję obiektów z nieznanymi wartościami, do wszystkich obiektów.

Najprostsza metoda uzupełnienia nieznanych wartości atrybutów (*missing data imputation*) [86] polegająca na zastępowaniu nieznanej wartości atrybutu jej średnią lub najczęściej występującą wartością przeważnie nie daje dobrych rezultatów. Lepiej sprawdzającą się techniką jest wypełnienie nieznanych wartości atrybutów na podstawie innych wartości tego atrybutu, występujących w innych obiektach klasyfikowanych, jako te same kategorie. Drugim dobrze sprawdzającym się podejściem jest zastępowanie nieznanych wartości atrybutów wartościami będącymi ułamkami tych wartości, wynikających z występowania w aktualnym zbiorze przykładów w proporcji zgodnej z częstością ich występowania.

Podczas używania drzewa decyzyjnego dla zbioru testowego, do podziału na podzbiory oparte na wynikach poszczególnych testów dla nieznanych wartości atrybutu, wprowadzić można zaliczanie do określonego podzbioru w sposób losowy z prawdopodobieństwem proporcjonalnym do liczby przykładów, bądź też można utworzyć oddzielną gałąź odpowiadającą nieznanym wynikom testu.

Istotnym problemem z drzewami decyzyjnymi jest brak prostej możliwości ich aktualizowania (douczania), bez przeprowadzania ponownego przebudowania całej struktury klasyfikacyjnej. Istniejące metody aktualizacji drzewa o nowo pozyskaną wiedzę tworzą przeważnie drzewo bardziej rozbudowane i o słabszych własnościach generalizacyjnych niż drzewo, które powstałoby z uczenia na zbiorze uczącym rozszerzonym o nową wiedzę.

Rozdział 3

Pamięć semantyczna jako system informatyczny

Sposób organizacji wiedzy przez człowieka stanowi wciąż niedościgły wzór dla programów gromadzących i przetwarzających dane. W szczególności język naturalny, jego użycie do komunikacji jest cały czas daleko poza zasięgiem możliwości współczesnych maszyn cyfrowych. Trudność w uchwyceniu tego fenomenu wynika głównie z konieczności ustanowienia kompromisu między elastycznością wyrażania myśli, a przetwarzaniem (wnioskowaniem) informacji jaką niesie [9]. Konieczność takiego kompromisu jest rezultatem formalizacji zjawiska języka, która jest niezbędna do realizacji języka naturalnego w formie programu przetwarzającego dane. Obok samej implementacji algorytmu języka, drugim zagadnieniem jest gromadzenie i aktualizacja danych lingwistycznych oraz zmiana sposobu ich przetwarzania, w zależności od kontekstu. Wymogi te powodują, że potencjalny algorytm języka staje się strukturą dynamiczną – zmieniającą się w trakcie wykonywania, co dodatkowo utrudnia jego realizację.

W historii informatyki przeprowadzanych było wiele prób implementacji kompetencji językowych w maszynie cyfrowej: głównie opierały się one na próbach ujęcia języka, jako formalnego, opartego na logice sposobu przetwarzania danych symbolicznych [41]. W ramach paradygmatu neuro-kognitywnego [92] [46] realizacja języka, jako algorytmu opiera się na implementacji struktur i metod przetwarzania informacji analogicznych do tych, które zachodzą w procesie ludzkiego poznania [27].

Semantyka w informatyce jest obecnie głównym podejściem do realizacji funkcjonalności lingwistycznych w programach. Proponowanym w pracy sposobem jej implementacji jest wyrastająca z psycholingwistycznych modeli pamięci skojarzeniowych organizacja wiedzy oparta na powiązaniach określonego typu między znaczeniami słów. Podejście to, opiera się na psycholin-

gwistycznej teorii, wprowadzającej do opisu procesu poznawczego człowieka mentalnego słownika – pamięci semantycznej, zawierającej powiązania pomiędzy pojęciami. Stanowi ona wzór do utworzenia podobnego kontenera wiedzy w systemie informatycznym.

Repozytorium powiązań miedzy pojęciami stanowić ma dla algorytmów przetwarzania danych językowych podstawę do rozumienia znaczenia określonych wyrazów. Dzięki reprezentacji tekstu opartej na powiązaniach między znaczeniami istnieje możliwość wprowadzania rozszerzenia kontekstu analizowanego pojęcia, co pozwala na efektywniejsze przetwarzanie wiedzy związanej z językiem naturalnym, np. poprawę jakości klasyfikacji tekstów [28].

Opracowanie programu komputerowego potrafiącego przechowywać w swoich strukturach danych wiedzę kontekstową wymaga przyjęcia określonego sposobu jej reprezentacji, a metody jej przetwarzania, stanowić będą podstawę użycia przez maszynę pojęć językowych. Dodanie do takiego programu interfejsu możliwie zbliżonego do języka naturalnego demonstrować będzie zdolności językowe zrealizowanego w paradygmacie neuro-kognitywnym systemu informatycznego.

Drugim, dużo trudniejszym zadaniem jest pozyskiwanie do struktur danych, mogących przechowywać informacje leksykalne wiedzy, która byłaby zbliżona do wiedzy, którą posiadają ludzie. Akwizycja wiedzy językowej wiąże się nie tylko z wprowadzaniem nowych pojęć do bazy danych, a także z wiedzą o ich znaczeniu, przejawiająca się w użyciu, ich weryfikacją oraz aktualizacją już zapisanych informacji [68].

W tym rozdziale przedstawione zostało podejście do komputerowej realizacji pamięci semantycznej, utworzonej w oparciu o jej biologiczny wzorzec opisany w rozdziale 1. Termin pamięć semantyczna oznacza tu struktury danych umożliwiające składowanie wiedzy w komputerze w postaci sieci powiązań między pojęciami.

W kolejnych paragrafach przedstawiony został sposób reprezentacji wiedzy przyjęty do organizacji pojęć językowych w struktury kontekstowe, algorytm operujący na takich danych, jego przykładowe zastosowania i metody pozyskiwania danych kontekstowych do pamięci semantycznej, w której są one składowane.

Demonstracja użycia wiedzy językowej składowanej w powiązaniach między pojęciami, zrealizowana została poprzez algorytm wyszukiwania obiektów z użyciem cech przez nie posiadanych, a dołączone mechanizmy parsowania i generowania zdań w języku naturalnym stanowią lingwistyczny interfejs pamięci semantycznej. Pozwalają one na demonstrację możliwości komunikacji za pomocą uproszczonych naturalnych konstrukcji językowych między człowiekiem, a semantycznym repozytorium.

Obszary zastosowań wiedzy semantycznej zorganizowanej jako typowane

powiązania między znaczeniami są szerokie. Mechanizm eksploracji w takiej składnicy wiedzy wspierać może organizację informacji w zasobach tekstowych, jak również może wspomagać procesy diagnostyczne. Użyty również może zostać w systemach interakcji maszyny cyfrowej z człowiekiem w języku naturalnym, gdzie pamięć semantyczna stanowi podstawę do rozumienia znaczenia pojęć języka naturalnego. Rozumienie języka oparte na modelach mentalnych, które łączą funkcje rozumienia i wnioskowania są próbą uchwycenia fenomenu języka poprzez realizację go jako algorytmu kognitywnego [40]. Zastosowanie takiego podejścia przynieść ma przede wszystkim poprawę przetwarzania lawinowo rosnącej ilości informacji, która znajduje się głównie w postaci tekstów zapisanych w języku naturalnym, a w długofalowej perspektywie znaczną poprawę komunikacji człowieka z maszyną.

3.1 Reprezentacja wiedzy dla pamięci semantycznej

Jednym z najbardziej popularnych sposobów opisu danych kontekstowych jest wiązanie ze sobą pojęć w trójki formujące atomowe jednostki wiedzy. Trójka taka jest ogólnym sposobem reprezentacji wiedzy, typowo używanym do zapisu wartości atrybutów dla obiektu (OAV – Object-Attribute-Value). Możliwy jest jednak szereg innych sposobów jej interpretacji np. obiekt – typ powiązania – cecha (RDF – Resource Description Framework). Koncepcja ta stanowi podstawę realizacji sieci semantycznych [80], gdzie zbiór wzajemnie połączonych ze sobą trójek tworzy sieć powiązań miedzy pojęciami językowymi, opisującymi wybrane zagadnienie. Zbiór pojęć i powiązań między nimi nazywany jest często ontologią. W pracy pojęcie ontologii odnosić się będzie do sieci semantycznej ζ, która może być przedstawiona jako graf powiązań między znaczeniami bądź wyrazami utworzony przez atomowe jednostki reprezentacji wiedzy (opisane dalej). Przestrzeń semantyczna Ψ określana będzie przez jej algebraiczną reprezentację w postaci macierzy opisującej wszystkie powiązania między występującymi w niej pojęciami.

Notacja wykorzystująca powiązania między trójkami tworzącymi sieci semantyczne pozwala na wygodną wizualną ich prezentację w postaci grafu, którego węzłami są pojęcia, a typy relacji związane są z krawędziami je łączącymi. Reprezentacja taka zbliżona jest do sposobu, w jaki człowiek opisuje zagadnienie przy użyciu wiążących się z nim pojęć. Możliwość graficznego przedstawienia wiedzy zapisanej przy użyciu tej metody reprezentacji jest bardzo użyteczna w sytuacji, gdy konieczne jest przeglądanie bazy wiedzy systemu. Podejście oparte na wiązaniu ze sobą elementarnych pojęć z obsza-

ru opisywanego zagadnienia pozwala wyrażać bardziej złożone twierdzenia, które mogą być przedstawione w postaci zdania w języku naturalnym, jak również interpretowane przez maszyny. Sieć typowanych powiązań między pojęciami umożliwia również zachowanie ekonomii kognitywnej – poprzez proceduralną realizację określonych typów relacji nie wymaga składowania pełnej wiedzy w bazie, a pozwala ją wywieść w procesie przetwarzania danych.

Bezpośrednia reprezentacja za pomocą trójek (jak i bardziej wyszukanych formalizmów) nie nadaje się jednak do przeprowadzania złożonych obliczeń matematycznych (umożliwiających np. zastosowanie metod drążenia danych - *Data Mining*) na danych zapisanych w dużej sieci, z różnymi typami powiązań, dla której pojawiają się trudności z rozstrzygalnością wiedzy oraz skończonością obliczeniową [9]. Wnioskowanie w obrębie samej sieci powiązań oparte jest na własnościach grafów i sprowadza się głównie do liczenia różnorodnie realizowanego podobieństwa między węzłami. Brak jest tu mechanizmu pozwalającego na przeprowadzanie, bardziej złożonego niż tylko opartego na odległościach między węzłami, wnioskowania wynikającego z przyjętego sposobu reprezentacji wiedzy. Podejście oparte na notacji trójkowej nie zawiera również możliwości opisu wiedzy częściowej oraz nie oferuje możliwości uczenia. Jednym ze sposobów zwiększenia użyteczności wiedzy zapisanej w sieci semantycznej ζ jest zamiana na odpowiadającą jej reprezentację geometryczną – przestrzeń semantyczną Ψ. Podejście takie pozwala na dokonywanie szeregu obliczeń algebraicznych, a poprzez dodanie możliwości aktualizacji stopni swobody opisujących obiekt w przestrzeni cech umożliwia zrealizowanie uczenia.

Reprezentacja geometryczna Ψ obiektu sieci semantycznej ζ powstaje z transformacji każdego węzła grafu pojęć w punkt w n-wymiarowej przestrzeni cech[1] tak, że każdy z obiektów opisywanej dziedziny reprezentowane jest przez zbiór cech. W podejściu tym obiekt (opisany pojęciem) jest reprezentowany wektorem zawierający powiązania z charakterystycznymi dla niego cechami. Wektor taki nazywany będzie CDV (***C**oncept **D**escription **V**ector*). Reprezentacja obiektów (pojęć językowych) przy użyciu CDV, w zależności od potrzeb może być rozwinięta w szereg kierunków: przydatność cechy do określenia obiektu można opisać za pomocą funkcji określającej jej siłę deskryptywności,jak również możliwa jest reprezentacja obiektu jako zbioru cech opisanych poprzez wektory powiązań z innymi pojęciami. W sytuacji takiej opisem obiektu będzie n-wymiarowa macierz cech x_i, z których każda opisana jest macierzą m_i-wymiarową złożoną z powiązań tej cechy z obiektami, z którymi jest ona powiązana.

[1] gdzie n jest liczbą wszystkich cech

Do realizacji reprezentacji obiektów pamięci semantycznej w przestrzeni semantycznej użyto wektorów opisu CDV, składających się z elementarnych jednostek wiedzy nazwanych vwCRK (vw–***w****eights,* ***C****oncept,* ***R****elation,* ***K****eyword*), opisujących poszczególne cechy składowanych w niej obiektów. Atom vwCRK jest rozszerzeniem notacji RDF, wprowadzającym do podstawowego podejścia wagę, pozwalającą określić pewność wiedzy oraz wagę umożliwiającą tę wiedzę rozmyć[2]. Poszczególne elementy wchodzące w skład atomu wiedzy vwCRK oznaczają:

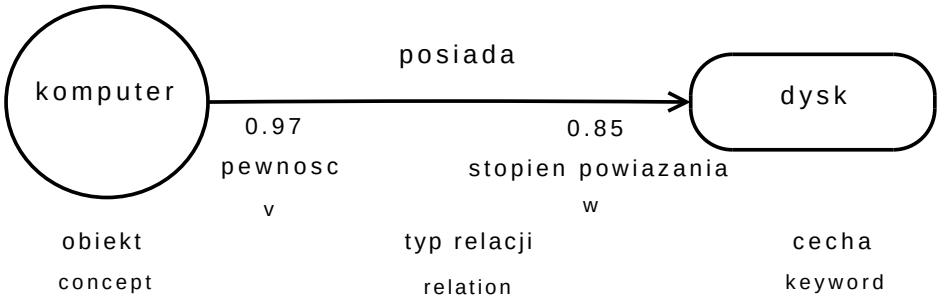
- v – liczba rzeczywista z przedziału $\langle 0, 1\rangle$ opisująca pewność posiadanej wiedzy – wartości bliższe 1 wskazują na większą pewność odnośnie prawdziwości powiązania.
- w – liczba rzeczywista z przedziału $\langle -1, +1\rangle$ opisuje stopień powiązania (jak bardzo cecha jest typowa dla opisu pojęcia). Wartości dodatnie określają, na ile określona cecha wiąże się określonym typem relacji z opisywanym obiektem. Analogicznie ujemne wartości określają, jak bardzo cecha nie jest typowa dla pojęcia. Użycie wartości ciągłych pozwala wyrazić powiązania typu np.: „przeważnie nie" lub „często", czy też „nigdy".
- C – opisywany obiekt.

  Jeśli słowo „obiekt" występuje w znaczeniu elementu pamięci semantycznej używane jest zamiennie ze słowem „pojęcie", które jest jego reprezentantem językowym (słowem bądź znaczeniem) w sieci semantycznej ζ.
- R – typ relacji na zasadzie, której obiekt jest powiązany z cechą.
- K – cecha wyrażająca określoną właściwość obiektu.

Przykład atomu wiedzy wyrażonej za pomocą notacji vwCRK przedstawiono na rysunku 3.1. Opisuje ona elementarne zdanie, jakie może zostać zapisane w pamięci semantycznej: „komputer posiada przeważnie ($w = 0,85$) dysk". Wiedza ta posiada wysoki współczynnik pewności ($v = 0,97$).

Wartość wagi w pozwala określać typowość powiązania obiektu z cechą, zarówno w stronę pozytywną (np. posiada) jak i negatywną (np. nie posiada). Waga v określa pewność takiej wiedzy – czy informacja zapisana w strukturach danych pamięci semantycznej jest zweryfikowana. Potwierdzenie wiedzy odbywa się w procesie interakcji systemu z człowiekiem. Np. podczas opisu

[2]rozmycie w sensie logiki rozmytej, pozwalającej na wprowadzenie wartości pośrednich między klasyfikacjami prawda – fałsz [103]



Rysunek 3.1: Elementarna jednostka wiedzy vwCRK

pojęcia *bocian* cecha *czarny* będzie posiadała wartości w okolicy 0,5 (rzadko), natomiast cecha *biały* w okolicy wartości 1 (określającej tę cechę jako typową). W przypadku tej asercji wartość v narastać będzie do 1 w procesie interakcji z ludźmi – im więcej potwierdzeń każdego z atomów wiedzy zostanie uzyskana, tym bardziej pewność wiedzy, którą opisuje zbliżać się będzie do 1.

Weryfikacja atomów wiedzy tworzących sieć semantyczną odbywa się przez zastosowanie ich w algorytmie wyszukiwania kontekstowego (paragraf 3.2), dzięki któremu dokonana zostanie ocena jakości całej sieci semantycznej (paragraf 5.2.2).

W przestrzeni semantycznej Ψ obiektów reprezentowanych przez CDV, złożonych z atomów wiedzy vwCRK, poprzez wartości wag v i w możliwe jest zrealizowanie uczenia – modyfikacje tych współczynników pozwalają dostroić bazę wiedzy oraz umożliwiają wprowadzenie do niej wiedzy rozmytej. Podejście takie jest przez to bardziej elastyczne niż trójki OVA czy RDF.

Sieć powiązań między pojęciami ζ złożona z vwCRK może zostać przetransformowana w odpowiadający jej model geometryczny (przestrzeń semantyczną Ψ), gdzie obiekty reprezentowane są jako CDV. Dzięki temu wiedza semantyczna zapisana w postaci grafu powiązań może zostać poddana obróbce z użyciem algorytmów drążenia danych (*Data Mining*). Umożliwia to wydobywanie dodatkowej wiedzy, nie zapisanej jawnie w sieci skojarzonych ze sobą pojęć, przez co przetwarzanie takie może zostać uznane za pewną formę wnioskowania.

Zamiana sieci semantycznej ζ na jej geometryczny odpowiednik Ψ polega na utworzeniu z vwCRK wektorów CDV, dla każdego z obiektów opisanej siecią semantyczną dziedziny. Przy tej zamianie reprezentacji wiedzy określone przetwarzanie wybranych typów relacji (związane z procedurą je realizującą) pozwala na uzyskiwanie dodatkowej wiedzy dla poszczególnych wektorów CDV. Obsługa typów relacji realizowana jest przez procedury wykonujące

określone działania na wektorze CDV, przy napotkaniu ustalonych typów relacji. W zrealizowanym systemie pamięci semantycznej zaimplementowano przetwarzanie czterech podstawowych typów relacji semantycznych:

1. ***is_a*** – W związku z nierozdzielaniem w pamięci semantycznej klas od obiektów relacja tego typu umożliwia wprowadzenie zawierania się pojęć, co pozwala na hierarchizację i dziedziczenie cech. Wystąpienie tego typu relacji pomiędzy obiektami sieci semantycznej dokonuje przepisania wszystkich cech z pojęć nadrzędnych do podrzędnych ze współczynnikiem pewności v równym iloczynowi współczynnika pewności związanego z powiązaniem *is_a* i wartościami określającymi pewność dla cech związanych z obiektem nadrzędnym. Pozwala to na przekazywanie cech wynikających z hierarchicznego ułożenia danych ze współczynnikami pewności wynikłymi zarówno z pewności powiązania tworzącego taksonomię, jak i z pewności powiązań w nadrzędnych wektorach CDV, od których pochodzą propagowane cechy.

2. ***similar*** – Podobieństwo. Zbiory cech wektorów CDV dwóch obiektów powiązanych relacją *similar* zostają sklejone ze sobą w ten sposób, że nie istniejące w zbiorze cechy zostają dołączone z drugiego zbioru, ze współczynnikiem równym wartości pewności powiązania v. Dla wartości współczynnika $w = 1$ realizowany jest typ relacji „*same*", implementujący tożsamość między obiektami pamięci semantycznej.

3. ***excludes*** – Wykluczanie. Wszystkie cechy, które nie występują między CDV obiektów powiązanych typem relacji *excludes*, zostają przepisane ze wskazanym współczynnikiem pewności. Natomiast wartości opisujące typowość powiązania otrzymują wagę w równą wadze powiązania posiadanego przez pojęcie przeciwne, pomnożoną przez wartość -1.

4. ***entail*** – Wynikanie – implikacja. Typ relacji pozwalający wywieść dodatkowe powiązania między cechami. Wystąpienie typu relacji *entail* pozwala na wprowadzenie dodatkowych cech do opisu obiektów, wynikłych z powiązań między cechami. Do CDV obiektu zawierającego cechę będącą w relacji *entail* zostaje dopisana druga cecha, występująca w tej relacji z wartością współczynnika pewności v równą pewności związanej z relacją *entail*.

Procedury przetwarzania typów relacji pozwalają na uzyskanie dodatkowej wiedzy (nowych cech) podczas tworzenia wektorów CDV, reprezentujących obiekty sieci semantycznej. Wpływ wprowadzenia interpretacji typów relacji na ilość wiedzy (średnią liczbę cech w wektorach CDV) dyskutowany jest w paragrafie 5.2.

Współczynnik pewności v umożliwia weryfikację wiedzy, zapisanej w bazie i wiedzy, która może zostać z niej wywiedziona. Odbywa się to poprzez proces interakcji z użytkownikiem. Typ relacji i jego waga w określają zasadę na jakiej obiekt się wiąże z cechą. Oba współczynniki v i w są wypadkową składowych, które mają wpływ na ich ostateczną wartość.

Pewność wiedzy v wyznaczana jest z trzech składników będących źródłem pochodzenia informacji: od użytkownika, zapisanej bezpośrednio w bazie i wiedzy wywnioskowanej. Podejście takie pozwala określić, jakiemu rodzajowi wiedzy bardziej ufamy. W implementacji pamięci semantycznej przyjęte zostało, że jeśli nie wiemy nic o powiązaniu obiektu z cechą, użyta zostaje wartość wagi w, którą udało się wywieść z już istniejących danych, a pewność v takiej wiedzy jest równa najmniejszej wartości pewności atomu wiedzy, który został użyty do wnioskowania. Jeśli w pamięci semantycznej posiadana jest już wiedza odnośnie jakiegoś obiektu i jego cech, używana wiedza jest uśrednioną sumą wiedzy zapisanej w bazie i wiedzy pochodzącej od użytkowników. Informacje uzyskane od użytkowników poprzez proces konsolidacji (wywoływany przez administratora) przechodzą w wiedzę bezpośrednio zapisaną w bazie wiedzy (proces konsolidacji opisany został w rozdziale 5 opisującym użycie aktywnych dialogów).

Współczynnik pewności dla atomowej jednostki wiedzy wyznaczany jest zgodnie z formułą 3.1.

$$v = \begin{cases} v_d & \text{, jeżeli } w = null \\ (1-\alpha)\ v_a\ +\alpha\ v_u & \text{, jeżeli } w \neq null \end{cases} \tag{3.1}$$

gdzie:

$v_d = \min_i \{\mathrm{val_v}(CDV_i)\}$ – jest wartością pewności wiedzy wynikłej z przetwarzania informacji zapisanej w bazie wiedzy. Dla nieznanej wartości wagi w, pewność wiedzy wydedukowanej v_d jest najmniejszą wartością pewności v dla ciągu wag w reprezentujących trójki wiedzy, przez które zaszło wnioskowanie w zbiorze wektorów CDV. Wnioskowanie realizowane jest przez wykonanie procedur związanych z określonymi typami relacji wektora CDV, pozwalającymi wyznaczyć wartość wagi w dla nieznanej cechy. Współczynnik v_d używany jest w sytuacji, gdy brak jest zapisanej w bazie wiedzy informacji na temat powiązania między obiektem a cechą.

α – współczynnik określa modyfikację wiedzy zapisanej w bazie wiedzy systemu (v_a) przez wiedzę uzyskaną w interakcji z użytkownikiem (v_u). Podejście takie pozwala na bieżąco (bez procesu konsolidacji) weryfikować informację zapisaną w bazie wiedzy z informacją nowo pozyskaną.

Przyjęto, że współczynnik α określony jest funkcją liczby głosów pochodzących z interakcji systemu z człowiekiem tak, że w sytuacji dużej ilości wiedzy płynącej od użytkowników ma ona większy wpływ na wypadkową wiedzę systemu.

$$\alpha = \tanh(votes) = \frac{2}{1 + e^{-\beta\ votes}} \tag{3.2}$$

gdzie:

votes jest różnicą liczby głosów użytkowników („za" i „przeciw"), która jest wypadkową wskazującą wartość pewności powiązania określonej cechy z obiektem,

parametr β określa szybkość, z jaką wiedza pochodząca od użytkowników zaczyna „wypierać" niepoprawną wiedzę zapisaną w bazie. Przyjęty został on arbitralnie na 5.

Wprowadzony współczynnik pewności wiedzy pozwala na obsługę wyjątków – sytuacji, w których wiedza systemu wynikła z dedukcji jest sprzeczna z wiedzą uzyskaną w procesie interakcji z użytkownikiem, bądź też jest sprzeczna z wiedzą zakodowaną bezpośrednio w bazie wiedzy systemu. Współczynnik v pozwala wywiedzioną wiedzę przyjmować z określonym prawdopodobieństwem, dzięki czemu możliwe jest jej skorygowanie, bądź przez bezpośredni wpis do bazy wiedzy lub poprawę na drodze interakcji z użytkownikiem.

Sposób obliczenia wartości w określającej siłę powiązania opisuje formuła 3.3, wyznaczająca stopień powiązania obiektu i cechy, jako ważoną sumę trzech składowych.

$$w = \frac{(\alpha w_d + \beta w_u + \gamma w_a)}{\alpha + \beta + \gamma} \tag{3.3}$$

gdzie:

w_d – wartość powiązania uzyskana z bazy wiedzy systemu, α – współczynnik określający jak duży wpływ ma ta składowa na całościową wartość wagi. Przyjęto $\alpha = 0{,}25$.

w_u – średnia wartość powiązania uzyskana z głosów użytkowników, $\beta = 0{,}33$.

w_a – wartość powiązania zakodowana bezpośrednio w bazie wiedzy systemu, $\gamma = 0{,}42$.

dla przyjętych arbitralnie współczynników $\alpha + \beta + \gamma = 1$

Algorytm wyznaczania współczynnika w dla cechy k obiektu c reprezentowanej wektorem CDV przedstawiony został w pseudokodzie:

function *calcW*(wektorCDV[], cechaK)

returns w

begin

`% wyznaczenie wagi` w_d

foreach cecha **in** wektorCDV

wektorCDVcechy = bazaWiedzy[cecha] `% pobierz wektor CDV dla cechy traktując ją jako obiekty`

case cecha[typ_relacji]:

wektorCDV += wektorCDVcechy `%przepisz elementy wektora CDV cechy zgodnie z interpretacją typu relacji`

w_d = avg(w[cecha]) `% dla powtarzających się cech uśrednij wartość wagi` w

`% wyznaczenie wagi` w_u

%uśrednij wagi w cechy dla obiektu c z bazy interakcji z użytkownikiem (wektorCDV_użytkownika)

w_u = avg(wektorCDV_użytkownika[c][cecha])

`% wyznaczenie wagi` w_a

w_a = wektorCDV[cechaK]

if w_a = **null** $w_a = 0$

$w = F(w_d, w_u, w_a)$ % zgodnie z 3.3

end

W powyżej zaprezentowany sposób zostaje wyznaczona wartość typowości powiązania (waga w), dla każdej kolejnej cechy wektora CDV reprezentującego zadany obiekt.

Na żądanie administratora pamięci semantycznej przeprowadzany jest proces konsolidacji. W jego rezultacie wartości współczynników wiedzy: pewności v_u i typowości w_u, pozyskanej w procesie interakcji z użytkownikiem i wiedzy dotychczas posiadanej przez system zostają uśrednione i zapisane jako wiedza systemu – bezpośrednio w bazie wiedzy, jako v_a i w_a. Po zrealizowaniu tej procedury wiedza pochodząca od użytkowników zostaje wyczyszczona i proces weryfikacji wiedzy zapisanej w pamięci semantycznej jest przeprowadzany dalej, dla nowo ustalonych wartości. W perspektywie kilkukrotnego powtórzenia tego procesu dane w pamięci semantycznej zostają na tyle uśrednione, że przy kolejnych wywołaniach procedura konsolidacji zmieniać je będzie bardzo nieznacznie.

Ważnym procesem związanym z konsolidacją jest użycie algorytmu generalizacji pozwalającego zachować ekonomię kognitywną. Usuwa on cechy powszechnie występujące przy szczegółowych obiektach i zapisuje te powiązania w relacji z bardziej ogólnymi pojęciami. W szczególności taki proces nie może zostać przeprowadzony automatycznie i wymaga interakcji z użytkownikiem (administratorem) potwierdzającym, czy zadana cecha jest na tyle ogólna, by mogła zostać przeniesiona na wyższy poziom w taksonomii. Sytuacja ta jest opisana w paragrafie dyskutującym użycie aktywnych dialogów do pozyskiwania wiedzy (rozdział 5).

3.2 Algorytm wyszukiwania kontekstowego

Zamiana reprezentacji wiedzy z postaci sieci semantycznej ζ na przestrzeń semantyczną Ψ – odpowiadający jej wektorowy opis obiektów reprezentowanych pojęciami w przestrzeni cech, umożliwia utworzenie grafu decyzji, dzięki któremu można zrealizować wyszukiwanie oparte na kontekście.

Powszechnie używane metody wyszukiwania oparte są na słowach kluczowych. Zawężają one przeszukiwany obszar poprzez dopasowywanie podanych przez użytkownika wyrazów do znanych systemowi wyszukiwawczemu obiektów, reprezentowanych przez ciągi znaków. Sytuacja taka ma miejsce np. w wyszukiwarkach internetowych, gdzie określenie podzbioru wyszukiwanych dokumentów odbywa się poprzez odnalezienie wskazanych słów w treści stron WWW. Użytkownikowi końcowemu zwrócony zostaje zbiór dokumentów spełniających kryterium dopasowania słów kluczowych, który uporządkowany jest wedle określonej miary rankingowej [47]. W efekcie zastosowania takiego podejścia, dla dużej ilości danych użytkownik otrzymuje ogromną

liczbę wyników spełniających warunki określone przez kryterium wyszukiwania. Dalsze przetwarzanie otrzymanego zbioru odbywa się zazwyczaj poprzez zawężanie go oparte na dopasowaniu kolejnych słów kluczowych. Wadą tego podejścia jest założenie, że w wypadku poszukiwania odpowiedzi na pytanie odpowiedź musi znajdować się w słowach tworzących zapytanie, co niekoniecznie musi zawsze mieć miejsce.

Zadaniem wyszukiwania kontekstowego jest odnalezienie obiektu nie przez dopasowanie słów kluczowych do pojęcia, lecz poprzez wskazywanie cech. Określanie obiektu w pamięci semantycznej odbywa się poprzez stopniowe doprecyzowywanie cech z nim związanych, a nie tylko przez dopasowywane ciągu znaków do słów opisujących przedmiot poszukiwania. Podejście takie może być potencjalnym rozszerzeniem dla wyszukiwarki internetowej opartej na dopasowywaniu słów kluczowych, które ma miejsce w pierwszej fazie a następnie wprowadzeniu kolejnego etapu szukania, w którym wyszukiwany obiekt doprecyzowany jest z użyciem cech [30].

Algorytm wyszukiwania kontekstowego wymaga utworzenia ontologii, w której wyszukiwane obiekty opisane są pojęciami określającymi ich cechy. Występujące w niej pojęcia opisują powiązania między obiektami i ich cechami z eksplorowanego obszaru wiedzy. Do przeszukiwania zasobów internetu podejście takie wymagałoby ogromnej bazy wiedzy (pokrywającej cały język: nie tylko potoczny, ale zawierający również specjalistyczne dziedziny). Brak takiej ontologii w znacznym stopniu ogranicza zastosowanie proponowanego podejścia na dużą skalę.

Wyszukiwanie kontekstowe może jednak znaleźć zastosowanie dla mniejszych obszarów wiedzy ludzkiej, o dobrze zdefiniowanej semantycznie dziedzinie. Przykładem może być tu np. wspomaganie decyzji medycznych, gdzie rezultatem wyszukiwania byłaby jednostka chorobowa, odnaleziona poprzez wskazanie określonych symptomów. Innym zastosowaniem takiego wyszukiwania jest demonstracja elementarnych zdolności językowych systemu informatycznego, wykorzystującego wiedzę o znaczeniu przejawiająca się np. w grach słownych.

Algorytm wyszukiwania kontekstowego oparty jest na metodzie znanej z budowy drzew decyzyjnych z użyciem zysku informacyjnego. Wskazuje on przydatność każdej z cech do klasyfikacji obiektów opisanych z ich użyciem. Zasadniczą różnicą, w prezentowanym tu podejściu jest odmienna przeszukiwana struktura – nie jest nią drzewo lecz graf (zakłada się, że każdy węzeł może być klasą decyzyjną), którego węzłami są pojęcia językowe z wybranej dziedziny.

Podczas wyszukiwania w pamięci semantycznej dopuszcza się możliwość, że informacja dotycząca obiektów opisanych w grafie decyzji jest niepełna, jak również może być niepoprawna. Skutkować to może błędnym rezultatem

wyszukiwania, na podstawie którego przeprowadzana jest reorganizacja bazy wiedzy.

Trzecią różnicą, w stosunku do klasycznego użycia drzew decyzyjnych jest dopuszczenie możliwości wystąpienia podczas procesu wyszukiwania błędnych odpowiedzi (niezgodności miedzy rezultatem testu a danymi zapisanym w drzewie).

Wyszukiwanie w pamięci semantycznej jest poruszaniem się po grafie cech opisujących obiekty. Odbywa się ono poprzez wybranie cechy najlepiej dzielącej przestrzeń wyszukiwanych obiektów i poddaniu jej do testu, poprzez sformułowanie zapytania do użytkownika, na ile poszukiwany obiekt wiąże się z zadaną cechą. Przyjęto, że na zadane pytania możliwe są następujące odpowiedzi: „tak", „nie", „nie wiem", „czasami" i „często", aczkolwiek w zależności od potrzeby zastosowań, możliwe jest zadawanie pytań innego rodzaju (np.: o wartości należące do przedziałów, lub ze zbioru wartości, jakie może przyjmować cecha). Udzielone odpowiedzi na pytania, zadawane przez algorytm wyszukiwania, pozwalają na zawężenie zbioru przeszukiwanych obiektów w stronę tych, które są najbardziej prawdopodobne. Najlepsze cechy do utworzenia zapytania wyznaczane są poprzez największy zysk informacyjny związany z każdą z nich, określający, które z cech najlepiej rozdzielają zbiór obiektów. W przestrzeni M obiektów (o) i N cech (c) dla każdej cechy zysk informacyjny można wyznaczyć zgodnie z formułą 3.4.

$$IG(c_n) = -\sum_{i=1}^{M} p(o_i) \log p(o_i) \tag{3.4}$$

gdzie:

$$p(o_i) = \frac{\text{abs } (w_{in})}{M} \tag{3.5}$$

gdzie:

w_{in} – jest wartością wagi łączącej obiekt i z cechą n.

W powyższy sposób zostaje wyznaczony zysk informacyjny dla każdej cechy. Największa jego wartość wskazuje cechę, o którą najlepiej jest się zapytać tak, by najkorzystniej podzielić przestrzeń obiektów – wybrać te, które posiadają określone powiązanie z cechą a odrzucić te, które nie są z zgodne z rezultatem testu. W perspektywie aplikacji algorytmu do wielowymiarowych przestrzeni cech, trzeba mieć na uwadze fakt, że cech mających największy zysk informacyjny może być więcej niż jedna. Powoduje to konieczność wprowadzenia dodatkowego kryterium wyboru, które może być oparte np. na wiedzy o rozkładzie prawdopodobieństwa wystąpienia obiektów, jako wyniku

klasyfikacji (5.8) lub popularności występowania w języku słowa opisującego cechę (4.1).

W algorytmie wyszukiwania kontekstowego uzyskane od użytkownika odpowiedzi tworzą wektor odpowiedzi $ANSW$, który używany jest do obliczenia odległości d pomiędzy wszystkimi obiektami (opisanymi wektorami CDV) z eksplorowanej przestrzeni semantycznej Ψ. Odległość d pomiędzy wektorem odpowiedzi użytkownika a wektorem CDV opisującym poszczególne obiekty w pamięci semantycznej wyznacza prawdopodobieństwo, dla każdego z poszukiwanych pojęć. Dla obiektu o reprezentowanego pełnym wektorem wiedzy CDV odległość ta liczona może być jako miara euklidesowa (3.6)

$$d_o = d(CDV, ANSW) = \sqrt{\sum_{i=1}^{K} (CDV_i - ANSW_i)^2} \tag{3.6}$$

gdzie:

K jest długością wektora ANSW, określającą krok wyszukiwania (liczbę zadanych do testów pytań).

Inną możliwością jest określenie odległości d miarą kosinusową 3.7, będącą znormalizowanym iloczynem skalarnym dwóch wektorów. Wartość kosinusa kąta pomiędzy wektorami opisuje ich podobieństwo. Jest to najczęściej stosowana miara w systemach wydobywania informacji (IR – *information retrieval*), uniezależniająca podobieństwo od wielkości dokumentów oraz od liczby przypisanych im termów [69].

$$d(CDV, ANSW) = \frac{\sum_i CDV_i ANSW_i}{\sqrt{\sum_i CDV_i^2}\sqrt{\sum_i ANSW_i^2}} \tag{3.7}$$

W sytuacji, gdy wiedza posiada różne wartości pewności (v), oraz jest rozmyta (w) użyta została miara opisana formułą 3.8.

$$d(CDV, ANSW) = \frac{1}{K} \sum_{i=1}^{K} (1 - \text{dist } (CDV_i, ANSW_i)) \tag{3.8}$$

gdzie:

$$\text{dist } (w_{CDV}, w_{ANSW}) =$$

$$\begin{cases} 0 & \text{, jeżeli } w_{ANSW} = NULL \\ -\frac{1}{K} \text{ abs } (w_{ANSW}) & \text{, jeżeli } v = 0 \\ v|w_{CDV} - w_{ANSW}| & \text{, jeżeli } v > 0 \end{cases} \tag{3.9}$$

gdzie:

v – jest pewnością wiedzy składowych wektora CDV,

w_{CDV} – jest wartością wagi w opisującej powiązania w wektorze CDV,

w_{ANSW} – wartością odpowiedzi użytkownika na pytania o określoną cechę, dla których przyjęto następujące kodowanie:

1	dla odpowiedzi „tak”,
0,5	dla odpowiedzi „przeważnie”,
-0,5	dla odpowiedzi „rzadko”,
-1	dla odpowiedzi „nie”,
0	określa odpowiedź „nie wiadomo”.

Odległość między wektorem CDV a ANSW liczona jest jako suma różnic pomiędzy odpowiedziami użytkownika a wiedzą systemu. Dla odpowiedzi użytkownika „nie wiadomo” składowa wektora ANSW (cecha) przy obliczaniu odległości zostaje pominięta. Dodatkowo wprowadzony współczynnik pewności wiedzy v pozwala wzmocnić składowe odległości, które są pewniejsze, a osłabić te, które są słabo zdefiniowane.

W celu przyspieszenia wyszukiwania, dla nieznanych wartości wagi $w = NULL$ i $v = 0$ możliwe jest wprowadzenie przybliżenia opartego na założeniu, że jeśli brak jest zdefiniowanego powiązania cecha – pojęcie, to cecha taka nie stosuje się do obiektu. Założenie $w = -1$ przyjmowane jest dla wartości pewności wiedzy $v = 0$ („nie wiadomo”), której wartość zmieniona na $v = 0,25$ zmniejszana jest o $\Delta = 0,05$ w kolejnych krokach wyszukania. Celem takiego uzupełnienia jest przyjęcie, dla nieznanych wartości powiązań, pewnego założenia pozwalającego szybciej (choć z możliwością błędu) w pierwszych krokach algorytmu wyszukiwania uzyskać zbiór najbardziej prawdopodobnych obiektów, które są później weryfikowane. Potencjalny niekorzystny wpływ przyjętego założenia na jakość wyszukiwania minimalizowany jest przez używanie coraz pewniejszej wiedzy, w miarę, jak algorytm zmniejsza liczbę obiektów, które mogą być rezultatem wyszukiwania. Zmniejszająca się w kolejnych krokach pewność wiedzy v, dla nieznanych powiązań pozwala w momencie gdy $v = 0$ wyłączyć zupełnie przyjęte założenie odnośnie braku wiedzy, przez co nie wpływa ono negatywnie na odnalezienie najbardziej prawdopodobnego obiektu.

Odległość między wektorem odpowiedzi ANSW a wektorami CDV opisującymi obiekty pozwala na utworzenie podprzestrzeni O(A) wykorzystywanej w kolejnym kroku algorytmu, gdzie użyte zostają tylko te obiekty, które są najbardziej prawdopodobne. W k kroku najlepszą z punktu widzenia wyszukiwania podprzestrzeń O(A) (3.10) będzie tworzyć podzbiór wszystkich

obiektów O, które mają minimalną odległość d_o (wektora CDV obiektu o od wektora odpowiedzi ANSW).

$$O(A)_k : o \in O \; d_o = \min_i \{d_i(ANSW, CDV(o_i))\} \tag{3.10}$$

Przyjęcie minimalizacji odległości do utworzenia zbioru najbardziej prawdopodobnych obiektów pozwala uzyskać najszybszą zbieżność algorytmu wyszukiwania. Podejście takie nie zawsze musi być jednak najkorzystniejsze z punktu widzenia wyszukiwania, co dyskutowane jest dalej. Również w sytuacji, gdy algorytm używany jest do akwizycji wiedzy, nie jest wskazane by obiekt wyszukiwany został odnaleziony w minimalnej liczbie kroków, ze względu na małą ilość wiedzy, która po takim procesie napływa do bazy wiedzy. W zastosowaniach dopuszczających niezgodności pomiędzy odpowiedziami użytkownika, a wiedzą systemu, rozbieżności w rezultatach testów kompensowane mogą być przez dodanie do odległości pewnego marginesu tolerancji.

W zastosowaniach algorytmu wyszukiwania kontekstowego do gry słownej (paragraf 3.4) kolejne podprzestrzenie O(A) wyznaczane są przez prawdopodobieństwa określające pewność, że obiekt jest rozwiązaniem. Wskazane obiekty pozwalają wyznaczyć zbiór cech, dla których zysk informacyjny wyznacza najkorzystniejszy test. W najprostszym podejściu prawdopodobieństwo, że obiekt zostanie użyty do utworzenia podprzestrzeni obliczone może być jako 3.11. Jego modyfikacja wiążąca odległość d z krokiem algorytmu k określona została dalej formułą 3.12.

$$p(O_i) = 1 - \frac{d_i}{\sum_{k=1}^{M} d_k} \tag{3.11}$$

gdzie:

d_i jest odległością obiektu O_i (reprezentowanego wektorem CDV) od wektora odpowiedzi, określoną formułą 3.8,

M jest liczbą obiektów.

Zaprezentowany w tym paragrafie algorytm wyszukiwania wykorzystujący opartą na wvCRK reprezentację wiedzy semantycznej wykazuje I tezę pracy: „reprezentacja wiedzy semantycznej stanowi podstawę dla algorytmu wyszukiwania kontekstowego". Zastosowanie wyszukiwania kontekstowego przedstawione zostało w kolejnych paragrafach 3.3 i 3.4. Natomiast w rozdziale 5 zaprezentowano jego użycie do uczenia pamięci semantycznej, opartego na dialogu z człowiekiem w języku zbliżonym do naturalnego.

3.3 Wyszukiwanie kontekstowe dla danych medycznych

Wyszukiwanie kontekstowe, oparte na wskazywaniu cech posiadanych przez poszukiwany obiekt, może zostać użyte do wspomagania procesu decyzyjnego. W szczególności proces taki można rozważyć jako stawianie diagnozy przez eksperta, który odnajdując czynniki charakterystyczne a następnie odróżniając podobne obiekty poprzez cechy różnicujące wskazuje najbardziej prawdopodobny przypadek. Oczywiście zagadnienie diagnostyki jest dużo bardziej złożone niż tylko wyszukiwanie w przestrzeni cech, niemniej jednak aspekt ten stanowi immanentną część procesu diagnostycznego.

Szczególnym przypadkiem procesów diagnostycznych są aplikacje medyczne, gdzie podstawowym celem diagnozy jest identyfikacja schorzenia umożliwiająca opracowanie leczenia. W specjalizacjach medycznych istnieją określone, ustalone schematy postępowania mające na celu rozpoznawanie chorób. Pozwalają one na identyfikację schorzenia i na jej podstawie ustalenie leczenia. Istnienie takich formalnych metod ze swojej natury redukuje zbiór patologii, które są brane pod uwagę, ograniczając go tylko do najbardziej typowych. Schematy diagnostyczne pozwalają początkującym lekarzom nabyć podstawowe umiejętności klasyfikacji schorzeń, a w zastosowaniach praktycznych zabezpieczają lekarza przed sankcjami prawnymi, wynikłymi z użycia niewłaściwej procedury do oznaczenia choroby.

Jednym z medycznych schematów diagnostycznych jest utworzone przez Amerykańskie Towarzystwo Psychiatryczne narzędzie do diagnoz schorzeń psychiatrycznych DSM IV (*Diagnostic and Statistical Manual of Mental Disorders*) [24]. Ten złożony system pozwala zarówno na klasyfikację zaburzenia, jak również na prognozę jego rozwoju i planowanie leczenia. Do określania choroby w systemie DSM służy drzewo decyzji, które zostało użyte do przetestowania właściwości algorytmu wyszukiwania kontekstowego. Wiedza zapisana przy użyciu drzewa diagnoz DSM posłużyła do utworzenia cech i obiektów dla pamięci semantycznej, na których zbadane zostały własności algorytmu wyszukiwania kontekstowego.

By można było przeprowadzić porównanie algorytmu wyszukiwania kontekstowego i diagnozy opartej na drzewie DSM, konieczne było przeprowadzenie w nim kilku modyfikacji. Z oryginalnej struktury drzewa DSM, usunięte zostały dwa powiązania tworzące cykle w drzewie (występujące w liściach przekierowania do innego poddrzewa decyzyjnego). Drugim dopasowaniem była zamiana 4 węzłów testowych na węzły będące klasami decyzyjnymi. W oryginale były one wskazaniami rezultatu diagnozy a dalsze testy przeprowadzane były w celu ich doprecyzowania i prowadziły do dokładniejszego

określenia schorzenia. W pamięci semantycznej istnieje możliwość doprecyzowywania obiektów poprzez utworzenie między nimi powiązania typu *is_a*. Dopasowanie takie jest jednak wprowadzaniem nowej wiedzy do bazy (w postaci typowanego powiązania), która nie była podana w oryginale. Kolejną modyfikacją było usunięcie dla 3 węzłów ścieżek o długości 2, które były krawędziami łączącymi je z nadrzędnymi węzłami. Usunięto je, ponieważ tworzyły dodatkowe ścieżki diagnostyczne, które mogły być zastąpione krótszymi. W celu zachowania przejrzystości dla osoby przeglądającej strukturę DSM, do każdego poddrzewa decyzji dodany został u jego szczytu jeden węzeł, będący etykietą opisującą rodzaj drzewa diagnostycznego – węzeł ten jest pomijany podczas wszystkich zrealizowanych doświadczeń.

W porównaniu do rozmiaru drzewa DSM IV wspomniane dopasowania są niewielkie. Minimalnie zmieniają jego strukturę, co potencjalnie może zmieniać wiedzę w nim zapisaną. Modyfikacje te nie są jednak na tyle duże, by znacząco zmienić znaczenie jednostek chorobowych zapisanych w DSM IV. Przeprowadzenie wymienionych dopasowań było konieczne w celu jednoznacznego wyznaczenia długości ścieżek decyzyjnych, co jest niezbędne do wykonania porównań z algorytmem wyszukiwania kontekstowego. Argumentem pozwalającym na użycie tak dopasowanych danych jest zastosowanie ich do prezentacji i zbadania własności algorytmu, a nie praktyczna realizacja systemu wspomagania decyzji medycznych, którego wykonanie wymagałoby zaangażowania w projekt eksperta z tej dziedziny. Zaznaczyć należy, że przeprowadzone dopasowania nie wynikają z ograniczeń powodowanych przyjętą reprezentacją wiedzy w pamięci semantycznej, ponieważ mogą być składowane w niej struktury wiedzy opisane z użyciem grafu, a nie tylko drzew. Tak więc rozwiązania przyjęte w pamięci semantycznej umożliwiają zapisanie pełnej wiedzy z DSM IV.

W wyniku dopasowań otrzymano przedstawioną na rysunku 3.2 strukturę drzewa diagnostycznego DSM IV. Składa się ono z 6 poddrzew decyzji dla kolejnych schorzeń psychicznych:

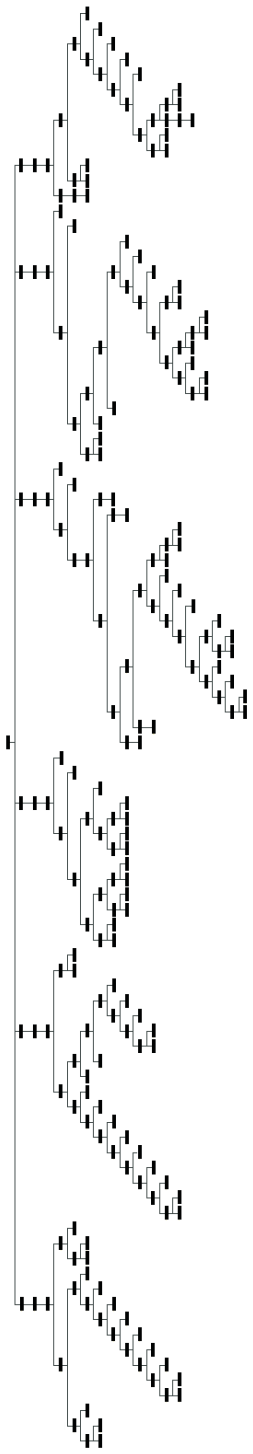
1. *differential diagnosis of mental disorders*, w dalszej części pracy opisywane skrócie jako „mental"

2. *differential diagnosis of substance-included disorders (not including dependence and abuse)*, skrót „substance"

3. *differential diagnosis of psychotic disorders*, skrót „psychotic"

4. *differential diagnosis of mood disorders*, skrót „mood"

5. *differential diagnosis of anxiety disorders*, skrót „anxiety"



Rysunek 3.2: Struktura drzewa decyzji DSM IV

6. *differential diagnosis of somatoform disorders*, skrót „somatoform"

Rozkład węzłów decyzyjnych i klas w poszczególnych poddrzewach przedstawiał się następująco: (węzły decyzyjne (testowe) : klasy – jednostki chorobowe) 15: 14, 17: 18, 12: 13, 22: 19, 16: 17, 16: 13. Podane liczby węzłów decyzyjnych nie uwzględniają 2 węzłów będących u szczytów poddrzew decyzyjnych, ponieważ nie wnoszą one nic do procesu decyzji.

W paragrafie tym przedstawiono otrzymane wyniki eksperymentów porównujących algorytm wyszukiwania kontekstowego z procesem decyzyjnym opisanym w DSM IV. Porównana została szybkość diagnozowania oparta na testowaniu sekwencyjnym węzłów drzewa DSM, z klasyfikacją przez testy cech, które wybierane są poprzez największy zysk informacyjny z nimi związany (3.4). Sprawdzony został także wpływ dodania marginesu do minimalnej odległości d (na podstawie której tworzone są kolejne podprzestrzenie wyszukań O(A)) na szybkość wyszukiwania V_s (szybkość zbieżności algorytmu do rozwiązania, będącego tu poszukiwaną jednostką chorobową). Dodanie tego rozszerzenia ma na celu uwzględnienie sytuacji, w których wiedza zapisana w bazie wiedzy nie jest całkowicie zgodna z rezultatami testów pochodzącymi od użytkownika. Niezgodność ta nazywana będzie dalej *błędami użytkownika*. Zaznaczyć tu należy fakt, że błędy dopasowania wyników testów powstają nie tylko z niepoprawnych odpowiedzi udzielonych przez człowieka, a wynikają również z błędnych (odmiennych do rzeczywistości) zapisów w bazie wiedzy. Ze względu na brak wystarczająco specjalistycznej wiedzy w obrębie medycyny, nie jest dyskutowana w tym rozdziale korekta bazy wiedzy poprzez interakcję z człowiekiem. Opis takiego podejścia i otrzymane rezultaty przedstawione są w dalszej części pracy, przy opisie aplikacji algorytmu wyszukiwania kontekstowego do zagadnień związanych z pozyskiwaniem wiedzy zdroworozsądkowej (paragrafy 5.1, 5.2). W kolejnych paragrafach tego rozdziału rozważany jest negatywny wpływ błędów popełnianych przez użytkownika na szybkość wyszukiwania V_s oraz konieczność rozszerzenia zbioru cech wektorów CDV w celu kompensacji tego błędu. Przedstawiona została możliwość poprawy szybkości wyszukiwania przez wprowadzenie zgadywania klasy, do której zakwalifikowany zostaje wyszukiwana przez użytkownika jednostka chorobowa, jeszcze przed wykorzystaniem wszystkich możliwości oddzielenia go od innych, znajdujących się w najbardziej prawdopodobnej podprzestrzeni. Podejście takie oczywiście niesie ze sobą możliwość nieprawidłowych klasyfikacji, co jest ujęte ilościowo w przeprowadzonych eksperymentach. Ostatni paragraf rozważa możliwość uzupełnienia nieznanych wartości cech i wpływ takiego dopełnienia CDV na jakość klasyfikacji.

3.3.1 Porównanie algorytmu wyszukiwania kontekstowego z drzewami decyzji DSM IV

Algorytm wyszukiwania kontekstowego oparty na wyznaczaniu cechy do testu używa wartości zysku informacyjnego, jaki niesie każda z cech z przestrzeni semantycznej. Podejście takie stosowane jest w algorytmie ID3 [71], który dla nie zbalansowanych drzew decyzyjnych powinien dawać lepsze rezultaty, niż poruszanie się po kolejnych węzłach decyzji. Eksperymentem pozwalającym ocenić jakość algorytmu wyszukiwania kontekstowego (w sensie szybkości wyszukiwania) jest sprawdzenie, czy używając zysku informacyjnego można efektywniej użyć drzewa decyzji DSM tak, by końcową diagnozę postawić szybciej, niż jeśli jest ona realizowana przez sekwencyjne sprawdzanie schematu DSM IV.

Porównanie szybkości wyszukiwania jednostek chorobowych z użyciem algorytmu wyszukiwania kontekstowego, z diagnozą przez drzewo decyzji DSM odbyło się poprzez ocenę średniej długości ścieżki w poszczególnych poddrzewach decyzji psychiatrycznych, ze średnią liczbą testów V_s przeprowadzanych przez algorytm, która jest potrzebna do uzyskania końcowej diagnozy.

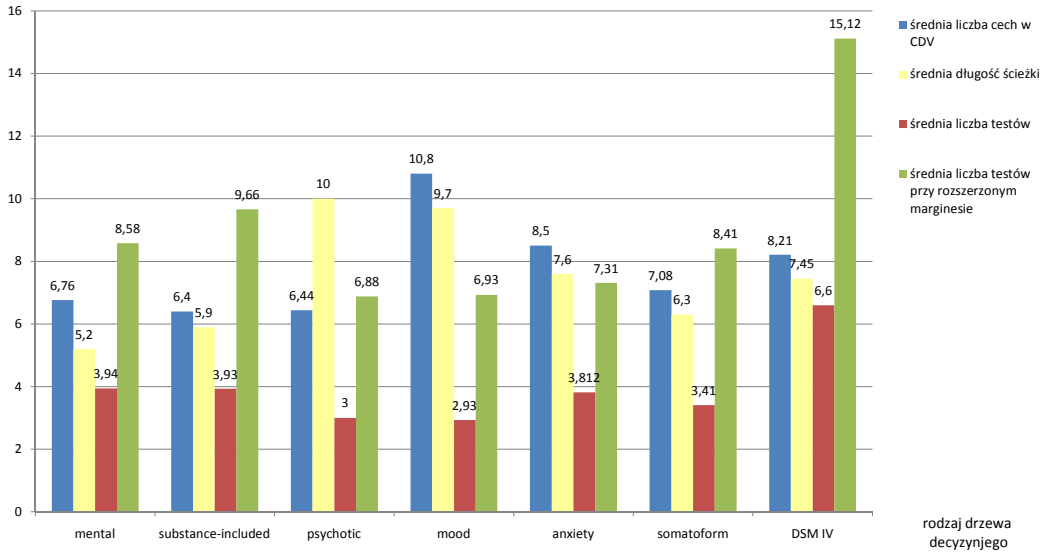
Otrzymane rezultaty przedstawione zostały na rysunku 3.3. Kolorem niebieskim zaznaczono słupki przedstawiające wartości średniej liczby cech w wektorach CDV opisujących jednostki chorobowe. Dla kolejnych poddrzew wyniki uzyskane przez algorytm wyszukiwania kontekstowego naniesiono na słupkach czerwonych. Przedstawiają one średnią liczbę testów V_s potrzebnych do przeprowadzenia klasyfikacji. Średnią długość każdego poddrzewa DSM obrazują słupki żółte. Przy obliczaniu średniej długości ścieżki wyłączono pierwsze węzły (widoczne u góry drzewa diagnostycznego na rysunku 3.2), ponieważ nie wnoszą one nic do procesu decyzji. Algorytm wyszukiwania kontekstowego węzły te automatycznie ignoruje, ze względu na niesiony przez nie zerowy zysk informacyjny.

Porównując wysokości słupków żółtych i czerwonych widać, że algorytm oparty o wybór cechy posiadającej największy zysk informacyjny jest lepszy niż sekwencyjne testowanie węzłów. Wynika to z efektywniejszego wydzielania kolejnych podzbiorów O(A) najbardziej prawdopodobnych obiektów, poprzez wybraną do testu cechę. Porównując wyniki uzyskane przez algorytm wyszukiwania kontekstowego w kolejnych poddrzewach decyzyjnych (średnią liczbę pytań w każdym z poddrzew) z wynikiem uzyskanym dla całego drzewa DSM (ostatnia grupa słupków) zaobserwować można znaczny skok średniej liczby zapytań. Wynika to ze sposobu działania algorytmu, który podczas przeszukiwania wpierw ustalić musi właściwe drzewo, co odbywa się w średnio 3 pytaniach (testach cechy). Na pewno uzyskać można by było dużo lepsze wyniki, wprowadzając dodatkowe cechy, które wiązałyby poddrzewa ze

sobą. Wiązałoby się to z koniecznością pozyskiwania nowej wiedzy do DSM IV, która pochodzić może tylko od ekspertów w dziedzinie medycyny.

Przy przeprowadzaniu porównania diagnozowania sekwencyjnie testującego węzły DSM, z wyszukiwaniem opartym na zysku informacyjnym przyjęto założenie, że wynik testu cechy jest zawsze zgodny z opisem poszukiwanej choroby w bazie wiedzy. Nie dopuszcza się tu wystąpienia niezgodności pomiędzy opisem wyszukiwanego obiektu (będącego zbiorem cech) a rezultatami testów. Założenie takie jest konieczne do przeprowadzenia samego porównania algorytmu wyszukiwania kontekstowego z wyszukiwaniem sekwencyjnym. Przyjmowane jest ono również przy realizacji wyszukiwania w typowym drzewie decyzyjnym – nie dopuszcza ono wystąpienia rozbieżności między wynikami testów a wiedzą zapisaną w drzewie. W rzeczywistych zastosowaniach takie rozbieżności będą jednak występować. Wynikać mogą one z szeregu czynników, np.: określona cecha może nie wystąpić przy schorzeniu, bądź też nie jest możliwe jej ustalenie albo zostało ono błędnie ocenione przez osobę przeprowadzającą diagnozę. W sytuacji takiej drzewo decyzyjne, oparte na typowym algorytmie dziel i rządź prowadzi do braku lub też błędnych klasyfikacji. Rozwiązaniem prowadzącym do uodpornienia algorytmu wyszukiwania kontekstowego na wystąpienie niezgodności między wynikami testów a wiedzą zapisaną w pamięci semantycznej (błędów użytkownika) jest dodanie do odległości (3.8) między wektorem odpowiedzi użytkownika ANSW a obiektami reprezentowanymi przez CDV, pewnego marginesu, dzięki któremu do kolejnych podprzestrzeni użyte zostaną nie tylko obiekty mające odległość minimalną, ale również te, które znajdują się w granicy przyjętej tolerancji.

Sytuacja, gdy zwiększony został margines przedstawiono na rysunku 3.3 w wartościach czwartego ze słupków. Pokazuje on liczbę testów w poszczególnych grupach drzew decyzji DSM w sytuacji, gdy jedynie dopuszczona została możliwość (błędy nie były popełniane) wystąpienia błędnych odpowiedzi, która jest kompensowana poprzez rozszerzony margines tolerancji. Testy w przypadku błędnych odpowiedzi zostały szczegółowo opisane w podsekcji 3.3.3. Zaprezentowane na wykresie wartości otrzymane zostały dla przyjętego marginesu dopuszczającego wystąpienie jednej różnej cechy wchodzącej w skład CDV i ANSW. Dla dodanego do odległości marginesu widać znaczne zwiększenie średniej liczby testów, która musi zostać przeprowadzona. Wartość ta jest większa zarówno w stosunku do sekwencyjnego poruszania się po drzewie, jak i od poruszania się opartego na zysku informacyjnym. Wynika to z konieczności „upewniania się" przez algorytm, że odrzucone w kolejnym kroku obiekty (nie włączone do kolejnej podprzestrzeni O(A)) nie są poszukiwaną jednostką chorobową. Podkreślić należy, że ta zwiększona średnia liczba pytań pozwalać będzie, pomimo błędnej odpowiedzi w teście węzła, na dokonanie właściwej diagnozy (dokonanie właściwej klasyfikacji), co przy



Rysunek 3.3: Wykresy porównujące algorytm wyszukiwania kontekstowego z drzewami decyzji DSM

zastosowaniu typowego drzewa decyzji nie jest możliwe.

Przeprowadzone doświadczenia wykazują przewagę wyszukiwania opartego na zysku informacyjnym nad sekwencyjnym testowaniem węzłów. Widoczna różnica w liczbie testów koniecznych do odnalezienia właściwej jednostki chorobowej wynika z użycia w algorytmie wyszukiwania kontekstowego efektywniejszego wydzielania najbardziej prawdopodobnych obiektów.

3.3.2 Rozszerzenie zbioru cech i wprowadzenia zgadywania

W systemie DSM drzewo decyzyjne jest stosunkowo niewielkie, zwłaszcza gdy analizie poddawane są osobno poszczególne jego poddrzewa. W sytuacji takiej, gdy wystąpi pojedyncza niezgodność między bazą wiedzy a odpowiedzią użytkownika (rezultacie testu), poszukiwany obiekt nie będzie poprawnie odnajdywany. Dzieje się to nie tylko poprzez to, że odcinane jest poddrzewo decyzji, w którym jest właściwa jednostka chorobowa, ale również z powodu, że często wyszukiwany obiekt staje się innym, już zapisanym w systemie – poprzez zamianę jednego atrybutu może nastąpić utożsamienie dwóch wektorów CDV. W okolicznościach takich poprawne odnalezienie staje się niemożliwe, bądź poprzez wskazanie przez algorytm niewłaściwego obiektu, bądź też poprzez niemożność oddzielenia właściwej jednostki chorobowej od

najbardziej do niej podobnej.

W celu ustalenia, ile procentowo obiektów będzie niewłaściwie klasyfikowanych przy pojedynczym przekłamaniu w teście cechy, przeprowadzony został następujący test: dla każdej wyszukiwanej jednostki chorobowej w kolejnych poddrzewach DSM wprowadzono dla pojedynczej cechy przekłamanie w wyniku testu. Schemat testu przedstawiony został w pseudokodzie:

```
foreach CDV do
przeprowadz_wyszukanie(CDV)
k = liczba_krokow_wyszukania
for n = 0 to k
begin
i = 0
while ! znalazl %szukaj obietu
    begin
    cecha=wyznacz_ceche_do_zapytania; i++
    if i = n % wygeneruj błąd testu
    w = CDV[cecha]
    case w :
       1 : test_result =  0
       0 : test_result = -1
      -1 : test_result =  1
    end
end
%wyznacz średnią wartość i
```

Dla każdej jednostki chorobowej procedura wyszukiwania powtarzana jest k razy. Przy każdym kolejnym wyszukaniu następuje wprowadzenie przekłamania. Miejsce wstrzyknięcia błędu (wskazanie, która cecha zostaje przekłamana) określają kolejne wartości z k wyszukań dla każdego z wektorów CDV. Pozwala to wyznaczyć średnią liczbę pytań konieczną do podjęcia decyzji, uniezależniając się od miejsca (kroku algorytmu – pytania użytkownika), w którym występuje niezgodność. Sposób zrealizowania błędu (instrukcja `case`

w algorytmie) daje najbardziej niekorzystne odcinanie podprzestrzeni oparte na wyznaczaniu odległości między CDV a wektorem ANSW (3.8).

Zsumowana średnia liczba nieprawidłowo rozpoznanych chorób w kolejnych poddrzewach decyzji, przy popełnieniu pojedynczego błędu wyniosła 18,5%. Dla całego drzewa DSM wartość ta wyniosła 9,7%. Mniejsza wartość w większym drzewie wynika z istnienia większej liczby cech, pozwalających lepiej rozdzielić jednostki chorobowe od siebie. Uzyskane wyniki otrzymano dla algorytmu używającego rozszerzonego marginesu do konstrukcji zbioru najbardziej prawdopodobnych obiektów O(A). Dla poddrzew DSM użycie algorytmu bez marginesu błędu skutkowało niepoprawną klasyfikacją rzędu 85%.

Otrzymana wartość błędu klasyfikacji przy popełnianiu błędów użytkownika jest dość znaczna. Wynika ona z faktu, że przyjęte rozwiązanie do poprawy błędów użytkownika przy pomocy marginesu dodawanego do odległości, powoduje sytuacje, w których nie można podjąć jednoznacznej decyzji klasyfikacyjnej (co wynika z niewystarczającej liczby cech, które można poddać do testu). Brak cech do testowania objawia się w końcowych krokach algorytmu, gdzie w skutek błędnych odpowiedzi nie można oddzielić dwóch jednostek chorobowych od siebie, przez co nie można postawić jednoznacznej diagnozy. Konieczne jest tu zastosowanie kompensacji braku wiedzy systemu poprzez rozszerzenie ilości danych – dodania nowych cech do wektorów CDV.

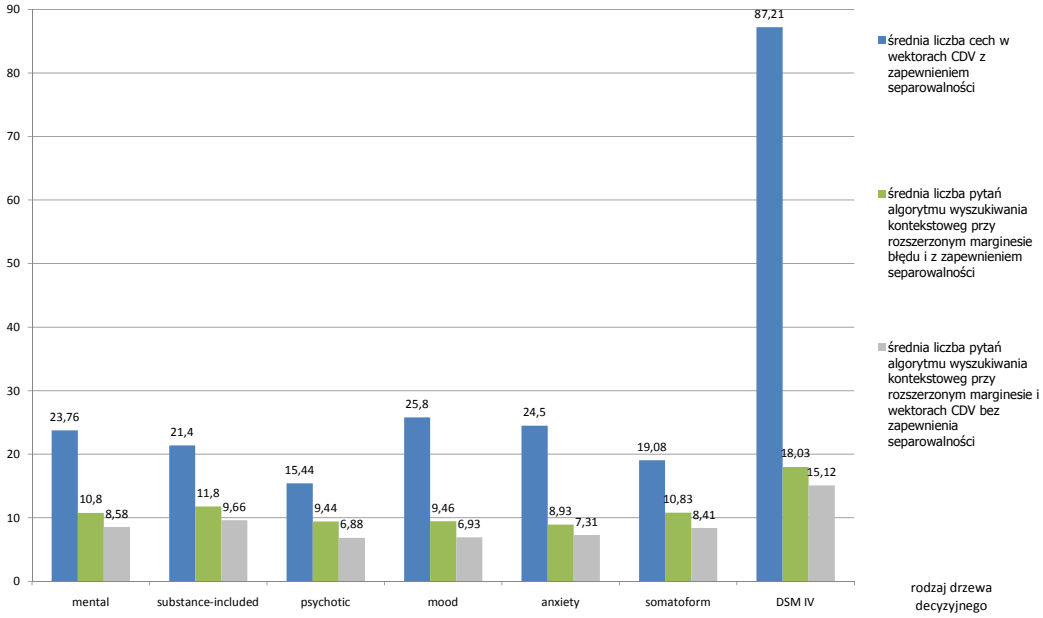
W celu przeprowadzenia eksperymentów badających własności algorytmu odpornego na błędy użytkownika dokonano dopasowania polegającego na rozszerzeniu każdego wektora CDV o dodatkowe cechy. Przyjęto, że dodatkowymi cechami są nazwy jednostek chorobowych, które będą miały ustawioną wartość powiązania na 1 (stosuje się) tylko dla opisywanej przez nie obiektów. Pozostałe cechy (wynikłe z nazw innych chorób) mają wartość -1. Dopasowanie takie zapewnia możliwość oddzielenia jednostek chorobowych od siebie nawet w sytuacji, gdy użytkownik popełni błąd i zabraknie cech do przeprowadzenia testu. Nazwane zostało *zapewnieniem separowalności*, co skutkuje poprawą rozdzielania obiektów widoczną pod koniec procesu decyzyjnego. Dodanie do wektora CDV cechy będącej nazwą jednostki chorobowej (klasy decyzyjnej) powodować będzie pojawienie się pytań typu: „czy jest to określona choroba".

W zastosowaniu do wspomagania rzeczywistego procesu diagnostycznego dodanie takiej cechy jest zbyt daleko idącym uproszczeniem (ponieważ może skutkować zadaniem pytań o to, czego dopiero szukamy jeszcze przed postawieniem diagnozy). Jednak do zbadania własności algorytmu wprowadzenie takich sztucznych cech pozwala sprawdzić jego odporność na błędy użytkownika. Kompensacja braku danych (cech opisujących jednostki chorobowe) wprowadzona została z powodu braku możliwości pozyskania rzeczy-

wistej wiedzy specjalistycznej, która dla praktycznych zastosowań pochodzić musiałaby od twórców systemu DSM. Ze względu na niski zysk informacyjny jaki niosą nowo utworzone cechy, są one używane w końcowych krokach algorytmu, gdy do rozdzielenia pozostaje kilka jednostek chorobowych. Dodatkowo, do algorytmu wyboru cech zaimplementowana została modyfikacja polegająca na tym, że w wypadku, gdy lista cech, o które można się zapytać (wyznaczana największym zyskiem informacyjnym) zawiera zarówno cechy pierwotnie pochodzące z DSM IV, jak i dodane cechy powstałe z nazw chorób do przeprowadzenia testu, wybierane są wpierw cechy diagnostyczne.

Zaznaczyć należy, że przeprowadzane w tym paragrafie eksperymenty badają działanie algorytmu w sytuacji, gdy wprowadzane są w nim rozszerzenia mające na celu uodpornienie na błędy użytkownika, a porównanie przeprowadzane jest z algorytmem bez takich rozszerzeń, co uwypukla różnicę w szybkości klasyfikacji. Przeprowadzone doświadczenia mają na celu zbadanie, jakie zmiany w szybkości klasyfikacji, w odniesieniu do pierwotnej wersji algorytmu powodują wprowadzone modyfikacje (rozszerzenie marginesu i zapewnienie separowalności). Nie jest tu analizowany wpływ błedów użytkownika na działanie algorytmu (zrealizowane zostało to w paragrafie kolejnym).

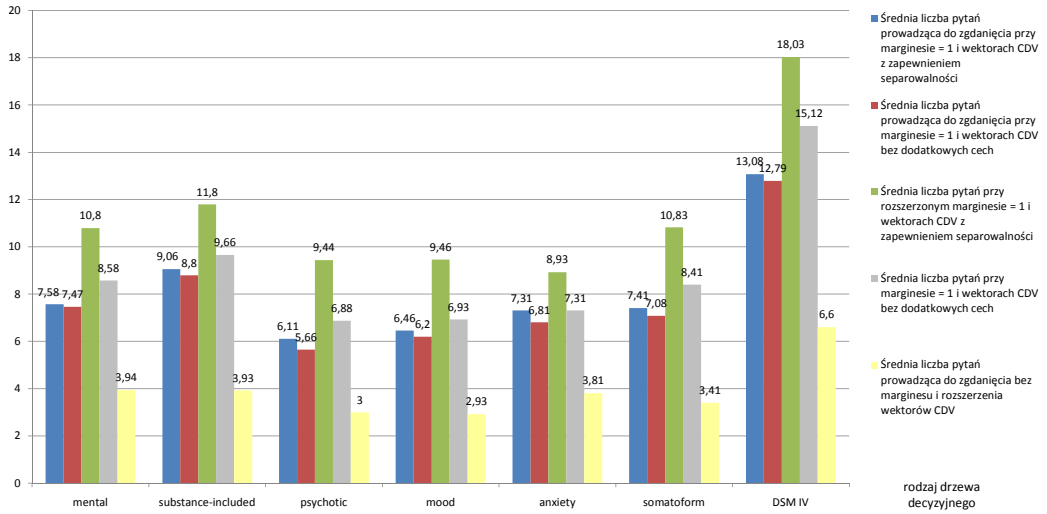
Na rysunku 3.4 wykresy dla kolejnych drzew decyzji przedstawiają kolorem niebieskim, jak zmieniła się średnia liczba cech w wektorach CDV po wprowadzeniu dodatkowych cech utworzonych z nazw jednostek chorobowych (zapewnieniu separowalności). Wpływ tej modyfikacji na szybkość wyszukiwania – średnią liczbę zapytań, jaka jest konieczna do postawienia diagnozy, przedstawiają wartości umieszczone w słupkach zaznaczonych kolorem zielonym, obrazujące działanie algorytmu z zapewnieniem separowalności. Dla porównania zamieszczone zostały wartości z rysunku 3.3 obrazujące działanie algorytmu bez wprowadzania rozszerzenia cech w wektorach CDV dla rozszerzonego marginesu (ostatni słupek). Porównując dwa ostatnie słupki w grupach widać, że wprowadzenie do wektorów CDV cech dodatkowych (pochodzących z nazw chorób) nieznacznie pogarsza szybkość klasyfikacji. Pamiętać jednak należy, że w wypadku pojawienia się błędów użytkownika te dodatkowe cechy zapewnią możliwość poprawnej klasyfikacji. Pogorszenie szybkości klasyfikacji wynika z pojawienia się pytań o cechy będące jednostkami chorobowymi, które słabo separują podprzestrzenie najbardziej prawdopodobnych obiektów. W szczególności, wynika to z zadawania na końcu dodatkowego pytania: czy wynik klasyfikacji – diagnoza, posiada cechę będącą wynikiem klasyfikacji. Dopasowanie polegające na rozszerzeniu zbioru cech, wraz ze zwiększonym marginesem błędu powodować będzie więc wydłużenie procesu decyzyjnego (średniej liczby pytań koniecznych do postawienia końcowej diagnozy). Wynika to z przyjętego warunku stopu, który jest warunkiem opartym na liczności obiektów w podprzestrzeni. W podsta-



Rysunek 3.4: Wpływ dodania marginesu do odległości między CDV i ANSW i rozszerzenia zbioru cech CDV na szybkość zbieżności algorytmu do wyszukiwanego obiektu

wowej wersji zatrzymanie algorytmu wyszukiwania i dokonanie klasyfikacji następuje, gdy w podprzestrzeni znajduje się tylko jedna choroba, która jest decyzją klasyfikacyjną. Decyzję taką można podejmować jednak wcześniej, gdy w kolejnych podprzestrzeniach pozostają te same choroby, a kolejne testy zwiększają jedynie pewność (poprzez zwiększenie odległości od wektora odpowiedzi wszystkich obiektów, oprócz jednego – najbardziej prawdopodobnego). W sytuacji, w której kolejne pytania nie zmieniają diagnozy, tylko upewniają najbardziej prawdopodobny obiekt można wprowadzić poprawę szybkości klasyfikacji polegającą na stawianiu diagnozy (klasyfikacji), gdy w podprzestrzeni znajduje się kilka obiektów. W celu poprawy szybkości wyszukiwania przyjęto, że podejmowanie decyzji diagnostycznej może nastąpić, gdy po kolejnych dwóch testach cech podprzestrzeń O(A) (zawierająca więcej niż jeden obiekt) się nie zmienia.

Wykresy na rysunku 3.5 przedstawiają wpływ wprowadzenia możliwości dokonania wcześniejszych klasyfikacji (nazwanych zgadywaniem) na liczbę pytań stawianych przez algorytm. Dwa pierwsze słupki obrazują działanie algorytmu w sytuacji użycia tej heurystyki, przy rozszerzonym marginesie błędu. Przedstawione wyniki zostały uzyskane dla CDV rozszerzonych o cechy zapewniające separowalność (kolor niebieski) i bez (kolor czerwony). Dla



Rysunek 3.5: Wpływ wprowadzenia zgadywania na szybkość zbieżności algorytmu, przy rozszerzonym marginesie i zapewnieniu separowalności

porównania naniesiono wartości z rysunku 3.4 obrazujące sytuację, gdy nie jest używane zgadywanie. W kolejnych słupkach podano średnie liczby kroków algorytmu dla rozszerzonych wektorów CDV (zielony) i CDV pozyskanych bezpośrednio z DSM (szary). Różnice w wysokościach tych słupków obrazują poprawę jaką dało wprowadzenia zgadywania.

Przeprowadzone eksperymenty z użyciem zgadywania bez zastosowania marginesu i dla podstawowych wektorów CDV nie zmieniły wyników uzyskanych bez stosowania takiej heurystyki, co przedstawiają wartości ostatniego słupka.

Przyjęta do zgadywania heurystyka używana w sytuacji, gdy użytkownik nie podawał błędnych odpowiedzi, nie powodowała błędnych klasyfikacji – ani razu wyszukiwana jednostka chorobowa nie została pomylona z inną. Eksperymenty z niezgodnymi z bazą wiedzy wynikami testów dyskutowane są poniżej.

3.3.3 Wpływ błędów użytkownika na wynik i szybkość diagnozy

Opisana wcześniej możliwość uodpornienia algorytmu wyszukiwania na wystąpienie niezgodności pomiędzy bazą wiedzy a testami cech – odpowiedziami udzielanymi przez użytkownika porównana została z podstawową wersją algorytmu, nie zawierającą uodpornienia na błędy. Wyniki uzyskane przez standardowy algorytm są wartościami referencyjnymi, pokazującymi,

jak algorytm działa w sytuacji optymalnej. Porównując wyniki uzyskane w tych dwóch podejściach widać, że wprowadzone rozszerzenia mają negatywny wpływ na szybkość procesu decyzyjnego (dyskutowany przy rysunkach 3.4 i 3.5). Pozwalają one jednak na poprawę klasyfikacji w wypadku zaistnienia błędów użytkownika. Przeprowadzona analiza rozszerzeń algorytmu uodparniających klasyfikację na wystąpienie niezgodności między rezultatami testów a bazą wiedzy rozważa tylko porównanie z podstawowym algorytmem, a pomija samą sytuację, gdy błędy zostają faktycznie popełnione. W paragrafie tym przedstawione zostaną otrzymane wyniki w sytuacji, gdy w procesie decyzyjnym wystąpił błąd użytkownika. W tym miejscu trzeba zaznaczyć, że nie tylko samo wystąpienie błędu użytkownika, ale również i miejsce jego popełnienia – numer pytania, w którym błędna odpowiedź została podana ma wpływ na liczbę kroków (pytań) zadawanych przez algorytm (szybkość klasyfikacji V_s). W związku z tym, otrzymanie wartości średniej długości procesu decyzji wymagało powtórzenia dla każdej jednostki chorobowej, wyszukiwania z błędem użytkownika pojawiającym się w kolejnych krokach procesu decyzyjnego.

W przeprowadzonych doświadczeniach wartości średniej liczby pytań otrzymane zostały według schematu przedstawionego pseudokodem w paragrafie 3.3.2. Każde wyszukanie zapisanej w pamięci semantycznej jednostki chorobowej (podjęcie decyzji diagnostycznej) przeprowadzane zostało k razy, gdzie k określa maksymalną liczbę zapytań potrzebnych do dokonania klasyfikacji. Każde kolejne wykonanie wyszukania przeprowadzane jest z „wstrzykniętym" błędem w kroku (pytaniu), określonym numerem kolejnego wyszukania. Uśredniona liczba wszystkich wyszukań jest wartością określającą długość procesu decyzyjnego dla jednego obiektu. Jest to składowa wartości określającej długość procesu diagnostycznego w całym poddrzewie DSM.

Każdy z przeprowadzonych eksperymentów zrealizowany został w dwóch przypadkach: dla algorytmu zgadującego diagnozę i dla algorytmu przeprowadzającego wyszukiwanie, aż do momentu, gdy w podprzestrzeni pozostanie tylko jedna jednostka chorobowa. Otrzymane wyniki przedstawiono na wykresach na rysunku 3.7, do którego załączono legendę na rysunku 3.6. Na przedstawionych wykresach, każde z poddrzew DSM reprezentowane jest wynikami w grupie dwóch słupków: pierwszy podaje wartości uzyskane dla algorytmu ze zgadywaniem, drugi wyniki uzyskane przy wyszukiwaniu przeprowadzanym, aż do momentu, gdy w podprzestrzeni pozostaje jeden obiekt. W sytuacji, gdy dopuszczamy możliwość zgadywania, mogą pojawić się błędy klasyfikacji wynikłe z przedwczesnego podjęcia decyzji diagnostycznej. Liczba niepoprawnych klasyfikacji, jaka została popełniona w każdym z poddrzew została przedstawiona wysokością czerwonego słupka. Ponieważ wyszukiwa-

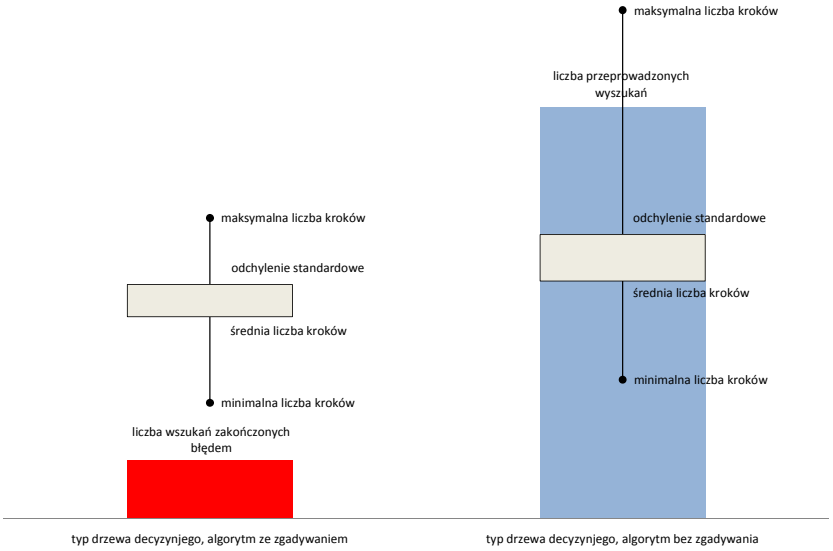
nie prowadzane jest do momentu, gdy w podprzestrzeni znajduje się jeden obiekt, nie skutkuje ono błędami decyzji, nie było więc celowe nanoszenie tych wartości na wykres. Wysokość drugiego słupka w grupie przedstawia liczbę zrealizowanych wyszukań (wynikającą z dodania „kroczącego" wstrzykiwania błędu). Wartość ta dla algorytmu ze zgadywaniem i bez jest taka sama.

Na wykresie 3.7 naniesiono również minimalne, maksymalne oraz średnie liczby wyszukań dla każdego przypadku użycia algorytmu wyszukiwania kontekstowego. Pomimo znacznego rozstrzelenia wartości minimalnych i maksymalnych liczby kroków algorytmu, ich wartość średnia dobrze je opisuje, co wskazują relatywnie niewielkie wartości odchylenia standardowego. W wypadku pojawienia się błędów użytkownika średnie liczby kroków algorytmu wskazują na ich negatywny wpływ na szybkość zbieżności algorytmu w stosunku do sytuacji, gdy udzielane były za każdym razem poprawne odpowiedzi (rysunki 3.4 i 3.5). Wyniki uzyskane w kolejnych poddrzewach wskazują, że wprowadzenie zgadywania skutkuje poprawą szybkości odnajdywania obiektów również w wypadku, gdy użytkownik udziela niepoprawnych odpowiedzi dla testowanej cechy. Poprawa szybkości odbywa się jednak kosztem dokładności klasyfikacji – pojawiają się błędne diagnozy, których liczby opisują wysokości słupków czerwonych.

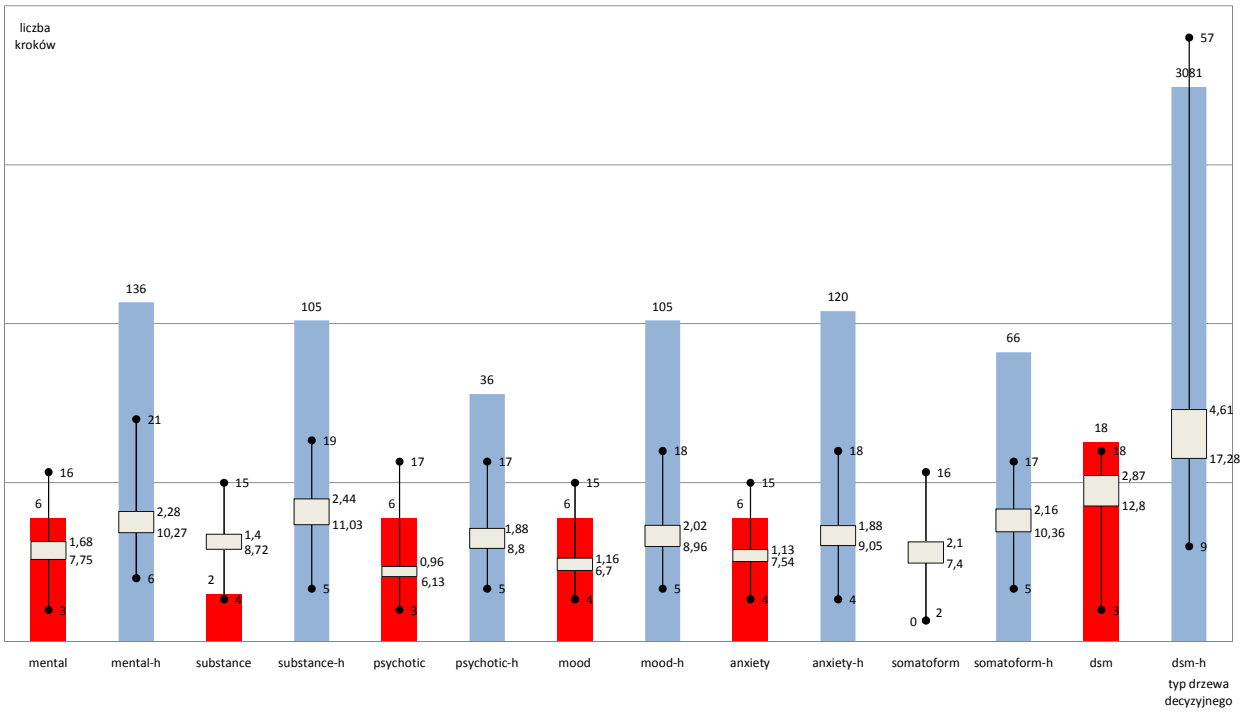
Za zasadniczy sukces wprowadzonych rozszerzeń w algorytmie wyszukiwania kontekstowego, które obrazują przeprowadzone eksperymenty należy przyjąć możliwość uwzględnienia wystąpienia niezgodności pomiędzy bazą wiedzy a rezultatami testów cech. Odbywa się to kosztem zmniejszenia szybkości klasyfikacji (zwiększonej liczby testów), pociągającej za sobą konieczność wprowadzenia nowych cech. Próby poprawy szybkości klasyfikacji przed uzyskaniem jednoznacznej decyzji poprzez wprowadzenie zgadywania skutkowały pojawieniem się nowych błędów klasyfikacyjnych. W związku z tym, użyteczność heurystyki zgadywania uzależniona jest od tego, jak ważna jest precyzja klasyfikacji.

3.3.4 Uzupełnienie nieznanych wartości atrybutów

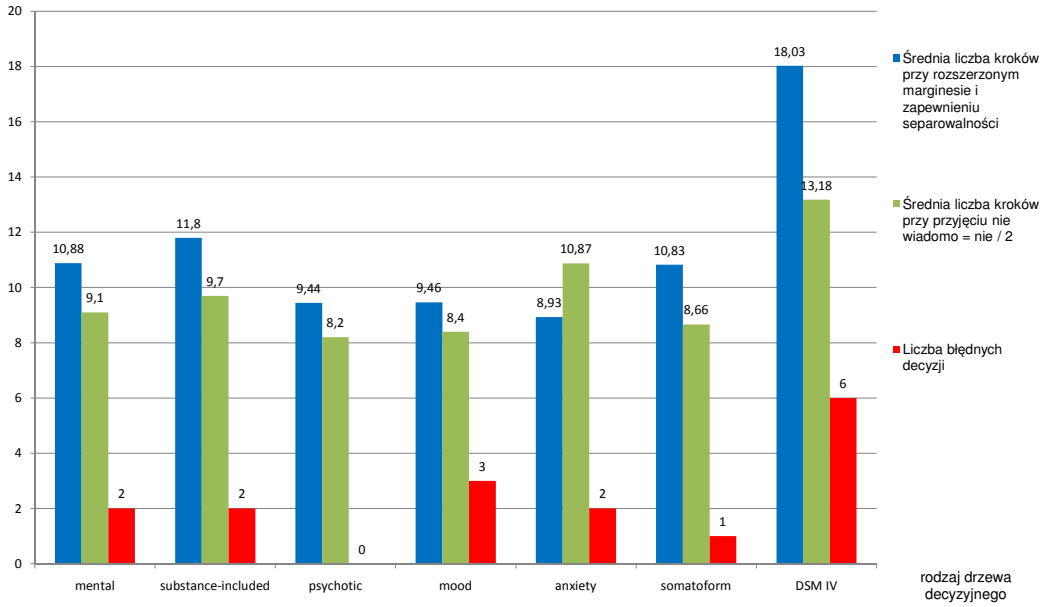
W przeprowadzonych badaniach użytkownik udzielając odpowiedzi na zadane pytanie (test cechy) dokonywał wskazania najbardziej prawdopodobnych jednostek chorobowych, które wyliczane były poprzez podobieństwo do wektora powstającego z udzielanych kolejnych odpowiedzi. Kolejne podprzestrzenie wyznaczane były przez odległość wektorów CDV (opisujących choroby) od wektora ANSW (3.8). Odległość ta wyznaczana mogła być jedynie w podprzestrzeni, dla której określone zostały wartości atrybutów w wektorach CDV. W drzewie DSM nie wszystkie cechy powiązane są ze wszystkimi



Rysunek 3.6: Legenda do wykresów przedstawionych na rysunku 3.7



Rysunek 3.7: Wpływ błędu użytkownika na średnią liczbę testów w poszczególnych poddrzewach DSM. Wyniki uzyskane dla algorytmu wyszukiwania kontekstowego ze zgadywaniem i bez (-h)



Rysunek 3.8: Wpływ przyjęcia nieznanych wartości atrybutu jako „nie/2” na szybkość wyszukiwania

chorobami. W efekcie tego, wektory CDV jednostek chorobowych posiadają braki wartości dla niektórych atrybutów. W eksperymencie poniżej dokonano klasyfikacji jednostek chorobowych, reprezentowanych przez pełne wektory CDV, uzupełniając automatycznie nieznane wartości atrybutów. W wypadku braku informacji o powiązaniu symptomu z chorobą, przyjmowane jest, że cecha ta nie stosuje się do obiektu. By odróżnić takie uzupełnienie od danych bezpośrednio wpisanych w bazie wiedzy dla braku informacji w wektorze CDV przyjęto, że wartość niewypełnionego atrybutu równa jest „nie/2”, co interpretowane jest jako: jeśli nie ma jawnie zapisanej informacji w wektorze CDV, to przyjmujemy, że taka cecha się raczej (nie/2) do obiektu nie stosuje. Podejście takie prowadzić może do przedwczesnych błędnych klasyfikacji, jak również do nierozdzielania chorób pod koniec procesu decyzyjnego. W związku z tym, ocena działania algorytmu, z tak przeprowadzonym dopasowaniem danych zrealizowana została dla przestrzeni z rozszerzonymi wektorami CDV poprzez dodanie nazw chorób jako cech (co zapewnia separowalność) oraz dla algorytmu z rozszerzonym marginesem błędu. Ponieważ przedwczesne zgadywanie w sytuacji przyjęcia dla nieznanych wartości cech wartości „nie/2” prowadziło do pogorszenia wyników klasyfikacji, zaprezentowane wyniki eksperymentów przeprowadzone zostały jedynie dla decyzji podejmowanej, gdy w podprzestrzeni pozostaje tylko jeden obiekt (bez zgadywania). Otrzymane rezultaty zaprezentowane zostały na rysunku 3.8.

Wyniki przy przyjęciu nieznanych wartości jako „nie/2", naniesione na słupkach kolorem zielonym pozwoliły uzyskać szybsze klasyfikacje niż podczas standardowego wyszukiwania (słupki niebieskie). Odbyło się to jednak kosztem jakości – pojawiły się błędne decyzje, których liczby określają wysokości słupków czerwonych. Obrazuje to sytuację, gdy zwrot (*recall*) zostaje poprawiony kosztem precyzji (*precision*) [12].

Przeprowadzone eksperymenty z algorytmem wyszukiwania kontekstowego demonstrują jego właściwości na przykładzie aplikacji w obszarze danych pozyskanych z DSM IV. Uzyskane rezultaty wskazują na możliwość zastosowania go w procesie wyszukiwania, gdzie poszukiwany obiekt jest określony zbiorem cech. W szczególności przeprowadzić można takie wyszukiwanie do wspomagania decyzji medycznej, gdzie diagnoza stawiana jest na podstawie odpowiedzi na wystąpienie określonych cech – objawów jednostki chorobowej.

Do praktycznych zastosowań konieczne byłoby zweryfikowanie tego podejścia poprzez konfrontację ze specjalistami z określonej dziedziny. Użyte dane medyczne posłużyły do wskazania właściwości algorytmu pozwalających na uwzględnienie przypadków, w których wyniki testów nie zgadzały się z wektorami cech opisującymi choroby, zapisanymi bazie wiedzy, czego nie realizują typowe algorytmy drzew decyzyjnych.

Przedstawione zastosowania pamięci semantycznej do wyszukiwania chorób poprzez ich symptomy są przykładem zadania dla uczenia maszynowego, w którym mamy wiele klas reprezentowanych przez pojedyncze obiekty. Klasa (decyzja klasyfikacyjna) jest tu reprezentowana jednym obiektem (jednostką chorobową), przez co są one tożsame. Przykład ten jest odmiennym do typowego zagadnienia klasyfikacji, w którym przeważnie szukamy jednej z kilku klas wielu obiektów.

3.4 Gra słowna

Algorytm wyszukiwania kontekstowego może znaleźć szereg zastosowań, np.: opisane wcześniej wspomaganie decyzji, wspomaganie klasyfikacji tekstów [51], bądź też może doprecyzowywać wyniki wyszukiwania w specjalistycznych wyszukiwarkach [30]. W zastosowaniu tym, posługując się siecią semantyczną związaną z określoną dziedziną można zadawać optymalnie wybrane, dodatkowe pytania precyzujące poszukiwane pojęcie i poprzez to związany z nią zbiór stron. Innym interesującym wariantem jest użycie pamięci semantycznej do identyfikacji obiektu oglądanego na obrazku, którego nazwa nie jest użytkownikowi znana. W tej sytuacji znalezienie nazwy za pomocą tradycyjnych wyszukiwarek staje się trudne. Powiązanie obrazu z jego opisem słownym jest postulowanym przez Paivio sposobem organizacji

wiedzy przez człowieka. Teoria podwójnego kodowania [65] jest próbą integracji podejścia pojęciowego i obrazowego, w której wykazane są wzajemne relacje i przejścia między tymi dwoma systemami reprezentacji. Pierwszy z nich oparty na sądach dysponuje czymś, co określone jest mianem *logogenów* – abstrakcyjnych pojęć, wyobrażeń odnoszących się do słów. Natomiast system oparty na obrazach wiąże tak zwane *immageny*, będące ogólnymi obrazami jakiejś klasy obiektów. Według Paivio istnieje sprzężenie między tymi dwoma sposobami reprezentacji tak, że podczas myślenia nieświadomie używamy obu sposobów kodowania. Powiązanie obrazu z systemem pojęciowym stanowi możliwość poprawy wyszukiwania w dużych bazach zawierających grafikę.

Zastosowanie algorytmu wyszukiwania kontekstowego wymaga bazy wiedzy o powiązaniach między pojęciami występującymi w eksplorowanej dziedzinie. Taką bazą wiedzy może być pamięć semantyczna, zawierająca informacje o powiązaniach między obiektami językowymi.

W tym rozdziale przedstawiona została koncepcja zrealizowanej na bazie wyszukiwania kontekstowego gry słownej w dwadzieścia pytań, demonstrującej kompetencje lingwistyczne pamięci semantycznej. W tradycyjnej wersji tej gry biorą udział dwie osoby: jedna wybiera obiekt, druga zadając pytania usiłuje odgadnąć, co przeciwnik ma na myśli. Idea ta posłużyła za przykład użycia pamięci semantycznej do przetwarzania symboli językowych. W jej komputerowej implementacji maszyna stara się odgadnąć pojęcie, które pomyślał człowiek. Odgadnięcie pojęcia z ograniczonej dziedziny odbywa się poprzez zadawanie użytkownikowi pytań, na które można odpowiadać: „tak", „być może", „nie wiem", „raczej nie", „nie", choć można wyobrazić sobie wiele innych wariantów tej gry.

W celu przetestowania podejścia wybrano dziedzinę królestwa zwierząt. Wiedza zawarta w tym obszarze jest wiedzą powszechną – dostępną praktycznie wszystkim ludziom, dlatego dobrze nadaje się do demonstracji możliwości językowych pamięci semantycznej.

Drugim argumentem za wyborem tej dziedziny jest dostępność (abstrahując tu od ich jakości) zasobów elektronicznych ustrukturalizowanej informacji dotyczącej pojęć z tej kategorii, pozwalających utworzyć podstawowy zasób wiedzy dla pamięci semantycznej.

Z założonego obszaru wiedzy uczestnik gry wybiera sobie obiekt (zwierzę), a komputer zadając coraz bardziej szczegółowe pytania usiłuje odgadnąć, co użytkownik ma na myśli. System na podstawie przeprowadzonych gier nabywa nową wiedzę i koryguje już posiadaną, tak że w efekcie końcowym w strukturach bazy wiedzy znajdują się informacje będące wypadkową wiedzy pojęciowej, z określonej dziedziny wspólnej wszystkim uczestnikom gry. Zastosowanie algorytmu wyszukiwania dla sieci semantycznej opisującej pojęcia

z królestwa zwierząt wraz z szablonami tworzącymi pytania z atomów wiedzy vwCRK pozwala zademonstrować możliwości językowe systemu informatycznego używającego numerycznej reprezentacji pojęć w postaci CDV. Kompetencje językowe wykorzystujące semantyczną organizację danych zależą nie od wydajności obliczeniowej maszyny cyfrowej, a od wiedzy przez nią posiadanej, pozwalającej na podstawie ogólnego opisu obiektu odnaleźć jego precyzyjną nazwę (np. termin techniczny lub nazwisko osoby). Ludzie znając kilka cech domyślają się, o jaki obiekt chodzi, chociaż dana nazwa nie została jeszcze jawnie wymieniona w czasie dialogu – dzięki temu potrafią zadawać sensowne pytania. Potrafią też wymienić większość cech związanych z danym przedmiotem. Antycypacja i wiarygodne uzupełnianie wypowiedzi o precyzyjne pojęcia jest jedną z ważniejszych funkcji, jakie musi spełniać każdy system do dialogu w języku naturalnym. Konieczne są do tego dobre modele pamięci semantycznej i pamięci epizodycznej człowieka [25].

Nazwa gry w dwadzieścia pytań pochodzi z faktu, że do zakodowania wyszukiwania miliona obiektów potrzebne jest binarne drzewo decyzyjne o głębokości 20. W sytuacji takiej, każdy test węzła dzieli zbiór najbardziej prawdopodobnych obiektów na połowę. W rzeczywistych zastosowaniach sytuacja taka oczywiście nie będzie występować, ponieważ powiązania cech z obiektami nie są równomierne (raczej nie zdarza się, by dla tej samej cechy było tyle samo powiązań pozytywnych, jak i negatywnych), jak również w bazie wiedzy występują braki. W przeprowadzonych eksperymentach najlepsze podziały pozwalały rozdzielić około 30% obiektów.

Istniejące w sieci podobne gry[3,4] opierają się na budowaniu macierzy obiektów i pytań używanych do wyszukania, stanowiących ich metodę reprezentacji wiedzy. Są to systemy, które nie korzystają z ogólnej wiedzy o pojęciach językowych i przez to nadają się tylko do jednego konkretnego zadania. Nietrudno sobie wyobrazić, że przy dużej liczbie użytkowników odwiedzających stronę WWW macierz obiektów i pytań zostaje dość szybko wypełniona. Podejście takie ma zasadniczą wadę: traci typ powiązania między obiektem a cechą (co w istniejących rozwiązaniach reprezentowane jest gotowym pytaniem). W związku z tym, przyjęty sposób zapisu wiedzy związany jest z konkretnym zastosowaniem, i powoduje zatracenie aspektu językowego, którym jest semantyczna reprezentacja wiedzy pozwalająca na wykorzystanie do różnych zadań. Dodanie nowego obiektu do pamięci semantycznej pozwala od razu na jej uwzględnienie w innych zastosowaniach, podczas gdy dodanie nowego obiektu w istniejących grach wymaga odpowiedzi na niektóre z zawartych tam specyficznych pytań. Możliwość użycia w

[3]http://www.20q.net ,XII 2007

[4]http://www.braingle.com/games/animal/index.php ,XII 2007

wielu zastosowaniach wpisuje system pamięci semantycznej w tzw. „szeroką sztuczną inteligencję" (*artificial general intelligence*) [96].

Implementacja gry w 20 pytań oparta na reprezentacji vwCRK jest ogólniejsza niż podejście używające macierzy obiektów i pytań. Pozwala na realizację innych programów wykorzystujących wiedzę o języku np. z użyciem vwCRK można przeprowadzić odwrotną do 20 pytań grę polegającą na generowaniu dla użytkownika zagadek dotyczących obiektów zapisanych w pamięci semantycznej.

Interakcja pamięci semantycznej z człowiekiem poprzez grę w 20 pytań umożliwia budowanie ontologii w dziedzinie, w której wyszukiwane są obiekty, co pokazane zostało w rozdziale 5. Do rozwoju pamięci semantycznej na szerszą skalę konieczne jest zbudowanie metaontologii. Pozwoli to rozbić całą przestrzeń semantyczną na podprzestrzenie o mniejszej liczbie wymiarów. Interesującym otwartym pytaniem pozostaje optymalny wymiar takich podprzestrzeni.

Implementacja gry zrealizowana została na bazie przedstawionego w paragrafie 3.2 algorytmu wyszukiwania kontekstowego, do którego wprowadzone zostały następujące modyfikacje:

1. Realizacja gry, w której obiekt jest odnajdywany poprzez cechy wskazywane największym zyskiem informacyjnym (3.4) powoduje, że w wypadku wielokrotnego poszukiwania tego samego obiektu, schemat wyszukiwania (kolejne pytania) za każdym razem będzie bardzo zbliżony. Wprowadzenie aktualizacji i reorganizacji wiedzy w pamięci semantycznej po każdej z gier (rozdział 5) pozwala dość szybko uzyskać dobre wyniki wyszukiwania. Zadawanie za każdym razem podobnych pytań, może prowadzić do znużenia użytkowników, jak również nie jest korzystne z punktu widzenia ilości wiedzy jaką system nabywa po każdej z gier, co dyskutowane jest w paragrafie 5.2.

   W celu uatrakcyjnienia gry, wprowadzono modyfikację polegającą na zmianie sposobu wyboru cechy do zadania zapytania. W pierwotnym podejściu pytania wynikają z obliczenia zysku informacyjnego dla każdej z cech (3.4) w podprzestrzeni gry O(A) i wybraniu z nich najlepszej. Wprowadzona modyfikacja do utworzenia zapytania wybiera cechę z prawdopodobieństwem proporcjonalnym do niesionego przez nią zysku informacyjnego (a nie jest wyznaczana bezpośrednio przez jego wartość). Sposób zmiany wyboru cechy pochodzi z metody reprodukcji ruletkowej używanej w algorytmach genetycznych. W podejściu tym prawdopodobieństwo wyboru z populacji łańcucha (reprezentującego rozwiązanie) jest proporcjonalne do jego jakości [35].

Podejście takie pozwala na wprowadzenie pewnej losowości zadawanych pytań – każda z gier, której celem jest odnalezienie tego samego obiektu może zadawać różne pytania, prowadzące do tego samego wyniku. Wybieranie cech z prawdopodobieństwem danym zyskiem informacyjnym pozwala na odnajdywanie szukanego obiektu w różny sposób, co nie było możliwe z użyciem maksymalizacji zysków informacyjnych cech dla, którego w każdym kroku gry dotyczącym tego samego obiektu padałoby to samo pytanie (oczywiście zakładamy, że użytkownik udzielałby takich samych odpowiedzi i nie zmienia się baza wiedzy). Użycie różnych pytań wynikłych z losowania cechy wpływać może niekorzystnie na szybkość zbieżności algorytmu do rozwiązania – zadawane zapytania o cechę nie są w tej sytuacji optymalne, a przestrzeń obiektów nie koniecznie musi być dzielona na najbardziej zbliżone połowie dwie części. Wyszukiwanie przez to jest wolniejsze, a w skrajnym wypadku może przekroczyć przyjęte założenie o zadawaniu maksymalnie dwudziestu pytań. W celu zabezpieczenia się przed tym niekorzystnym efektem, wybór cech do utworzenia zapytania odbywa się naprzemiennie: raz z użyciem wyżej opisanej metody losowania, a następnie z użyciem maksymalizacji zysku informacyjnego. Zmiana taka z jednej strony utrzymuje szybkość zbieżności algorytmu wyszukiwania do poszukiwanego obiektu, z drugiej strony wprowadza różnorodność zadawanych zapytań.

Obok uatrakcyjnienia gry, niedeterministyczne zadawanie pytań jest również korzystne z punktu widzenia weryfikacji i pozyskiwania wiedzy w przestrzeni semantycznej. Większa różnorodność pytań pozwala zweryfikować i pozyskać więcej wiedzy, co dyskutowane jest w paragrafach 5.1 i 5.2. W sytuacji, gdy nadrzędnym celem była precyzja wyszukiwania (w eksperymentach porównujących szybkość wyszukiwania algorytmu z wyszukiwaniem stosowanym przez ludzi – paragraf 5.3) randomizacja zapytań była wyłączana, co powodowało budowanie najlepszego drzewa klasyfikacyjnego, jednak ograniczało ilość wiedzy pozyskiwanej do pamięci semantycznej.

W zaimplementowanym rozwiązaniu losowe zadawanie pytań związane zostało z jakością pamięci (5.2). Przyjęto, że randomizacja pytań jest włączana, jeśli jakość pamięci liczona dla 100 ostatnio wykonanych gier jest $Q > 0,9$, co oznacza aktywowanie losowego zadawania pytań po przekroczeniu progu 90% poprawnych wyszukań.

2. Druga modyfikacja algorytmu wyszukiwania kontekstowego związana jest ze sposobem zawężania podprzestrzeni w kolejnych krokach gry.

Interakcja z wieloma użytkownikami wymaga dopuszczania zarówno pomyłek, jak i różnic w odpowiedziach pochodzących od różnych ludzi. Różne osoby mogą udzielić różnych odpowiedzi na to same pytanie związane z tym samym wyszukiwanym obiektem. Wynika to z odmiennych wyobrażeń o obiektach świata (a przez to i cechach je określających), jakie mają różni ludzie. Osobną kwestią jest również jakość wiedzy zapisana pierwotnie w bazie wiedzy systemu (pochodząca ze słowników maszynowych – rozdział 4). Może ona zawierać błędy, jak również i niedookreślenia, co w procesie uczenia – interakcji maszyny z człowiekiem jest korygowane i sprowadzane do średniej będącej wypadkową wiedzy wspólnej wszystkim ludziom.

Przy wyżej dyskutowanych uwarunkowaniach tworzenie podprzestrzeni O(A) używanej w kolejnych krokach algorytmu do wyznaczania najbardziej informacyjnej cechy, poprzez minimalizację odległości między wektorem odpowiedzi ANSW, a wektorami CDV reprezentujących obiekty w bazie wiedzy, może powodować nieodnalezienie obiektu będącej przedmiotem gry. W związku z tym, zmieniony został sposób wyznaczania kolejnych podprzestrzeni O(A). Dotychczas wykorzystywane prawdopodobieństwo (3.10) użycia obiektu w kolejnych krokach algorytmu oparte tylko na mierze odległości wektora CDV od wektora odpowiedzi ANSW rozszerzone zostało dodatkowo o margines błędu, który zmniejszany jest w funkcji kroków gry. Używając koncepcji pochodzącej z metody symulowanego wyrzażania [44] przyjęto, że użycie obiektu do utworzenia kolejnej podprzestrzeni O(A) określone jest prawdopodobieństwem danym rozkładem Boltzmanna 3.12, gdzie k jest związane ze schematem studzenia.

$$p(\Delta d, k) = \frac{1}{1 + \exp\left(\frac{\Delta d}{ck}\right)} \tag{3.12}$$

gdzie:

Δ d – jest przyrostem wartości odległości wektora CDV od ANSW liczonej od minimalnej wartości odległości (najbardziej prawdopodobnego obiektu)

k – jest krokiem gry

c – stała eksperymentalnie ustawiona na 0,2

3. Założenie, że obiekt musi zostać odgadnięty w dwudziestu pytaniach, dla małej liczby pojęć w przestrzeni semantycznej jest nadmiarowe.

Często się zdarza, że obiekt, który pomyślał użytkownik odnajdywany jest dużo szybciej, a kolejne pytania potwierdzają tylko, że jest on właściwy. Jedną z możliwości poprawienia szybkości wyszukiwania jest implementacja heurystyki pozwalającej na zgadywanie poszukiwanego obiektu wcześniej niż przed zadaniem 20 pytań. Istnieje kilka możliwości modyfikacji warunku stopu algorytmu:

- przyjąć można, że gra jest kończona, gdy w podprzestrzeni znajduje się tylko jeden obiekt. Podejście takie może jednak generować błędy w przypadku, gdy jest mała baza wiedzy, a odnalezienie obiektu zaszło bardzo szybko – w kilku krokach, poprzez faworyzowanie lepiej opisanych obiektów (mających więcej cech w wektorach CDV), dlatego czasami warto zadać jest jeszcze jedno pytanie.
- rozwiązaniem wcześniej opisanej niedogodności jest sprawdzanie, czy w kolejnych krokach gry nie zmieniają się obiekty w podprzestrzeni O(A) i w sytuacji, gdy dwie kolejne podprzestrzenie są takie same, przerwanie wyszukiwania, co będzie implikować zgadywanie najbardziej prawdopodobnego obiektu. Rozwiązanie takie może generować nieprawidłowe wskazania przy wprowadzeniu zwiększonego marginesu błędu, dopuszczającego do podprzestrzeni O(A) niezgodne z odpowiedziami użytkownika obiekty.
- przyjętym rozwiązaniem jest sprawdzanie, czy w kolejnych krokach algorytmu można zidentyfikować najlepszy obiekt, różniący się znacząco od innych, w podprzestrzeni O(A), i jeśli w kolejnych podprzestrzeniach prawdopodobieństwo to wzrasta, można przyjąć, że jest on przedmiotem gry.

Analiza odległości najbardziej prawdopodobnego obiektu od następnego po nim, w kolejnych krokach gry pozwala ocenić, czy można zgadywać, że najlepiej oceniony (mający największe prawdopodobieństwo) obiekt jest tym, którego poszukuje użytkownik.

Zgadywanie obiektu użyte w grze zrealizowane zostało następująco:

(a) jeśli najlepiej oceniony obiekt wyraźnie odstaje od innych,

(b) i odstawanie to w kolejnych krokach się nie zmniejsza,

(c) wtedy, zgaduj najlepszy obiekt z prawdopodobieństwem określonym krokiem gry i wartością określoną odległością od drugiego najbardziej prawdopodobnego obiektu.

Za warunek wyzwalający zgadywanie przyjęto miarę odróżnienia najlepszego obiektu od innych określonego jako wartość odchylenia standardowego w podprzestrzeni zawierającej najbardziej prawdopodobne obiekty.

$$d_p = \Delta(d_{min+1} - d_{min}) > \text{std}(O(A)) \tag{3.13}$$

gdzie:

d_{min} jest najmniejszą odległością między CDV (najbardziej prawdopodobnego obiektu), a ANSW

d_{min+1} jest odległością między CDV obiektu kolejnego a ANSW

std$(O(A))$ jest standardowym odchyleniem odległości w zbiorze najbardziej prawdopodobnych obiektów O(A)

W sytuacji takiej podjęcie przez system decyzji o zgadywaniu odbywa się z prawdopodobieństwem, które przyjęto, że jest opisane rozkładem normalnym:

$$p(d_p, k) = \frac{1}{k\sqrt{2\pi}} \exp\left(-\frac{d_p^2}{2k^2}\right) \tag{3.14}$$

Rosnąca liczba kroków algorytmu i zmniejszająca się odległość między najbardziej prawdopodobnym obiektem, a pozostałymi w podprzestrzeni powodować będzie zatrzymanie gry z prawdopodobieństwem określonym 3.14, i zadanie zapytania, czy rezultatem wyszukiwania jest najbardziej prawdopodobny obiekt z podprzestrzeni gry O(A). Przyjęto, że do aktywacji zgadywania prawdopodobieństwo dla najlepszego obiektu musi być $> 0,7$.

Modyfikacja umożliwiająca wcześniejsze zgadywanie zmniejszać będzie ilość wiedzy płynącą do systemu w wyniku interakcji z człowiekiem. Wynika to z mniejszej liczby cech, dla których użytkownik udzielił odpowiedzi, o które modyfikowany jest opis w przestrzeni semantycznej wyszukiwanego obiektu. Dlatego używanie zgadywania nie jest korzystne z punktu widzenia akwizycji wiedzy, jak również ze względu, że zgadywanie obiektu może prowadzić do niepoprawnego wyniku wyszukiwania, który dzięki dodatkowym pytaniom mógłby być inny, w szczególności poprawny. Zgadywanie wyszukiwanego obiektu pokazuje potencjalną możliwość zwiększenia szybkości wyszukiwania kontekstowego, co zostało użyte do porównania z wyszukiwaniem przez ludzi (opisanego dalej na rysunku 5.4).

Podejście to może być stosowane dla dobrze opisanej sieci semantycznej, gdzie liczba błędnych klasyfikacji będzie niewielka. W implementacji użycie zgadywania związane zostało z jakością pamięci, analogicznie, jak w przypadku aktywacji randomizacji zapytań.

Zanim przedstawione zostaną wyniki zastosowania wyszukiwania kontekstowego do implementacji gry w dwadzieścia pytań omówiony zostanie sposób pozyskiwania wiedzy do pamięci semantycznej, gdyż przede wszystkim od tej wiedzy zależy jakość działania systemu.

Rozdział 4

Pozyskiwanie asocjacji z zasobów cyfrowych

Użyteczność algorytmu wyszukiwania kontekstowego w dużym stopniu zależy od wiedzy w pamięci semantycznej opisującej dziedzinę, która ma być eksplorowana. Akwizycja wiedzy jest bardzo istotnym zagadnieniem przy tworzeniu systemu ekspertowego [59], a pozyskiwanie wiedzy kontekstowej jest zasadniczym problemem w systemach używających semantyki. Jedną z możliwości akwizycji typowanych powiązań między pojęciami, w obrębie wybranej dziedziny jest pozyskiwanie wiedzy poprzez interakcję z ludźmi. Propozycja zastosowania kooperacyjnego pozyskiwania wiedzy przedstawiona została przez utworzenie projektu portalu do gry w 20 pytań (opisanego w załączniku A).

Drugie podejście oparte jest na automatycznym pozyskaniu wiedzy z zasobów cyfrowych: słowników maszynowych (ustrukturalizowanych), definicji encyklopedycznych (sformatowanych), otwartych tekstów (tekst swobodny). Zasadniczym problemem w tym podejściu jest jakość wiedzy zapisana w postaci elektronicznej. Ustrukturalizowane słowniki maszynowe ogólnego przeznaczenia zawierają dużo informacji nadmiarowej, jak również często brak jest w nich informacji oczywistych dla człowieka, dodatkowym kłopotem są również istniejące w nich błędy. Problemy te wynikają przede wszystkim z powodu, że maszynowe słowniki ogólnego zastosowania powstają w oderwaniu od aplikacji, która wnosiłaby informację zwrotną o jakości zgromadzonych danych, przez co umożliwiając ich korektę. W słownikach maszynowych brak jest również możliwości prostego tworzenia filtrów pozwalających określić rodzaj interesującej informacji. Istniejące słowniki maszynowe dla dedykowanych zastosowań są przeważnie albo bardzo specjalistyczne, albo zawierają wiedzę, której sposób reprezentacji praktycznie ogranicza je do jednego zastosowania.

Jeszcze większy kłopot jest z wiedzą w postaci swobodnego tekstu, którego samo przetworzenie niesie za sobą szereg zagadnień, które muszą zostać wzięte pod uwagę, np. obok zbudowania złożonego algorytmu do automatycznego pozyskania wiedzy, konieczność jej dalszej, ręcznej korekty.

W tym paragrafie opisane zostało podejście użycia istniejących zasobów cyfrowych do wypełnienia pamięci semantycznej wiedzą kontekstową.

Uzyskane wyniki w algorytmie wyszukiwania dla ograniczonej dziedziny pozwalają ocenić przydatność danych pozyskanych z ustrukturalizowanych zasobów semantycznych do budowy pamięci semantycznej na szeroką skalę.

Paragraf kolejny opisuje podejście do pozyskania wiedzy z otwartego tekstu, zorganizowanego w zasobach encyklopedycznych, sformatowanych w postaci pojęcie – definicja.

4.1 Wordnet

Jednym z najpopularniejszych istniejących zasobów semantycznych jest projekt WordNet [58], będący leksykalną bazą danych języka angielskiego. Inicjatywa ta rozwijana jest od 1998 roku na uniwersytecie Princeton, gdzie prowadzone są prace nad utworzeniem takiego słownika dla języka angielskiego. Równocześnie na świecie prowadzonych jest szereg zbliżonych projektów rozwijających taką sieć semantyczną dla innych języków. Słownik taki, również powstaje i dla języka polskiego[1].

Organizacja wiedzy w słowniku WordNet oparta jest na znaczeniach – *synsetach* elementarnych jednostkach wiedzy składających się z definicji i zbioru słów z nią skojarzonych, będących jednocześnie swoimi synonimami. Synsety wiążą się ze sobą określonymi typami relacji i dzięki temu tworzą sieć znaczeniową. Dodatkowo, w słowniku zdefiniowane są powiązania między słowami pozwalające doprecyzować słowa, w kontekście określonego znaczenia [87].

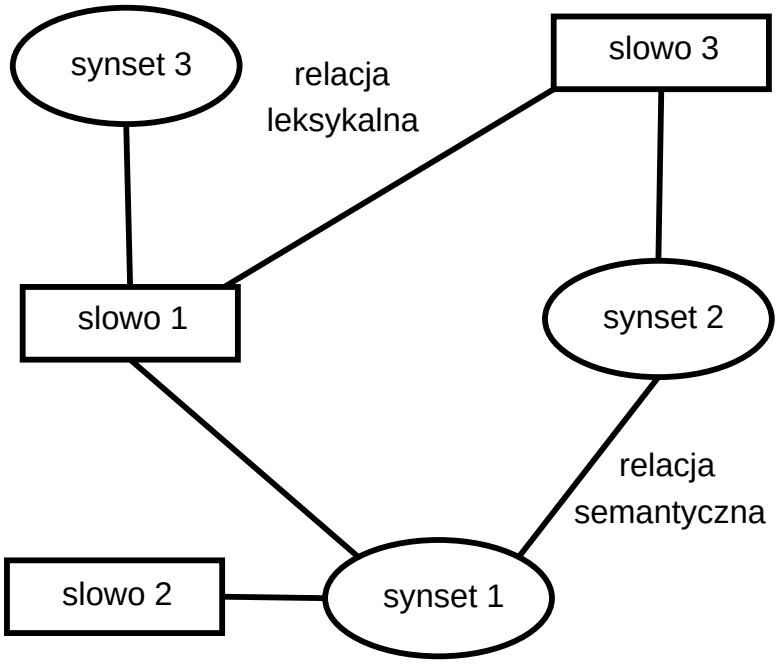
Omawiając użycie WordNeta dla pamięci semantycznej, wspomnieć należy, że komponenty, z których zbudowana została pamięć semantyczna (wizualny edytor sieci powiązań, zaprezentowany na rysunku A.2), posłużyły również do budowy interaktywnej wizualizacji[2] tego słownika [88], która została umieszczona na liście projektów stowarzyszonych[3].

Wybór słownika WordNet podyktowany został jego dostępnością oraz zastosowaniami w różnorodnych projektach lingwistycznych, które znalazły

[1] http://plwordnet.pwr.wroc.pl/browser/ ,II 2008

[2] http://wordventure.eti.pg.gda.pl ,II 2008

[3] WordNet related projects: http://wordnet.princeton.edu/links ,II 2008



Rysunek 4.1: Schemat sposobu reprezentacji wiedzy w słowniku WordNet

również zastosowanie praktyczne. Przykładem jest tu użycie słownika WordNet do osiągnięcia komercyjnego sukcesu firmy Google opartego na reklamach AdSense[4]. Przykład tego zastosowania wskazuje na potrzebę istnienia i potencjalną możliwość komercjalizacji sieci semantycznych opisujących język.

Fragment semantycznego słownika WordNet, po konwersji w struktury danych pamięci semantycznej użyty został do oceny przydatności bazy leksykalnej uniwersytetu Princeton, jako źródła wiedzy mogącego posłużyć do zrealizowania algorytmu wyszukiwania kontekstowego na szeroką skalę.

Poniżej opisana została procedura zamiany słownika na struktury danych pamięci semantycznej, oraz ocena ich jakości w aplikacji do gry w dwadzieścia pytań.

4.1.1 Konwersja danych systemu WordNet w struktury pamięci semantycznej

Elementarne jednostki wiedzy składowane w słowniku WordNet schematycznie przedstawione mogą zostać, jako struktura zaprezentowana na rysunku 4.1. Wyróżnić można tu słowo oraz znaczenie – synset. Między synsetem a słowem występują powiązania organizujące słowa w zbiory znaczeń – two-

[4]http://en.wikipedia.org/wiki/AdSense ,II 2008

rzące w ten sposób grupy synonimów (na rysunku 4.1 synonimami są słowo 1 i słowo 2, słowo 1 dodatkowo występuje w dwóch znaczeniach – w kontekście synsetu 1 i synsetu 3 – w sytuacji tej jest ono homonimem).

Istniejące w słowniku WordNet typy relacji można podzielić na dwie grupy:

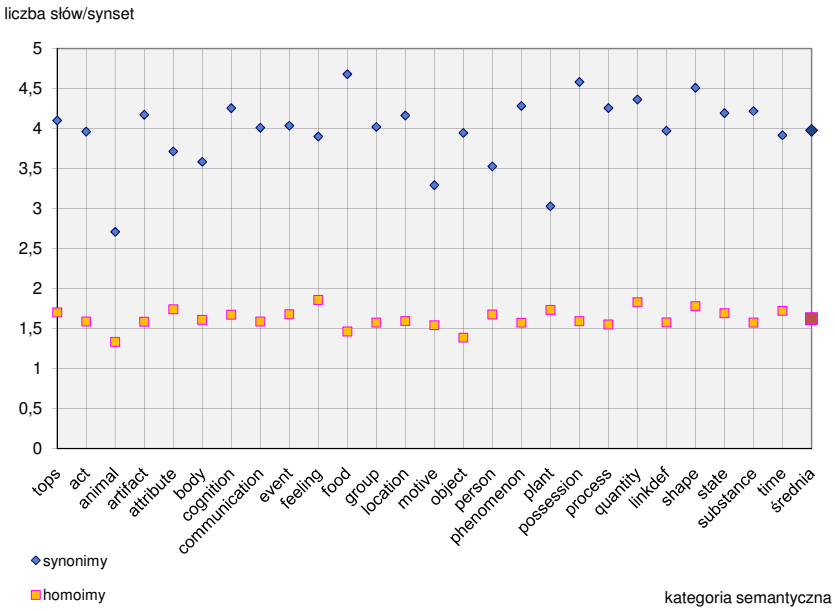
1. **relacje leksykalne** występujące pomiędzy słowami, w kontekście określonych znaczeń (powiązań z synsetami),

2. **relacje semantyczne** występujące między synsetami.

Każda z tych grup relacji posiada swoje określone typy pozwalające dookreślić rodzaj powiązania zachodzącego między elementarnymi jednostkami wiedzy w słowniku (słowami bądź synsetami). Struktury danych użyte do składowania informacji w systemie WordNet różnią się od przyjętych dla pamięci semantycznej. Podstawową różnicą jest brak w pamięci semantycznej osobnej struktury do składowania znaczenia, a operowanie tylko na słowach tworzących elementarną jednostkę pojęciową, reprezentowaną przez wektor CDV. Jedną z prostych możliwości rozwiązania tego problemu byłoby utworzenie ze wszystkich synonimów i definicji synsetu jednego opisu, funkcjonującego w pamięci semantycznej jako pojedyncze słowo. Rozwiązanie takie, jednak znacznie utrudnia automatyczne generowanie z vwCRK zdań formułowanych jako elementy dialogu z użytkownikiem w formie, która ma być zbliżona do języka naturalnego oraz praktycznie uniemożliwia przetwarzanie statystyczne słów. Dlatego konieczne jest przy pozyskiwaniu danych z bazy WordNet zastosowanie algorytmu konwertującego znaczenia na słowa oraz przyjęcie założeń odnośnie sposobu rozwiązania niejednoznaczności słów [89].

Aby można przenieść dane z bazy WordNet przyjęto, że w obrębie jednej dziedziny, określonej kategorią semantyczną, znaczenia muszą być jednoznacznie identyfikowane poprzez pojedyncze słowo. Identyfikacja typu semantycznego, pozwalającego określić poddziedzinę wyznaczającą zbiór używanych w niej słów, odbyła się poprzez znacznik istniejący w słowniku WordNet *semantic category*. Założenie o jednoznaczności pojęcia i słowa w obrębie określonej kategorii semantycznej pozwala dopasować struktury danych bazy WordNet do reprezentacji wiedzy użytej w pamięci semantycznej. Wymaga ono jednak rozwiązania konfliktów wynikłych z istnienia homonimów oraz dla zbioru synonimów wskazanie jednego słowa, najlepiej opisującego znaczenie.

Słowa w obrębie synsetów w poszczególnych kategoriach semantycznych nie są jednoznaczne. Obrazuje to wykres na rysunku 4.2 przedstawiający liczby synonimów i homonimów w poszczególnych, rzeczownikowych kategoriach semantycznych.



Rysunek 4.2: Liczby synonimów i homonimów systemie WordNet, dla synsetów rzeczownikowych, w podziale na poszczególne kategorie semantyczne

Liczba słów w synsetach opisujących poszczególne znaczenia w kategoriach rzeczownikowych przedstawiona na wykresie pokazuje częste występowanie synonimów (średnio 3,98 słowa w synsecie), oraz wieloznaczeniowość rzeczowników średnio 1,67 słowa/znaczenie. W związku z tym, konieczne jest wyznaczenie etykiety (słowa najlepiej reprezentującego znaczenie) dla każdego z synsetów. Identyfikacja ta przeprowadzona została z użyciem algorytmu ujednolicenia słów i znaczeń przebiegającego następująco:

1. Uszereguj synonimy synsetu w kolejności od najbardziej deskryptywnego słowa do najmniej. Uporządkowanie zrealizowane zostało z wykorzystaniem pola z systemu WordNet *tagcount* dołączanego do każdego z synsetów. Pozwala on określić, które słowa są bardziej istotne dla danego synsetu.

2. Uszereguj synsety w kolejności ich popularności. (Przyjęta miara popularności opisana jest dalej (4.1))

3. Dla kolejnych synsetów z powyższej listy przypisz słowo będące pierwszym na liście jego synonimów, a następnie usuń to słowo ze wszystkich innych synsetów.

Otrzymane w ten sposób jednoznaczne powiązanie synset – słowo pozwala na znalezienie pojedynczych słów, które użyte są dalej, jako identyfikator

znaczenia. W wyniku przeprowadzenia takiego dopasowania część z synsetów (bliskoznacznych względem siebie) pozostała bez słów je opisujących. Znaczenia takie, ze względu na przyjętą ich małą istotność z punktu widzenia wiedzy ogólnej zostały zastąpione przez najbardziej bliskoznaczny im synset, do którego dołączone zostały powiązania z usuwanego znaczenia. Proces ten wzbogacony o przepisywanie definicji między synsetami może zostać użyty, jako metoda klasteryzacji znaczeń słownika WordNet, alternatywna np. do CoreLex [13], pozwalająca grupować synsety o najbardziej zbliżonym znaczeniu.

W przedstawionej koncepcji algorytmu przypisania jednoznacznych słów do synsetów użyta została miara określająca popularność sensu. Utworzona miara oparta została na założeniu, że synset jest tak popularny, jak najpopularniejsze słowo w zbiorze synonimów go opisujących. Innym rozwiązaniem byłoby przyjęcie ważonej sumy dla popularności słów w synsecie i ich normalizacja, jednak rozwiązanie takie dało słabsze rezultaty podczas wstępnej oceny wyników uzyskanych dla wybranych znaczeń. Brak dobrze określonej miary istotności synsetu, i popularności słów w określonym kontekście jest istotnym brakiem systemu WordNet.

4.1.1.1 Utworzenie miary popularności słowa

Zagadnienie popularności słów języka naturalnego jest szeroko dyskutowanym problemem w kręgach lingwistycznych [5]. Przy przetwarzaniu języka naturalnego najczęściej przyjmuje się, że popularność danego słowa jest jego częstością występowania w określonym języku. Jest to pewne przybliżenie bardzo subiektywnej oceny – popularności pojęć, jako że różne osoby mogą mieć różne osądy odnośnie tego, które słowo jest bardziej powszechne od innych. Wynikać może to z uwarunkowań socjologicznych, jak i z wiedzy oceniającego. Trzeba mieć również na uwadze, że częstości wystąpień słów będą inaczej wyglądać dla języka mówionego, jak i pisanego [39]. Utworzona miara popularności słowa opisana została formułą 4.1.

$$Rank_{word} = \frac{IC + GR + BNC}{\max\left(Rank\right)} \tag{4.1}$$

Wartość opisująca popularność słowa jest złożeniem trzech miar. Każda z uzyskanych składowych została znormalizowana, tak że wszystkie mają jednakowy wpływ na wartość miary. Poszczególne składowe uzyskano z następujących źródeł częstości wystąpień słów:

1. IC – *Information Content*. Miara oparta została na liczbie wystąpień określonego słowa w opisach synsetów w bazie WordNet. Uzyskanie do-

brej miary tego typu wymagało użycia *tokenizera*[5] do podzielenia poszczególnych definicji słownika na pojedyncze słowa, a następnie aplikacji dla każdego z nich *stemmera*[6], tak by otrzymać podstawową formę słowa, występującą w otwartym tekście opisu znaczenia.

2. GR – *Google Rank*. Miara oparta została na wyszukiwarce Google. Popularność słowa określona została, jako liczba stron zwracanych przez wyszukiwarkę dla danego słowa, będącego kierowanym do niej zapytaniem. Uzyskanie tej miary wymagało implementacji *crawlera* – programu wykonującego zapytania do wyszukiwarki Google, kolejno o wszystkie słowa występujące w słowniku WordNet. Następnie, ze zwróconych wyników wyekstrahowana została informacja dotycząca liczby stron, na której wystąpiło podane słowo.

3. BNC – *British National Corpus*. Dane służące do utworzenia tej miary zostały oparte na zasobach British National Corpus[7]. Korpus ten zawiera 100 milionów słów języka angielskiego, zarówno pisanego jak i mówionego. W BNC dostępna jest lista częstości występowania zadanego słowa w zasobach tekstowych oraz liczba zasobów, w których zadane słowo występuje. Iloczyn tych dwóch wartości został przyjęty jako użyta miara BNC.

Źródła tekstowe użyte do utworzenia tej miary wiążą ją z językiem angielskim. Arbitralnym również pozostaje sposób ich doboru, który dokonany został ze względu na następujące przesłanki:

- opracowana miara użyta zostanie do przetworzenia zasobów słownika WordNet,
- wyszukiwarka Google jest największą składnicą informacji tekstowej,
- BNC jest jedną z największych baz statystyk częstości słów.

4.1.2 Utworzenie przestrzeni semantycznej z bazy WordNet

Synsety w systemie WordNet powiązane są z innymi określonym typem relacji. Do utworzenia przestrzeni semantycznej Ψ_{wn} z bazy WordNet, opisującej obiekty pamięci semantycznej wykorzystanych zostało 5 wybranych typów relacji semantycznych, wiążących synsety typu rzeczownikowego:

[5] http://opennlp.sourceforge.net/ ,VII 2007
[6] http://tartarus.org/∼martin/PorterStemmer ,VII 2007
[7] http://www.natcorp.ox.ac.uk/ ,XII 2007

- Hypernimy i hyponimy – tworzące taksonomie
- 3 typy meronimów (-part, -member, -substance meronim) – tworzących powiązania określające części składowe

Pierwsza grupa typów relacji tworzy strukturę taksonomii, dostarczając w ten sposób szkieletu do dziedziczenia cech. W zależności od typu meronimii powiązania te zawierają dodatkowe meronimy wynikłe z przechodniości relacji, przez co tworzą bogatsze zbiory cech związane z określonym pojęciem. Zagnieżdżona meronimia wymaga wprowadzenia w obrębie pamięci semantycznej specjalnej procedury obsługi tego typu relacji, dzięki której wektory CDV pozyskują cechy wynikłe z zawierania się meronimów w sobie. Procedura ta realizowana jest podczas zamiany reprezentacji wiedzy z sieci semantycznej ζ na przestrzeń cech Ψ.

Dużą wadą systemu WordNet, w użytej wersji 2.1 jest brak powiązań między rzeczownikami a czasownikami, jak również miedzy rzeczownikami a przymiotnikami. Brak tego typu powiązań powoduje, że utworzone wektory CDV zawierają cechy, które są głównie rzeczownikami. Wzbogacenie słownika WordNet o te powiązania, planowane przez jego twórców w przyszłości, co na pewno znacznie wpłynie na poprawę jakości danych w nim składowanych.

Rozmiar początkowej sieci semantycznej ζ_{wn} dla kategorii *noun.animals* wynosił 7457 pojęć i 8152 cech je opisujących 1.172.977 powiązaniami. Występująca tu duża liczba obiektów wynika z istnienia specyficznych określeń, nazw łacińskich oraz zbliżonych do siebie znaczeniowo pojęć. Z punktu widzenia wyszukiwania popularnych obiektów utworzona z tych danych przestrzeń semantyczna Ψ_{wn} zawiera również dużo szumu – istnieją tu informacje nie będące oczywiste dla przeciętnego użytkownika języka, dlatego użycie jej wymaga wykonania szeregu dopasowań. Proces importu danych do pamięci semantycznej przebiegał następująco:

1. Utworzenie taksonomii.

   Dla zbioru pojęć należących do wskazanej kategorii semantycznej *noun.animal* utworzona została hierarchia na podstawie relacji hypernimicznych. Pozwoliły one na zorganizowanie pojęć w strukturę 7457 obiektów powiązanych z 1367 pojęciami nadrzędnymi opisaną 77154 powiązaniami, posiadająca średnio 10,34 cech na obiekt. W taksonomii tej, 6234 pojęć należało do gałęzi drzewa hierarchii, którego średnia głębokość wynosiła 10,42. Tak duża głębokość taksonomii wynika ze sposobu pozyskiwania danych z systemu WordNet, odbywająca się dla każdego pojęcia z wybranej kategorii semantycznej do najwyższego węzła w słowniku. Dla testowanej kategorii semantycznej zbiór pojęć

został ograniczony przez synset *animal* – wszystkie obiekty nie mające powiązania z tym synsetem zostały usunięte. Dopasowanie to zmniejszyło głębokość taksonomii do 6,41, a średnia liczba cech opisujących obiekt wyniosła 7,0.

2. Utworzenie zbioru cech.

   Otrzymana taksonomia stanowi szkielet wiedzy, po którym są dziedziczone cechy z pojęć nadrzędnych do podrzędnych. Cechy takie pozyskane zostały z powiązań meronimicznych sytemu WordNet. Dla pojęć należących do kategorii semantycznej *body part* wyróżnionych zostało 361 cech wiążących się z 4636 pojęciami kategorii semantycznej *noun.animal* przez 68073 powiązań. Dało to średnią liczbę cech dla pojedynczego obiektu posiadającego swój meroniczny opis 14,68, a dla wszystkich obiektów w bazie wiedzy 9,1 cech na obiekt.

3. Zawężenie zbioru obiektów.

   Ponad 6 tysięcy pojęć występujących w jednej, testowanej kategorii semantycznej jest liczbą znacznie przekraczającą wiedzę przeciętnego człowieka. Zmniejszenie liczby pojęć dokonane zostało z użyciem opracowanej miary popularności słowa, oraz poprzez zgrupowanie obiektów najbardziej podobnych do siebie – przyjęto, że jeśli obiekty (reprezentowane przez CDV) różnią się jedną cechą, to są identyczne.

   Do testu algorytmu gry w 20 pytań wybranych zostało $\sim$10% najbardziej popularnych pojęć. Poniżej zaprezentowano pierwsze dziesięć pojęć, które zostały odrzucone (znalazły się poniżej przyjętego 10% progu najbardziej popularnych obiektów):

   water flea ; wahoo ; white whale ; yellow jacket ; nematoda ; river horse ; dominick ; rock bass ; blastocyst ; escherichia coli

   Pojęcia na tej liście nie są powszechnie znane, co wskazuje na potencjalną słuszność przyjętego ograniczenia, które zweryfikowane zostało podczas testów wyszukiwania.

4. Redukcja cech i końcowa struktura zmodyfikowanej przestrzeni semantycznej

   Zawężony zbiór obiektów i połączona taksonomia z relacjami meronimów utworzyły przestrzeń semantyczną Ψ_{wn} opisującą 676 obiektów 938 cechami. Zamiana sieci semantycznej ζ_{wn} na jej geometryczną reprezentację dała średnią liczbę cech w wektorach $\mathrm{CDV}_\rho = 25,03$. Duża

liczba cech wchodzących w skład wektorów CDV wynika z podobnych, jak wymienione przy zawężaniu zbioru obiektów, właściwości słownika WordNet: część użytych cech nie jest wiedzą używaną potocznie. Użycie mało znanych cech podczas szukania kontekstowego skutkować będzie częstymi odpowiedziami „nie wiadomo", co jest niekorzystne z punktu widzenia szybkości odnajdywania właściwego obiektu. Dlatego dokonane zostało dopasowanie polegające na usunięciu ∼20% najmniej znaczących (mających najmniejszą wartość popularności) cech. Spowodowało to zmniejszenie średniej liczby cech w przestrzeni semantycznej do $\mathrm{CDV}_\rho = 23,62$.

Pozyskane do pamięci semantycznej dane ze słownika WordNet umożliwiły utworzenie wektorów CDV dla ustalonego również na podstawie tego słownika zbioru 676 obiektów. Otrzymane w procesie importu przykładowe wektory CDV opisujące 4 popularne obiekty przedstawione zostały w tabeli 4.1.

4.1.3 Przykłady gier i ocena jakości

Ocena jakości przestrzeni semantycznej oparta została na algorytmie wyszukiwania kontekstowego, zastosowanego do przeprowadzenie gry w 20 pytań. Jakość przestrzeni określona została jako proporcja wszystkich przeprowadzonych gier, do gier zakończonych sukcesem. Ocena jakości pamięci dyskutowana jest szerzej w paragrafie 5.2.2. Za miarę błędu pamięci semantycznej przyjęto proporcję miedzy wyszukaniami zakończonymi sukcesem N_S, a liczbą wszystkich przeprowadzonych wyszukań N. Wartość błędu E opisana jest formułą 4.2.

$$E = 1 - \frac{N_S}{N} \tag{4.2}$$

Ponieważ głównym celem opisanych w tym rozdziale eksperymentów było przetestowanie przydatności danych ze słownika WordNet do zastosowania w grze w 20 pytań, badania przeprowadzane zostały dla algorytmu wybierającego cechę do utworzenia pytania, poprzez maksymalizację zysku informacyjnego. Pozwala to przeprowadzać w miarę jednorodne gry – do kolejnych wyszukań tego samego obiektu używane będą takie same pytania, chyba że aktualizacja bazy wiedzy zmieni ranking zysków informacyjnych dla cech.

Przeprowadzenie procedury testowej pamięci semantycznej polegało na wyszukaniu losowo wskazanych obiektów. W celu szybkiego oczyszczenia pamięci semantycznej, obiekty do wyszukiwania wybierane zostały z prawdopo-

Puma	Giraffe	Cobra	Butterfly
is_a animal	is_a animal	is_a animal	is_a animal
is_a being	is_a being	is_a being	is_a arthropod
is_a carnivore	is_a creature	is_a creature	is_a being
is_a cat	is_a living entity	is_a living entity	is_a creature
is_a creature	is_a fauna	is_a fauna	is_a living entity
is_a living entity	is_a mammal	is_a object	is_a fauna
is_a fauna	is_a object	is_a organism	is_a insect
is_a feline	is_a organism	is_a reptile	is_a invertebrate
is_a mammal	is_a placental	is_a serpent	is_a object
is_a object	is_a vertebrate	is_a snake	is_a organism
is_a organism	has belly	is_a vertebrate	has ala
is_a placental	has body part	has belly	has body part
is_a vertebrate	has cannon	has body part	has cell
is_a wildcat	has cell	has cell	has cuticle
has belly	has chest	has chest	has face
has body part	has coat	has costa	has foot
has cell	has costa	has digit	has head
has chest	has digit	has face	has shell
has coat	has face	has head	has shield
has costa	has hair	has rib	has thorax
has digit	has head	has tail	has wing
has face	has hock	has thorax	
has hair	has hoof		
has head	has rib		
has paw	has shank		
has rib	has tail		
has tail	has thorax		
has thorax			

Tabela 4.1: Przykładowe wektory CDV uzyskane z bazy WordNet

dobieństwem danym popularnością słowa opisującego obiekt, co powoduje, że do gry wybierane były najbardziej znane zwierzęta. Podejście to pokazuje możliwość zastosowania danych z bazy WordNet do wyszukiwania najlepiej znanych obiektów. Ocena średnio-dobrze opisanych obiektów pamięci semantycznej przeprowadzona została przy ocenie pamięci już wytrenowanej (paragraf 5.2.2).

Oczyszczanie pamięci semantycznej zrealizowane zostało przez wprowadzenie rozszerzenia polegającego na usuwaniu cech, dla których użytkownik wyszukujący testowy obiekt podał odpowiedź „nie wiem". Modyfikacja ta aktywna jest tylko podczas procedury testowej, oceniającej jakość pamięci dla 10 wylosowanych, testowych obiektów. Ponieważ aktywność taka mocno modyfikuje bazę wiedzy, dostęp do niej ma jedynie administrator pamięci semantycznej, co zapewnia ochronę przed nadmiernym usunięciem cech z pamięci semantycznej. Usuwanie cech skutkuje zmniejszeniem średniej liczby cech w wektorach CDV, jednak pozwala na oczyszczenie wiedzy w strukturach danych pamięci semantycznej. W efekcie końcowym prowadzić będzie to do poprawienia szybkości i jakości wyszukiwania. W trakcie realizacji procedury testowej przyjęto, że wyszukiwanie odbywa się do momentu, gdy w podprzestrzeni O(A) pozostanie jeden obiekt, lub gdy zostaną wyczerpane cechy, co wymusza zgadywanie najbardziej prawdopodobnego obiektu.

Poniżej umieszczono kilka przykładów przeprowadzonych gier w 20 pytań z użyciem danych ze słownika WordNet. Uproszczony format prezentacji przebiegu gry pomija moduł generujący z vwCRK zapytanie do użytkownika w formie zbliżonej do języka naturalnego, przedstawiając jedynie użytą cechę i typ relacji z atomu wiedzy, które podane zostały w nawiasach kwadratowych. Następująca po nim litera określa odpowiedź użytkownika, dla której przyjęto oznaczenia: Y – tak, Y/2 – może być, N/2 – raczej nie, N – nie, R – nie wiadomo (skutkujące usunięciem cechy). W przedstawionych przykładach, obok usuwania cechy, użyte zostały mechanizmy korekcji bazy wiedzy po każdej z gier, opisane w paragrafie 5.1.

Puma: [has chest] R, [is vertebrate] Y, [is placental] R, [is mammal] Y, [has canon] R, [has shank] R, [has hock] R, [has hoof] N, [has paw] Y, [has pulp] R, [has root] R, [has stump] R, [has socket] R, [has matrix] R, [has corpus] R, [has marrow] R, [has crown] R, [it a canine] N, [is cat] Y, [is wildcat] Y, [is leopard] N, [is puma] Y.
I guess it is **puma**. Y.
22 pytania, z czego 14 „nie wiem" skutkowało usunięciem cech z bazy wiedzy.

Cobra: [is vertebrate] Y, [is mammal] N, [has plumage] R, [is bird] N, [has flipper] N, [is reptile] Y, [is serpent] R, [is snake] Y, [is viper] N, [is boa] N,

[is racer] Y, [is cobra] Y, [is asp] N.
I guess it is **cobra**. Y.
12 pytań prowadzących do poprawnego odnalezienia obiektu.

Butterfly: [is vertebrate] N, [has cell] Y, [is invertebrate] Y, [is insect] Y, [is butterfly] Y, [is copper] N, [is emperor] N, [is admiral] Y/2, [is monarch] N, [is viceroy] N, [is comma] N, [is large white] N.
I guess it is **butterfly**. Y.
11 pytań prowadzących do poprawnego odnalezienia obiektu.

Powyższe przykłady pokazują gry zakończone poprawnym wyszukaniem obiektu. Nie wszystkie jednak wyszukania zakończone zostały sukcesem, co wynika z małej ilości wiedzy w systemie WordNet, która może być użyta w procesie wyszukiwania obiektu. Efektem czego jest brak możliwości oddzielenia od siebie obiektów w podprzestrzeni O(A) w końcowych etapach gry. Drugą przyczyną porażek systemu jest odmienna organizacja wiedzy w źródłowym słowniku leksykalnym od tej, jaką posiada zwykły człowiek. Braki te w pamięci semantycznej poprawiane są przez mechanizm korekty bazy wiedzy oparty na aktywnych dialogach (rozdział 5). Przykład takiej sytuacji obrazuje przedstawiona poniżej gra:

Lion: [is vertebrate] Y, [is mammal] Y, [has hoof] N, [has paw] Y, [is canine] N, [is cat] Y, [is wildcat] Y,

Organizacja pojęcia *lion* w taksonomii WordNet nie uwzględnia go jako *wildcat*, co powoduje, że algorytm wyszukiwania idzie w złą stronę, wynikiem czego jest niepoprawny wynik gry:

[is leopard] N, [is panther] N, [is puma] N.
I guess it is **lynx**. N.
Well, I failed guess your concept. What it was?
lion.
10 pytań – niepoprawny rezultat wyszukania.

Po otrzymaniu poprawnej odpowiedzi odnośnie obiektu wyszukiwania system pamięci semantycznej reorganizuje swoją wiedzę. Polega ona na zmianie typu powiązania dla CRK lion – wildcat, w stronę wartości dodatniej określającej, że takie powiązanie zachodzi. Dodanie nowej wiedzy skutkuje również zamianą rankingu zysków informacyjnych dla cech, dlatego kolejna gra wygląda trochę inaczej, pomimo tych samych odpowiedzi udzielanych na

te same pytania. Zauważyć tu można zapytanie o nie występującą wcześniej cechę: *has mane*, co pozwala zidentyfikować poprawnie obiekt *lion*.

Lion: [is vertebrate] Y, [is mammal] Y, [has hoof] N, [has paw] Y, [is canine] N, [is cat] Y, [is wildcat] Y, [is leopard] N, [has mane] Y.
I guess it is **lion**. Y.
9 pytań prowadzących do poprawnego wyszukania, po korekcie bazy wiedzy.

Drugi przykład błędnego wyszukiwania obiektu, wynikający z odmiennych przyczyn, pokazany został dla wyszukiwania pojęcia z innej niż „ssaki" gałęzi taksonomii.

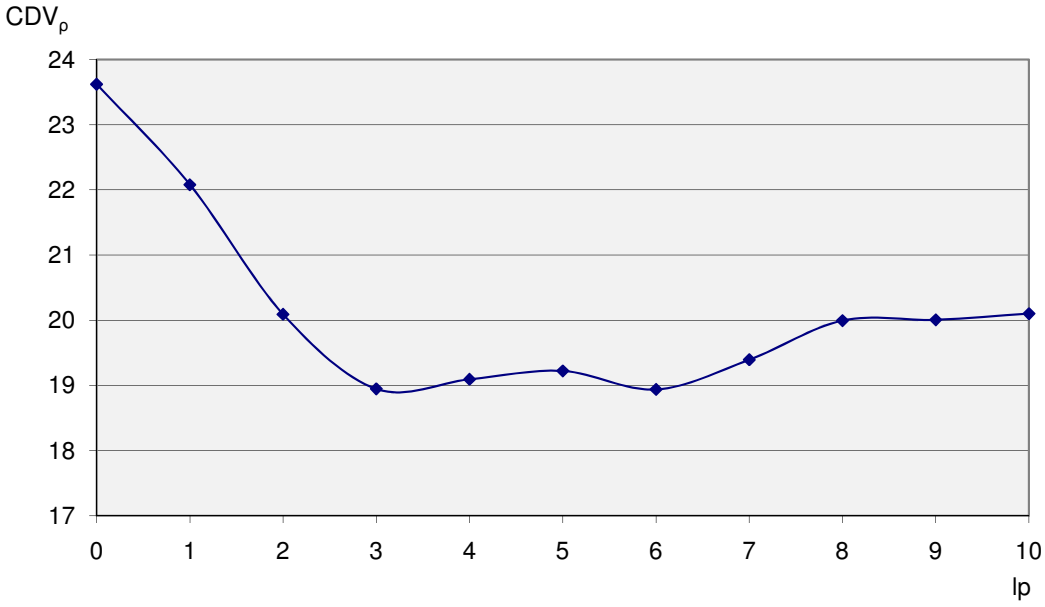
Spider: [is vertebrate] N, [has wing] N, [has tail] N, [is invertebrate] Y, [has shield] N,

W bazie wiedzy systemu WordNet zapisana jest informacja *spider–is_a–arthropod* oraz *arthropod–has–shield*. Użycie dziedziczenia cech prowadzi do błędnego wyszukania obiektu, ponieważ przeciętny użytkownik języka na pytanie: „czy pająk ma pancerz", z pewnością odpowiedziałby „nie". W podobny sposób pojęcie *arthropod* otrzymuje inne, błędne z punktu widzenia zdroworozsądkowej wiedzy powiązania np. *has–ear*, będące częścią głowy, którą *arthropod* posiada. Rozbieżność ta, w dalszej części wyszukania prowadzi do niepoprawnego wyszukania obiektu:

[is worm] N, [is anemone] N, [is coral] N, [is polyp] N.
I guess it is **medusa**. N.
9 pytań prowadzących do niepoprawnego odgadnięcia obiektu.
Well I failed guess your concept. What it was?
spider.

Przeprowadzona z systemem gra w grę 20 pytań (zakończona porażką) i uzyskana właściwa odpowiedź pozwala na pozyskanie nowych informacji do bazy wiedzy, dzięki czemu kolejne wyszukanie obiektu *spider* kończy się sukcesem po 8 pytaniach.

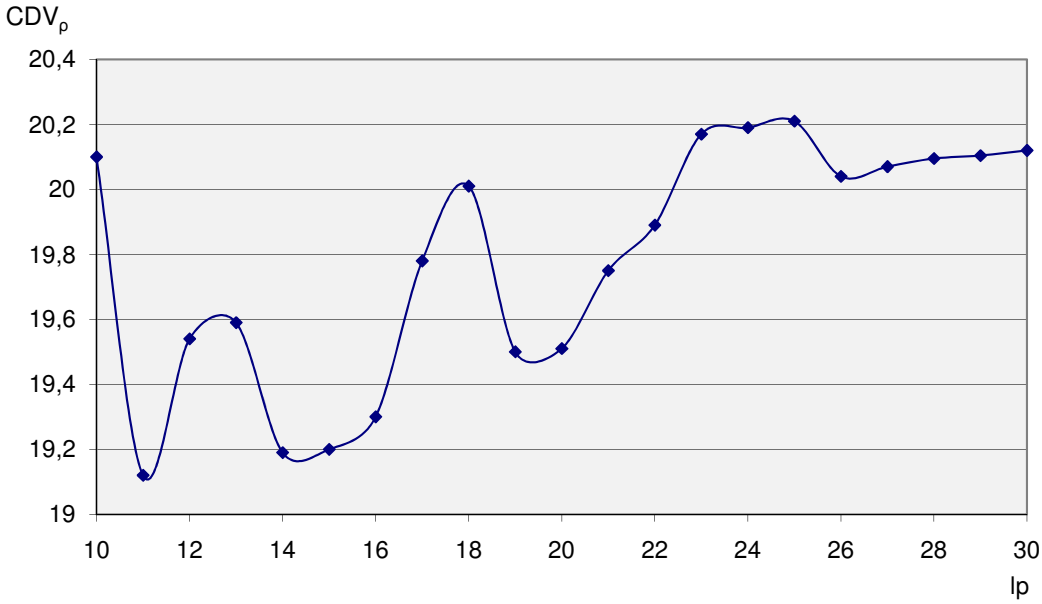
Pierwsze przeprowadzenie procedury oceny jakości przestrzeni semantycznej dało miarę błędu $E = 0,4$. W związku z usuwaniem mało znanych cech, ich średnia liczba w wektorach CDV (CDV_ρ), w kolejnych krokach maleje. Dynamikę tej zmiany w kolejnych krokach procedury testowej przedstawia wykres na rysunku 4.3. Widoczne na nim obszary spadków CDV_ρ wynikają z usuwania cech, na które użytkownik testujący pamięć odpowiedział „nie wiem". Wzrosty CDV_ρ wynikają z aktualizacji bazy wiedzy po grach zakoń-



Rysunek 4.3: Zmiana średniej liczby cech, w wektorach CDV pozyskanych z bazy WordNet, w kolejnych krokach procedury testowej

czonych sukcesem lub wskazaniach właściwego obiektu po grach zakończonych porażką. Widoczne są one przede wszystkim w miejscach, w których nie było usuwania mało istotnych cech. Procedura oceny szacująca jakość pamięci semantycznej, dla świeżo pozyskanych danych ze słownika WordNet przez grę w dwadzieścia pytań jest restrykcyjna. Wygrana dla większej liczby obiektów świadczyłaby o bardzo dobrej jakości danych językowych zapisanych w słowniku WordNet. Otrzymany wynik $E = 0,4$ wskazuje, że dla najpopularniejszych zwierząt, niewiele ponad 50% gier zakończonych zostało sukcesem. Wartość ta, wskazuje na ograniczenie jakościowe pamięci semantycznej zbudowanej ze słownika maszynowego, która podczas pierwszych gier musi zostać przez interakcję z człowiekiem oczyszczona z mało istotnych cech. W celu dokładniejszego oszacowania jakości i bardziej precyzyjnej oceny procesu nabywania wiedzy przeprowadzone zostało dalsze, dodatkowe dwukrotne powtórzenie procedury testowej.

Wykres na rysunku 4.4 obrazuje zmianę średniej liczby cech w CDV, w dalszym procesie testowania pamięci semantycznej. Łączny błąd po wykonaniu 20 testowych wyszukań zmniejszony został do $E = 0,3$. Wskazuje to na poprawienie się jakości pamięci, pomimo zmniejszania się CDV_ρ, co wynika z usunięcia mało popularnych cech. Jest to zgodne z intuicją, że jakość wyszukań niekoniecznie musi zależeć tylko od ilościowego opisu pojęć w pamięci semantycznej, ale również od ich jakości odniesionej do wiedzy posiadanej



Rysunek 4.4: Zmiana średniej liczby cech, w wektorach CDV pozyskanych z bazy WordNet, w kolejnych krokach dwóch procedur testowych

przez ludzi. Oczekiwać można, że przy dalszym powtarzaniu procedur testowych liczba usunięć będzie znacząco maleć – wskazuje na to malejący trend sumarycznej liczby usunięć w kolejnych grach: 5 w pierwszych 10 grach, 4 w kolejnych 10 i tylko 2 w następnych 10. Usunięte w początkowym etapie testów cechy są jednymi z najczęściej występujących atrybutów w wektorach CDV – wynika stąd ich bardzo duży wpływ na średnią liczbę cech w CDV. Usunięcie ich powoduje używanie bardziej deskryptywnych i popularnych cech do wyszukiwania i ustabilizowanie CDV_ρ, który w miarę przeprowadzania dalszych gier z systemem będzie rosnąć. Przy dalszych grach usunięcia mało istotnych cech będą odbywać się na coraz niższych (bardziej szczegółowych) poziomach taksonomii. Wiązać się będą one z mniejszą liczbą obiektów, a przez to wpływ na CDV_ρ będzie mniejszy, co pozwoli wiedzy uzyskiwanej podczas gier poprawiać tą wartość.

Powiązania użyte z bazy WordNet do utworzenia wektorów CDV, wykorzystujące typy relacji *is_a* i *has_a* dają potencjalną możliwość utworzenia przestrzeni semantycznej, która może być przeszukiwana kontekstowo [31]. Zaznaczyć jednak należy, że wiedza oparta tylko na dwóch grupach typów relacji jest stosunkowo uboga. Pomija ona szereg aspektów występujących w języku naturalnym. W zastosowaniu słownika WordNet do gry w 20 pytań uwidacznia się niedoskonałość bazy lingwistycznej ogólnego zastosowania. Objawia się ona brakiem informacji, które są oczywiste dla ludzi, jak również

istnieniem wielu specyficznych pojęć, które nie są znane powszechnie. Dużym brakiem (nad którym trwają obecnie prace w Princeton) jest brak powiązań rzeczowników z czasownikami.

Pomimo, że WordNet stanowi cenne źródło powiązań między pojęciami językowymi, nie może być jednak użyty, jako jedyna baza, zdroworozsądkowej wiedzy dla pamięci semantycznej. W związku z tym, w przeprowadzonych eksperymentach z aktywnymi dialogami (paragraf 5.2) do utworzenia pamięci semantycznej użyte zostały jeszcze inne, dodatkowe zasoby ustrukturalizowanej wiedzy pozwalające na zwiększenie liczby cech w wektorach CDV.

4.2 Strukturalizowanie otwartego tekstu

Jednym z najbardziej atrakcyjnych (w sensie ilości wiedzy, która potencjalnie może zostać pozyskana) sposobów akwizycji danych semantycznych na masową skalę jest automatyczne strukturalizowanie otwartego tekstu. Zadanie to jest wyzwaniem dla algorytmów drążenia danych, a ogólne rozwiązanie ekstrakcji wiedzy z tekstu jest trudne i do tej pory nie istnieje jedno uniwersalne podejście [42].

Umiejętność automatycznego pozyskiwania z otwartego tekstu typowanych powiązań między pojęciami umożliwiłaby realizację sieci semantycznych na szeroką skalę, co znacznie zwiększyłoby obszar ich zastosowań oraz przyspieszyłoby rozwój systemów przetwarzających zasoby językowe. Z metod automatycznej ekstrakcji powiązań semantycznych z dokumentów wyróżnić należy dwa podejścia wynikające z różnego rodzaju semantyki pojawiającej się w tekście: semantyki epizodycznej i semantyki strukturalnej.

Koncepcja pierwsza przyjmuje założenie, że epizodyczne wystąpienia powiązań między słowami w tekście niosą informację o strukturze języka. Motywacją do przeprowadzenia eksperymentów z podejściem opartym na statystycznych własnościach języka jest intuicja, że powtarzające się współwystąpienia słów w tekście odzwierciedlają powiązania występujące w pamięci semantycznej człowieka. Prawdopodobnie na podobnej zasadzie w procesie poznawczym człowieka odbywa się uczenie: z epizodycznych percepcji powstają w mózgu generalizacje, które zostają dalej utrwalone w strukturach pamięci semantycznej.

Podejścia do analizy semantyki w tym paradygmacie opierają się głównie na analizie częstości współwystępowania słów w dokumencie. Najbardziej typowe podejście tworzy z korpusu tekstowego macierz sąsiedztwa słów: każde słowo reprezentowane jest wektorem wartości określających jego współwystępowanie z innymi słowami w tekście. Tekst reprezentowany jest przez macierz kwadratową o rozmiarze $N \times N$, gdzie N jest liczbą niepowtarzających się

słów w dokumencie. W miejscach przecięcia się kolumn i wierszy znajdują się wartości oznaczające stopień powiązania dwóch słów, tak że element $w_{i,j}$ oznacza wartość powiązania słowa i ze słowem j. Typowo stopień powiązania w wyliczany jest jako ważona wartość częstości kookurencji dwóch słów i uśrednionej odległości w tekście między nimi (4.3), choć używane są różne jego modyfikacje.

$$w_{i,j} = \frac{\sum_{k=1}^{N} d_k k}{\sum_{k=1}^{N} d_k} \tag{4.3}$$

gdzie:

d_k jest kolejną wartością odległości (liczbą słów pośrednich) między parą słów i, j.

Otrzymane w ten sposób wektory kookurencji słów używane są do wyznaczenia podobieństwa między słowami. Przestrzeń semantyczna HAL (*Hyperspace analog to language*) [50] zbudowana ze wszystkich słów występujących w analizowanym korpusie tekstu jest strukturą o bardzo dużych rozmiarach. W celu jej zmniejszenia stosuje się algorytmy redukcji wymiarowości np.: SVD, PCA [97] przedstawiające przestrzeń bazową jako liniową kombinację cech pierwotnych. Przyjmując określoną stratę informacji można znacznie ograniczyć rozmiar analizowanej macierzy, bez znaczącego ujemnego wpływu na jakość przetworzenia danych, a nawet ją poprawiając. Jest to podejście powszechnie stosowane do przetwarzania języka naturalnego (NLP) [23], które na celu ma eliminację szumu wynikającego z przypadkowych wystąpień słów. Pozwala ono na podstawie ciągów znaków tworzących tekst określić np. podobieństwo dokumentów, jak ma to miejsce w LSI [32].

Rezultaty uzyskane przy użyciu tego podejścia okazały się zbyt zaszumione, aby mogły zostać użyte jako baza wiedzy dla gry w 20 pytań, dlatego nie są tu opisywane szerzej. Podejście to wspomniane zostało przede wszystkim ze względu, że jest ono utartym paradygmatem przetwarzania tekstu, opartym na podejściu BOW (*bag of words*).

Uzyskane przez analizę częstości współwystępowania słów w tekście kontekstowe powiązania nie pozwoliły na stworzenie dobrej jakościowo przestrzeni semantycznej, weryfikowanej przez zastosowanie w grze słownej. Niepowodzenie podejścia opartego na takiej analizie korpusów tekstowych do aplikacji w grze dwudziestu pytań pokazuje, że automatyczne utworzenie przestrzeni semantycznej jest wyzwaniem dla algorytmów wydobywania informacji z tekstu, a samo wyszukiwanie kontekstowe jest nie tylko demonstracją przetwarzania symboli językowych posiadających znaczenie, ale również wymagającym kryterium pozwalającym ilościowo określić jakość przetworzenia danych tekstowych.

Drugim podejściem do analizy tekstu jest analiza powiązań między słowami poprzez dopasowanie szablonów gramatycznych do fraz występujących w dokumencie. Podejście takie dało zdecydowanie lepsze rezultaty od wspomnianego tu podejścia HAL. Drążenie tekstu (*Text Mining*) oparte na gramatycznej analizie pozwoliło również pozyskać wiedzę o relacjach negatywnych (typu „nie stosuje się") między obiektem, a jego cechą. Informacja taka nie jest dostępna powszechnie w ustrukturalizowanych słownikach maszynowych (*Machine-readable dictionaries*). Zastosowane podejście i wyniki opisane zostały w kolejnym paragrafie.

4.2.1 Aktywne szukanie

Pozyskiwanie ustrukturalizowanej wiedzy z tekstu, automatyczna ekstrakcja typowanych powiązań z dokumentów jest zadaniem trudnym [37] – obrazują to mało zadowalające rezultaty podejścia HAL oparte na statystycznej analizie tekstu, które nie nadawały się do zastosowania w grze w dwadzieścia pytań.

Strukturalizowanie informacji zawartej w dokumencie wymaga posiadania pewnej wiedzy o budowie języka, która zostanie użyta do analizy tekstu. Dzięki takiej wiedzy możliwe jest częściowe ustrukturalizowanie informacji zapisanej w postaci języka naturalnego. Wiedzą taką może być na przykład znajomość składni lub typowo używanych konstrukcji językowych. Wiedza o strukturze języka pozwala na wyodrębnienie elementarnych jednostek leksykalnych, z których tworzone są powiązania posiadające określone znaczenie. Zastosowanie tej wiedzy do pozyskiwania danych dla pamięci semantycznej umożliwi automatyczne rozszerzenie opisów obiektów w postaci wektorów CDV.

Automatyzacja procesu pozyskiwania wiedzy semantycznej z tekstu jest konieczna do utworzenia pamięci semantycznej na dużą skalę. Wynika to z ograniczonej liczby zasobów ludzkich, które mogą zostać użyte do ręcznego przetwarzania dokumentów.

Zaprezentowane w tym paragrafie podejście do akwizycji wiedzy z definicji słownikowych użyte zostało do pozyskiwania danych do pamięci semantycznej. Metoda tu zaprezentowana użyta również została w module analizy tekstu systemu, gdzie informacje wprowadzane przez użytkownika po zakończeniu gry, (w postaci zdania w języku naturalnym) zostają zamienione na wiedzę ustrukturalizowaną w postaci wCRK[8]. Dzięki pozyskanym w ten sposób informacjom do pamięci semantycznej trafiają nowe cechy, które służą do

[8]wCRK - jest atomem wiedzy opisanym w rozdziale 3.1, nie zawierającym wagi v określającej jej pewność

opisu obiektów, dla których poprzez kolejne gry powstają nowe powiązania. Proces ten opisany jest szerzej w paragrafie 5.1.

Aktywne szukanie jest propozycją ogólnej metody ekstrakcji wiedzy w postaci trójek wCRK z zasobów słownikowych. Algorytm aktywnego szukania opiera się na analizie składniowej tekstu, na bazie której utworzona jest funkcja dokonująca oceny prawdziwości zadanej trójki w postaci: obiekt – predykat – cecha. Zastosowanie i uzyskane wyniki algorytmu aktywnego szukania przedstawione zostały na przykładzie pozyskiwania z tekstu relacji typu „posiada", wiążących obiekt z cechą. Zaproponowane podejście jest na tyle ogólne, że umożliwia ekstrakcję innych typów powiązań (wskazywanych określonymi predykatami) np. charakteryzujących powiązania taksonomiczne.

Eksperymenty z automatycznym pozyskiwaniem wiedzy z tekstu użyte zostały w celu wzbogacenia opisów obiektów w pamięci semantycznej. Polegały one na akwizycji wiedzy z korpusu tekstowego, utworzonego z definicji słownikowych systemu WordNet dla wybranych przez wskazanie kategorii semantycznej pojęć. Otrzymane wyniki automatycznego przetworzenia takiego korpusu tekstu porównano z już istniejącymi powiązaniami w bazie WordNet.

4.2.2 Algorytm aktywnego szukania

Ekstrakcja cech z definicji słownikowych zrealizowana została przez algorytm aktywnego szukania. Działanie algorytmu opiera się na założeniu, że dostarczony tekst, na podstawie którego dokonana zostaje ocena jest opisem dla określonego pojęcia. Pozwala to uniknąć poszukiwania w treści tekstu podmiotu mającego określić, czego on dotyczy – zakładamy, że wszystko w dostarczonej definicji dotyczy zadanego pojęcia. Zaletą użycia definicji słownikowych jest również fakt, że w takich opisach znajdują się cechy deskryptywne obiektów, będące charakterystyczne dla analizowanego pojęcia. Analiza definicji słownikowej badającej gramatyczne i morfologiczne otoczenie poszukiwanej cechy pozwala na ustalenie, na jakiej zasadzie zachodzi powiązanie między obiektem a cechą.

Algorytm aktywnego szukania oparty jest na regułowej funkcji kryterialnej określającej, czy i jak poszukiwana cecha stosuje się do badanego obiektu (danego definicją określającą go pojęcia). Funkcja oceniająca na podstawie dostarczonej definicji określa, czy dany obiekt posiada, bądź nie posiada powiązania określonym predykatem ze wskazaną cechą, jak również ma możliwość wskazania sytuacji, gdy na podstawie zadanej definicji słownikowej nie można dokonać klasyfikacji. Funkcja kryterialna określona została następująco:

$$FC(cecha, predykat, definicja(pojcie)) =$$

$$\begin{cases} > 0 & \text{, jeżeli cecha } \textbf{stosuje} \text{ się do obiektu} \\ < 0 & \text{, jeżeli cecha } \textbf{nie stosuje} \text{ się do obiektu} \\ 0 & \text{, jeżeli nie można rozstrzygnąć} \end{cases} \tag{4.4}$$

Reguły funkcji kryterialnej FC podzielone są na dwie fazy:

Faza I Analiza końcówek cechy:

1. W zadanym tekście, będącym definicją analizowanego obiektu poszukiwana jest zadana cecha.
2. Dla odnalezionej cechy wykonywana jest funkcja ekstrakcji jej rdzenia (stemming[9]), tak by otrzymać formę podstawową poszukiwanej cechy.
3. Dla utworzonej formy podstawowej ustalona zostaje końcówka (występująca w tekście), tworząca odnalezioną cechę.

Jeśli powiodło się dopasowanie końcówki do zbioru zdefiniowanych dla predykatu sufiksów można na tej podstawie dokonać klasyfikacji i w zależności od przypisanej do końcówki decyzji ustalić, czy zadana cecha stosuje się do obiektu, czy nie.

Dla predykatu *have* końcówka *-ed* wskazywać będzie na posiadanie cechy np. *horn**ed***, natomiast końcówka *-less* będzie wskazywać na negatywne powiązanie obiektu z cechą np. *horn**less***.

Jeśli algorytm nie dokona klasyfikacji, następuje przejście do drugiej fazy.

Faza II Analiza otoczenia predykatu

1. Na podstawie zadanego predykatu utworzone zostają dwa zbiory: synonimów i antonimów.

   Dla predykatu *have* zbiór synonimów zawierał słowa: [*with, has, have, posses, having*], natomiast w zbiorze antonimów były słowa: [*have not, without, lack*]. Do oceny częściowej przynależności zdefiniowane zostały analogiczne dwa zbiory określające powiązania typu pośredniego: „czasami" [*usual, frequently, often*] i „rzadko" [*seldom, raely, uniquely*].

[9] http://tartarus.org/∼martin/PorterStemmer ,VII 2007

2. W otoczeniu znalezionej w definicji cechy wyszukany zostaje najbliższy (w sensie odległości miedzy wyrazami) predykat z powyższych zbiorów.
3. Jeśli jakikolwiek predykat zostanie odnaleziony i żaden inny czasownik nie wystąpił bliżej szukanego słowa (cechy), następuje klasyfikacja. Dla synonimów *have* określone zostaje, że cecha się stosuje, a jeśli wystąpił antonim – cecha się nie stosuje. Analogiczna klasyfikacja zachodzi dla powiązań pośrednich.

Opisana regułowa funkcja kryterialna pozwala na podstawie określonego zasobu tekstowego (definicji) dokonać oceny trójki: obiekt – predykat – cecha i przypisać wagę określającą kierunek powiązania. Ustalenie reguł funkcji oceniającej polega na określeniu poszczególnych zbiorów, na bazie których podejmowana jest decyzja: w pierwszej fazie algorytmu, końcówek charakteryzujących powiązania dodatnie, bądź ujemne, a dla drugiej fazy, zbiorów synonimów oraz antonimów wskazujących rodzaj powiązania. Metoda taka pozwala na elastyczne budowanie nowych funkcji ocen dla innych predykatów, dzięki którym możliwa jest szersza analiza otwartego tekstu.

4.2.3 Przykłady aplikacji i wyniki eksperymentów

Algorytm aktywnego szukania zastosowany został do przetworzenia definicji słownika WordNet. Uzyskane rezultaty ekstrakcji cech ocenione zostały w aplikacji do gry w 20 pytań oraz przez porównanie do już istniejących w bazie WordNet ustrukturalizowanych powiązań między pojęciami.

Do testów wybrano pojęcia wyznaczone semantyczną kategorią *noun. animal*, natomiast cechy wyznaczył zbiór pojęć z semantycznej kategorii *noun.body part*. Predykatem wiążącym obiekt z cechą był czasownik *have*, dla którego ręcznie ustalone zostały zbiory pozytywnych i negatywnych sufiksów (dokonujących klasyfikacji powiązania cechy: tak/nie) oraz na podstawie danych zawartych w bazie WordNet utworzone zostały dla predykatu *have* zbiory synonimów i antonimów. Użycie ich umożliwia dokonanie klasyfikacji typu: tak, nie, czasami, rzadko. Klasyfikacja powiązań dokonana została na podstawie definicji synsetów z bazy WordNet, dla pojęć z kategorii semantycznej *noun.animal*.

Aktywne szukanie wykonywane jest kolejno dla wszystkich cech – słów z semantycznej kategorii *noun.body part*. W pierwszym kroku zbiór analizowanych definicji ograniczony zostaje tylko do tych, w których występuje ciąg liter będących zadaną cechą. Ograniczenie polegające na dopasowaniu ciągu znaków może dawać czasami niepoprawne rezultaty np. dla cechy *horn*, do

analizy użyte zostają również definicje zawierające cechę *thorn*. Nie wpływa to jednak negatywnie na klasyfikację, ponieważ zostaną one odrzucone podczas porównania rdzenia odnalezionego słowa i zadanej cechy.

Poniżej przedstawiono przykłady oceny prawdziwości asercji CRK, jakie dała funkcja kryterialna dokonująca klasyfikacji na podstawie dostarczonej definicji w języku naturalnym:

1. definicja słownikowa: [catamount, lynx] – *short-**tailed** wildcats with usually tufted **ears**; valued for their fur*

   Asercja FC(ryś, posiada, ogon)

   > rezultat funkcji kryterialnej = 1, uzyskany na podstawie analizy końcówek (I fazy FC)

   Dla oceny asercji FC(ryś, posiada, ucho)

   > rezultat funkcji kryterialnej = 1 , uzyskany na podstawie analizy otoczenia predykatów (II fazy FC)

2. definicja słownikowa: [anuran, batrachian, frog, salientian, toad, toad frog] – *any of various **tailless** stout-bodied amphibians **with** long hind **limbs** for leaping; semiaquatic and terrestrial species*

   Asercja FC(żaba, posiada, ogon)

   > rezultat funkcji kryterialnej = -1, uzyskany na podstawie analizy końcówek

   Asercja FC(żaba, posiada, noga)

   > rezultat funkcji kryterialnej = 1, uzyskany na podstawie analizy synonimów predykatu

3. definicja słownikowa: [mouse] *any of numerous small rodents typically resembling diminutive rats having pointed snouts and small ears on elongated bodies with slender usually **hair**less tails*

   Asercja FC(mysz, posiada, futro)

   > rezultat funkcji kryterialnej = -1, uzyskany na podstawie analizy końcówek. Błędna klasyfikacja wynika z założenia przyjętego, że wszystko w definicji dotyczy rozpatrywanego obiektu. Przykład ten obrazuje, jak bardzo różnorodne są możliwości wyrażenia w języku naturalnym prostych informacji, co stawia wyzwanie inteligentnym metodom pozyskiwania wiedzy z otwartego tekstu.

Przetworzone algorytmem aktywnego szukania opisy słownika WordNet dla pojęć z kategorii semantycznej *noun.animal* pozwoliły uzyskać 5773 nowe powiązania typu „posiada" wiążące 2103 obiekty z 267 cechami. Spośród otrzymanych powiązań 104 były typu negatywnego, czyli „nie stosuje się". Ocena otrzymanych wyników przeprowadzona została w dwojaki sposób:

1. W odniesieniu do pierwotnie utworzonej przestrzeni semantycznej Ψ_{wn} z bazy WordNet (opisanej w paragrafie 4.1).

   Do porównania użyta została przestrzeń semantyczna powstała z mapowania znaczeń na słowa, bez ograniczenia jej rozmiarów. Porównanie dwóch przestrzeni semantycznych Ψ dla wybranej relacji (tu: *has*, opisanej w bazie WordNet relacjami meronimicznymi) pozwoliło określić, jaki procent cech i powiązań pozyskanych przez aktywne szukanie pokrywa się z danymi już zdefiniowanymi w słowniku leksykalnym.

2. Przy użyciu algorytmu wyszukiwania kontekstowego zastosowanego do gry w 20 pytań.

   W podejściu tym, wynik przeprowadzonej gry słownej na utworzonej poprzez aktywne szukanie przestrzeni semantycznej Ψ_{as} pozwala na ocenę jakości danych uzyskanych z użyciem tej metodologii.

Poprzez zastosowanie aktywnego szukania w definicjach słownika WordNet odnalezionych zostało 65 cech, posiadających zdefiniowane powiązania w bazie WordNet. Dla tych cech algorytm aktywnego szukania odnalazł 2736 powiązań dla pojęć z kategorii semantycznej *noun.animal*. Poprzez akwizycję wiedzy z tekstu odtworzono 24% cech zapisanych w bazie WordNet, przy czym powiązania, które tworzą te cechy stanowią 47,3% całej wiedzy pozyskanej w procesie aktywnego szukania. 61,8% z tych powiązań udało się potwierdzić poprzez istnienie analogicznych powiązań w bazie WordNet. Pozostała część uzyskanych powiązań jest nowa, bądź błędna (co nie mogło zostać ocenione ze względu na brak ich w bazie WordNet). Wystąpienie nieprawidłowych powiązań wynika z przyjętego założenia, że wszystko w dostarczonej do funkcji kryterialnej definicji pojęcia odnosi się do opisywanego obiektu. Możliwość pozyskania przez aktywne szukanie nieistniejących powiązań w referencjalnej przestrzeni semantycznej wynika z niekompletności leksykalnej bazy WordNet.

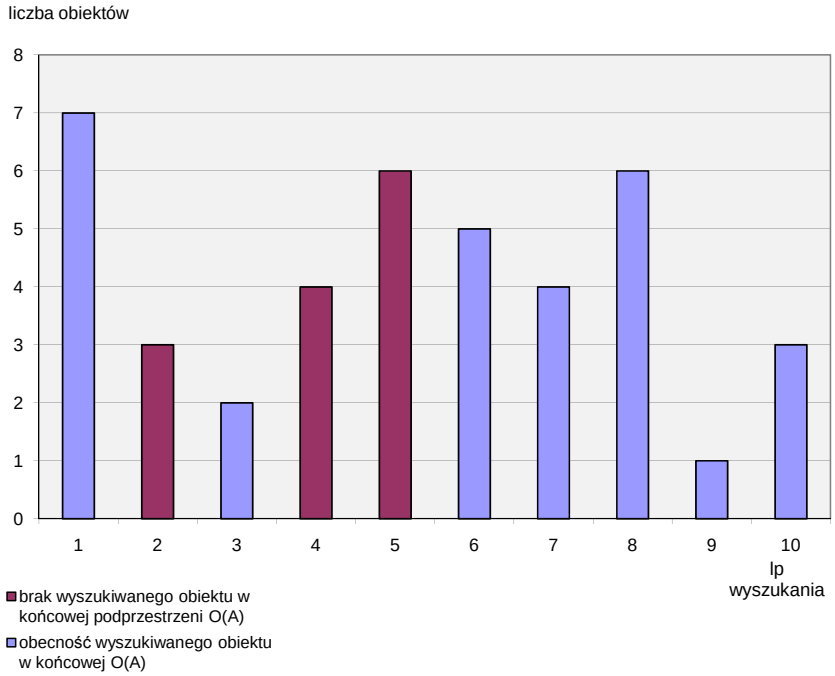
Porównując powiązania otrzymane w procesie aktywnego szukania z relacjami zapisanymi w systemie WordNet zauważyć można, że WordNet zawiera wiedzę bardziej o charakterze strukturalnym, która związana jest z pojęciami ogólnymi (wyżej w taksonomii). W wypadku analizy opisów słownikowych pozyskana wiedza związana jest bardziej z konkretnymi obiektami i ma charakter definicyjny (bardziej szczegółowy).

Uzyskane powiązania – relacje typu *has_a* występujące między określonymi pojęciami (reprezentującymi obiekty pamięci semantycznej) a cechami – słowami z semantycznej kategorii *noun.body part* pozyskane z definicji słownika WordNet użyte zostały jako dane dla pamięci semantycznej. Same relacje typu *has_a* nie umożliwiają zastosowania żadnego wnioskowania (proceduralnej interpretacji typów relacji), które pozwoliłoby uzyskać dodatkową wiedzę (której używa referencjalna przestrzeń Ψ_{wn} utworzona z bazy WordNet opisana w rozdziale 4.1). W związku z tym, w przeprowadzonych testach oceniających jakość danych pozyskanych z tekstu, poprzez grę w 20 pytań użyta została dodatkowo taksonomia, utworzona z hypernimów bazy WordNet. Pozwala ona dziedziczyć relacje typu *has* uzyskane przez aktywne szukanie. W wypadku oceny jakości przy użyciu gry, wzbogacenie przestrzeni semantycznej o powiązania taksonomiczne pozwoliło z jednej strony poprawić jakość gry, z drugiej zachować sprawiedliwe odniesienie (bo oceniane są tylko relacje typu „posiada"), którym była przestrzeń semantyczna wygenerowana z meronimow zapisanych bazie WordNet, a do dziedziczenia których użyta została taksonomia tego słownika.

Wyniki zastosowania oceny przez wyszukiwanie w przestrzeni cech uzyskanej przez aktywne szukanie dały wartość jakości Q = 0,1. Oznacza to, że jedynie jedna gra na dziesięć testowych zakończyła się sukcesem. Przyczynę tak niskiego wyniku upatrywać można w występowaniu błędnych powiązań w wektorach CDV. Niepoprawne wyszukania wynikają również z braku możliwości rozdzielenia pod koniec procesu wyszukiwania podobnych obiektów do siebie, ponieważ gry odbywają się jedynie z użyciem cech uzyskanych w aktywnym szukaniu, których jest niewiele. Na rysunku 4.5 wysokościami słupków przedstawiono liczby nierozróżnialnych obiektów na zakończenie gry w kolejnych 10 testowych wyszukaniach.

Wykres obrazuje, ile obiektów pozostało w podprzestrzeni najbardziej prawdopodobnych obiektów O(A) po wyczerpaniu się cech używanych do zadania zapytania. W związku z tym, że sytuacja taka kończy proces wyszukiwania, obiekty gry nie mogły zostać odnalezione. Niebieskim kolorem oznaczone zostały końcowe podprzestrzenie O(A), w których był poszukiwany obiekt, czerwonym, w których poszukiwanego obiektu nie było. Przyjmując jako przyczynę porażki gry zbyt małą liczbę cech, (która może zostać skompensowana dodaniem nowych danych, z innych źródeł wiedzy) ocenić można błąd pamięci dla testowych wyszukiwań, które mogłyby zakończyć się powodzeniem na $E = 0,3$.

Przedstawiona metoda aktywnego szukania wskazuje na potencjalną możliwość strukturalizacji wiedzy z zasobów tekstowych. Wydajność jej w znacznej mierze zależeć będzie od liczby utworzonych funkcji kryterialnych, określających różne możliwe powiązania między słowami, jakie mogą zachodzić



Rysunek 4.5: Liczby nierozróżnionych obiektów w końcowej podprzestrzeni O(A), w kolejnych testowych grach

w języku. Dla testowej funkcji kryterialnej, do pozyskiwania relacji typu *has*, liczba powiązań uzyskana poprzez aktywne szukanie wskazuje, że metoda ta może zostać użyta jedynie jako uzupełnienie wiedzy w pamięci semantycznej, a nie do konstrukcji podstawowej bazy wiedzy.

Możliwość rozstrzygania prawdziwości asercji w postaci trójek CRK na podstawie tekstu w języku naturalnym jest wyzwaniem dla inteligencji obliczeniowej. Jakość uzyskanych wyników zależy od zastosowanych algorytmów wydobywania tej wiedzy (reguł użytych do konstrukcji funkcji kryterialnej), jak również od zasobów tekstowych na bazie, których taka klasyfikacja będzie zachodzić.

Dużą trudnością jest również masowa ocena uzyskanych wyników. W przeprowadzonych eksperymentach uzyskane wyniki odniesione zostały do słownika WordNet. Ze względu na niepełność informacji płynącej zarówno z aktywnego szukania (będącą funkcją wiedzy zawartej w definicjach słownikowych), jak również niekompletność wiedzy w bazie WordNet, ocena jakości otrzymanej wiedzy możliwa jest jedynie dla obszaru wyznaczonego częścią wspólną cech występujących w bazie leksykalnej i pozyskanych z definicji.

Zaletą akwizycji wiedzy z tekstu przy użyciu aktywnego szukania jest możliwość pozyskania powiązań innych niż tylko relacje pozytywne (relacje

typu nie stosuje się). Ontologie powszechnie budowane są jedynie z powiązań typu „pozytywnego". Do opisu obiektów w języku naturalnym używana jest również i wiedza negatywna (co obrazuje np. występowanie w definicjach słownikowych relacji typu „nie posiada"). Wiedza taka jest bardzo przydatna z punktu widzenia wyszukiwania kontekstowego, jak również istotna jest z punktu widzenia kompletności opisu obiektu w wektorze CDV. Brak tego rodzaju powiązań w ustrukturalizowanych słownikach jest ich dużym mankamentem. Problem ten do pewnego stopnia może być rozwiązany przez aktywne szukanie, którego użycie umożliwia pozyskiwanie powiązań negatywnych.

Drugą zaletą zastosowania aktywnego szukania jest możliwość pozyskiwania cech definicyjnych, które typowo są używane w opisach słownikowych. Związane są one z konkretnym obiektem i bardzo korzystnie wpływać będą na jakość wyszukiwania, zwłaszcza w końcowych etapach gry, ponieważ stanowią one wiedzę o konkretnym obiekcie (bezpośrednio z nią związaną), a nie wynikającą ze struktury wiedzy np. taksonomii. Użycie taksonomii w słownikach maszynowych przeważnie nie uwzględnia wyjątków na bardziej szczegółowych poziomach. Efektem tego jest pojawianie się nieprawdziwej wiedzy np. w słowniku WordNet z pojęciem *vertebrate* związana jest z cechą *has tail*, co poprzez dziedziczenie *mammal–is_a–vertebrate* powoduje błędną wiedzę *human–has–tail*. Aktywne szukanie może tego typu wyjątki wychwycić, o ile są one wymienione w treści definicji.

Pomimo relatywnie małej ilości pozyskanej wiedzy (w stosunku do słownika WordNet), algorytm aktywnego szukania okazał się użyteczny do drążenia zasobów tekstowych – pozyskał nową wiedzę, z użyciem której możliwe było przeprowadzenie wyszukiwania kontekstowego, zakończonego częściowym powodzeniem. Przedstawione podejście obrazuje ogólną metodę, z użyciem której można pozyskiwać różnego rodzaju strukturalizowaną wiedzę z zasobów tekstowych, opartą na budowie odpowiednich funkcji kryterialnych.

W podsumowaniu tego paragrafu należy jeszcze raz wspomnieć, że strukturalizowanie informacji zawartej w tekście jest zadaniem trudnym, ponieważ język naturalny posiada możliwość wyrażenia tego samego znaczenia na bardzo wiele sposobów, co powoduje duże trudności w formalizacji takiego procesu. Główną zaletą aktywnego szukania jest możliwość automatycznego pozyskania wiedzy oczywistej dla ludzi, a której nie ma w słownikach maszynowych. W lingwistyce obliczeniowej najtrudniejsze dla komputerów jest to, co jest całkiem oczywiste dla człowieka, dlatego domena zwierząt może się wydawać trywialna, ale dla maszyn jest trudna. Cechy takie, jak np.: „ma nogi" (lub ich brak), „lata" (lub nie) nie są w słownikach maszynowych ogólnego przeznaczenia dobrze określone, a mogą zostać one pozyskane (przynajmniej częściowo) w automatyczny sposób z opisów słownikowych, bądź artykułów encyklopedycznych.

Rozdział 5

Akwizycja wiedzy semantycznej przez interakcję z człowiekiem

5.1 Pozyskiwanie wiedzy poprzez grę w 20 pytań

Jednym z rodzajów wiedzy, jaką posiada człowiek, jest wiedza zdroworozsądkowa (*common sense knowledge*) [82], będąca takim obszarem wiedzy ludzkiej, który jest wspólny użytkownikom języka. Teorie psycholingwistyczne wskazują pamięć semantyczną, jako miejsce składowania takich informacji w mózgu. Pamięć ta zawiera powszechnie znane powiązania między pojęciami reprezentującymi obiekty a ich cechami. Skojarzenia takie są oczywiste dla ludzi, jednak dla maszyn – nie. By mogły być używane przez komputery, muszą zostać wpierw wprowadzone do bazy wiedzy systemu używającego języka.

Akwizycja wiedzy zdroworozsądkowej jest zadaniem trudnym [81]: wiedzę taką powszechnie posiada każdy człowiek i przeważnie nie używa jej jawnie w akcie komunikacyjnym. Stanowi ona tło znaczeniowe, dzięki któremu podczas komunikacji zachodzi rozumienie. Wyposażenie maszyn cyfrowych w ten rodzaj wiedzy stanowi nieodzowny i jeden z pierwszych kroków na drodze zaimplementowania w nich możliwości rozumienia znaczenia pojęć, co jest niezbędne do posługiwania się językiem naturalnym.

Zdroworozsądkowa wiedza na temat popularnych obiektów należących do kategorii naturalnych może zostać pozyskana na szereg sposobów. Wśród głównych podejść wymienić należy:

1. bezpośrednie (ręczne) wprowadzanie danych do bazy wiedzy systemu
2. pozyskiwanie wiedzy z zasobów cyfrowych:

ustrukturalizowanych słowników maszynowych,

definicji słownikowych,

korpusów tekstu w języku naturalnym,

3. obserwacja działalności człowieka i na jej podstawie rozszerzanie wiedzy systemu. W szczególności może być to interakcja w języku naturalnym.

Obok metody wyszukiwania obiektów poprzez ich cechy algorytm wyszukiwania kontekstowego zastosowany do gry w dwadzieścia pytań może zostać użyty jako metoda akwizycji wiedzy w postaci powiązań między pojęciami (będącymi reprezentacjami obiektów zapisanych w pamięci semantycznej).

Poniżej, przedstawione zostało podejście do gromadzenia wiedzy poprzez aktywne dialogi – szablony interakcji człowieka z pamięcią semantyczną. Pozwalają one nabywać wiedzę w postaci zdroworozsądkowych powiązań między pojęciami języka naturalnego z użyciem gry słownej.

Akwizycja wiedzy do pamięci semantycznej poprzez aktywny dialog wymaga dodania do gry dwudziestu pytań elementu uczenia. Nabywanie nowej wiedzy do pamięci semantycznej odbywa się na podstawie wyniku każdej z przeprowadzonych gier. Aktualizacja taka wprowadza powiązania zarówno dla nowych obiektów, jak i koryguje już istniejące. Reprezentacja umożliwiająca doskonalenie wiedzy opisana została w rozdziale 3.1.

Każda zakończona gra zawiera zbiór pytań i odpowiedzi dla cech użytych do wyszukiwania określonego obiektu. Wiedza pochodząca od użytkownika pozwala na korektę pamięci semantycznej. W zależności od wyniku gry, wektor CDV właściwego obiektu (złożony z atomów wiedzy vwCRK) zostaje zmodyfikowany dla cech użytych w grze, o wartości podpowiedzi użytkownika, zgodnie z formułami 3.1 i 3.3.

Dotychczas przedstawiona interakcja użytkownika z pamięcią semantyczną odbywała się przez odpowiedzi na pytania, które zadaje system podczas gry, a na które człowiek odpowiada: „tak", „nie", „nie wiem", „czasami", „rzadko". W celu jej rozszerzenia wprowadzono szablony dialogów, które są sposobem na zadawanie użytkownikowi dodatkowych pytań, umożliwiających pozyskanie nowej wiedzy. Na pytania te człowiek może udzielać systemowi bardziej złożonych odpowiedzi, w szczególności mogą być to proste zdania w języku naturalnym. W grze 20 pytań zrealizowano następujące szablony interakcji:

- Podczas porażki systemu, gdy obiekt nie został odgadnięty – do pamięci semantycznej pozyskiwana jest nowa wiedza poprzez uzupełnianie wektora CDV obiektu będącego odpowiedzią użytkownika na zapytanie: *„Poddaję się, nie udało mi się zgadnąć. Co miałeś na myśli?*". W sytuacji takiej możliwe są dwa scenariusze:

– Jeśli obiekt był znany wcześniej – pytania z wektora ANSW korygują wektor CDV obiektu wskazanego przez odpowiedź użytkownika.

– Jeśli obiekt nie był znany wcześniej – uzyskane odpowiedzi stanowią bazowy wektor CDV dla nowo pozyskanego obiektu.

Potencjalną możliwością rozszerzenia tego dialogu jest zadanie w tym miejscu pytania pozwalającego na rozszerzenie zbioru cech uzyskanych podczas gry. Uzyskanie odpowiedzi na pytanie: *„Nie znałem wcześniej tego obiektu. Do czego on jest podobny?"* pozwala (ze zmniejszoną wartością pewności wiedzy v) uzyskać dodatkowe cechy z wektora CDV już znanego obiektu. Modyfikacją tego podejścia jest umieszczenie nowo pozyskanego obiektu pomiędzy dwoma najbardziej podobnymi do wektora ANSW wektorami CDV, wyznaczanymi przez dwa najbardziej prawdopodobne obiekty w końcowej podprzestrzeni gry O(A), która zakończona została porażką.

- Po wygranej systemu – właściwe odgadnięcie obiektu, który użytkownik miał na myśli, daje analogiczną do wcześniej przedstawionej, możliwość korekty wektora CDV, która przeprowadzana jest dla poprawnie odgadniętego obiektu. Po grze zakończonej sukcesem, użytkownikowi zadane zostaje dodatkowe pytanie *„Co jest jeszcze charakterystyczne dla odgadniętego obiektu?"*. Odpowiedź użytkownika pozwala wzbogacić przestrzeń semantyczną o nowe, do tej pory niewystępujące w bazie wiedzy cechy. Podejście to wymagało realizacji analizatora języka naturalnego, który potrafi przełożyć proste zdania w postaci języka naturalnego na reprezentację vwCRK. Zrealizowane rozwiązanie oparto na komponentach OpenNLP[1], umożliwiających gramatyczną analizę języka.

Aktywne dialogi dają możliwość korekty i wprowadzania nowej wiedzy do pamięci semantycznej. Przedstawione schematy interakcji z użytkownikiem mogą być rozszerzane o kolejne scenariusze, pozwalające na zdobywanie różnego rodzaju wiedzy. Przykładem jest aktywny dialog pozwalający na wprowadzenie generalizacji cech między podobnymi obiektami (podobieństwo wyznacza tu poziom taksonomii). Uruchamiany jest on poza grą, podczas procesu konsolidacji, którego celem jest uśrednienie wiedzy systemu i wiedzy pochodzącej od użytkowników oraz zachowanie ekonomii kognitywnej. Zamiana wiedzy zapisanej w postaci wektorów CDV (przestrzeni semantycznej Ψ) na

[1] http://opennlp.sourceforge.net/ ,VII 2007

sieć semantyczną ζ pozwala przedstawić dane zapisane w pamięci semantycznej w postaci grafu, w którym liczba redundantnych cech jest minimalizowana przez użycie typu relacji *is_a*. Mechanizm ten polega na identyfikacji redundantnych cech wśród pojęć należących do tej samej podgałęzi taksonomii. Wyróżnioną w ten sposób cechę można przenieść na bardziej ogólny poziom abstrakcji (reprezentowany taksonomią). Często występuje sytuacja, w której cecha występuje dla dużej grupy obiektów podrzędnych (lecz nie dla wszystkich). Przeniesienie takiej cechy na bardziej ogólny poziom taksonomii nie może odbyć się automatycznie, a wymaga uzyskania potwierdzenia od użytkownika, czy taka operacja jest dopuszczalna. Odbywa się to poprzez zadanie użytkownikowi pytania, czy określona cecha jest typowa dla pojęcia nadrzędnego. Proces generalizacji wiedzy przedstawić można następująco:

1. dla każdego obiektu zidentyfikuj cechę, która występuje we wszystkich podgałęziach taksonomii. Przenieś ją do nadrzędnego węzła usuwając z obiektów podrzędnych.

2. zidentyfikuj cechę, która występuje w powiązaniu z największą liczbą obiektów w podgałęzi taksonomii.

3. zadaj pytanie *„Czy «zidentyfikowana cecha» jest cechą obiektu «nadrzędny węzeł»?”*

4. jeśli użytkownik odpowie twierdząco – usunięte zostają powiązania cechy ze wszystkimi pojęciami podrzędnymi i następuje powiązanie jej z pojęciem nadrzędnym.

Użyty szablon aktywnego dialogu pozwala wprowadzać reorganizację wiedzy w bazie, podczas zamiany reprezentacji z postaci wektorów CDV na sieć semantyczną ζ. Umożliwia on usuwanie redundantnych cech i pozwala na zachowanie elementarnej (opartej na typie relacji is_a) ekonomii kognitywnej.

Istnieje szereg innych możliwości implementacji dialogów opartych na reprezentacji w postaci sieci powiązań między pojęciami: jako przykład wskazać tu można wspomniane wcześniej dopytywanie się o podobieństwa, lub też zadawanie pytań o analogie na wysokim poziomie w taksonomii. Przykładem obrazującym potencjalne użycie analogii do akwizycji wiedzy jest sytuacja, gdzie mamy obiekty połączone z cechami predykatem „porusza” np.: ptak – porusza – lata, ryba – porusza – pływa, a brak jest wiedzy ssak – porusza – ?, o którą można zadać pytanie. Możliwość użycia analogii do pozyskiwania wiedzy zdroworozsądkowej w procesie interakcji z ludźmi przedstawiona została szerzej w opisie projektu *Learner* [16].

W przedstawionych dalej wynikach eksperymentów z użyciem interpretacji typów relacji i ich wpływem na ilość wiedzy w systemie (rysunek 5.1) konieczne było ręczne utworzenie powiązań w pamięci semantycznej dla określonych typów relacji. W pamięci semantycznej brak jest możliwości pozyskiwania powiązań pomiędzy cechami realizowanymi przez relacje typu *entail* i *excludes* (opisanych w rozdziale 3.1). Implementacja mechanizmów pozwalających pozyskiwać wiedzę tego typu jest jednym z ważniejszych kierunków rozwoju aktywnych dialogów.

5.2 Wyniki eksperymentów pozyskiwania wiedzy przez interakcję z człowiekiem

W tym rozdziale przedstawione zostały przeprowadzone eksperymenty i uzyskane wyniki użycia algorytmu wyszukiwania kontekstowego do pozyskiwania wiedzy dla pamięci semantycznej. Akwizycja wiedzy przeprowadzona została w oparciu o grę w dwadzieścia pytań rozszerzoną o opisane w poprzednim paragrafie szablony aktywnych dialogów. Opracowane miary jakości pozwalają śledzić dynamikę procesu nabywania wiedzy w pamięci semantycznej.

Godny jest zaznaczenia fakt dokonania ilościowego ujęcia kompetencji językowych programu komputerowego, co powszechnie oceniane jest jakościowo (np. test Turinga).

5.2.1 Struktura sieci semantycznej

Przeprowadzenie eksperymentów z automatycznym budowaniem przestrzeni semantycznych na dużą skalę, opartą o słowniki maszynowe wymaga wpierw utworzenia dobrze opisanej ontologii dla jednego obszaru, na którym można zweryfikować przyjętą metodologię akwizycji wiedzy. Utworzenie przestrzeni referencyjnej, stanowiącej punkt odniesienia do oceny jakości metodologii pociąga za sobą pytanie o jej kompletność. Przyjęto, że kompletność wyznaczana jest poprzez zastosowanie – odnalezienie obiektu w przestrzeni semantycznej Ψ. Przy takim założeniu wektor CDV jet kompletny, gdy reprezentowany przez niego obiekt jest odnajdywany w grze w 20 pytań. W zastosowaniu do gry słownej kompletność wiedzy w wektorach CDV zapewnia jedynie poprawną separowalność obiektu podczas procesu szukania w przestrzeni semantycznej Ψ, a nie jest pełną wiedzą człowieka związaną z określonym pojęciem.

Podstawowym problemem napotkanym podczas przeprowadzania eksperymentów jest brak istniejących zasobów elektronicznych, zawierających do-

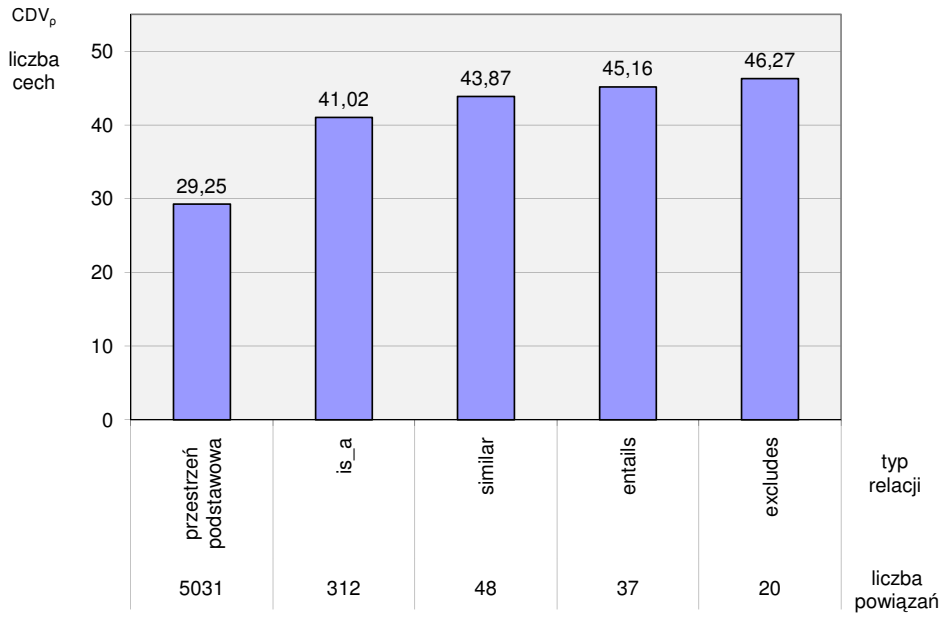
brze zdefiniowane opisy pojęć. Dlatego, konieczne było utworzenie w sposób automatyczny testowej przestrzeni, a następnie jej ręczna edycja i wstępne oczyszczenie w grze słownej. Utworzenie bazowej ontologii ζ_B odbyło się poprzez wybranie zbioru obiektów i cech, a następnie odtworzenia występujących między nimi powiązań z użyciem istniejących zasobów cyfrowych. Testowa sieć semantyczna ζ_B powstała w wyniku nadzorowanej, ręcznej agregacji danych z dostępnych zasobów sieciowych: WordNet [58] (przetworzonej w sposób opisany w paragrafie 4.1), Microsoft MindNet [93], MIT ConceptNet [49], Sumo/Milo Ontology [63], dołączeniu danych pozyskanych w procesie aktywnego szukania (opisanego w paragrafie 4.2.1) oraz ręcznej edycji. Przyjęto, że do pamięci semantycznej wprowadzane są dane, które pojawiły się, w co najmniej dwóch wyżej wymienionych zasobach.

Dane w tak utworzonej sieci semantycznej stanowią punkt wyjściowy dla zbudowania dobrej ontologii (w sensie gry w 20 pytań), dla wybranej dziedziny. Docelowo wiedza opisana przez taką sieć semantyczną powinna być wynikiem interakcji z wieloma użytkownikami, jako że wspólna, zdroworozsądkowa wiedza powinna być wypadkową wiedzy wielu osób. Testowe gry przeprowadzone przed przedstawionymi tu wynikami doświadczeń pozwoliły wstępnie oczyścić dane, poprzez usuwanie powiązań dla cech mało istotnych (tak, jak miało to miejsce przy imporcie danych ze słownika WordNet, opisanych w paragrafie 4.1). Utworzona w ten sposób przestrzeń semantyczna Ψ_B dla królestwa zwierząt stanowiła punkt wyjściowy do eksperymentów pozwalający śledzić dynamikę procesu powstawania nowych powiązań między pojęciami, poprzez interakcję z człowiekiem w grze 20 pytań.

Otrzymaną w wyniku agregacji zasobów semantycznych sieć opisów pojęć ζ_B przyjęto jako bazę wiedzy do przeprowadzenia eksperymentów z aktywnymi dialogami. Składała się ona z 172 obiektów i 475 cech powiązanych 5031 powiązaniami. Wykres 5.1 pokazuje zmianę średniej liczby cech w wektorach opisu CDV_ρ wraz z użyciem procedur interpretacji określonych typów relacji (opisanych w paragrafie 3.1), co ma miejsce przy zamianie sieci semantycznej ζ_B na przestrzeń semantyczną Ψ_B.

Wysokość pierwszego słupka obrazuje, jak wyglądała sieć semantyczna ζ_B (bez użycia interpretacji powiązań). Wartość ta pokazuje liczbę powiązań zapisaną w bazie wiedzy pamięci semantycznej. Kolejne wysokości słupków pokazują zmianę średniej liczby cech w wektorach CDV wraz z użyciem interpretacji określonego typu relacji. Interpretację typów relacji można uważać za rodzaj wnioskowania – dzięki niemu wzrasta wiedza systemu, co obrazuje zwiększenie średniej liczby cech w wektorach CDV dla kolejnych typów relacji, podczas zamiany sieci semantycznej ζ na przestrzeń semantyczną Ψ.

Na wykresie zaobserwować można silną zależność przyrostu wiedzy od liczby powiązań określonego typu. Nadmienić należy, że relacje *similar*, *en-*



Rysunek 5.1: Zmiana liczby cech w CDV w zależności od typu relacji

tails, *excludes* zostały w większej części dodane do pamięci semantycznej ręcznie, ponieważ baza WordNet zawiera niewiele informacji o tego typu powiązaniach, a w innych użytych zasobach takie informacje nie występują. Mała liczba powiązań typu *similar*, *entails*, *excludes* powoduje jedynie nieznaczny wzrost wiedzy systemu wynikający z odpowiedniego dla każdej z nich przetworzenia wektorów CDV.

5.2.2 Ocena jakości pamięci

Algorytm wyszukiwania kontekstowego, obok wyszukiwania informacji w sieci semantycznej i opartej na nim akwizycji wiedzy może zostać również użyty do oceny jakości ontologii, opisującej pojęcia w obrębie określonego zagadnienia. Ponieważ proces wyszukiwania kontekstowego jest silnie związany z wiedzą o języku może zostać użyty do oceny danych lingwistycznych. Wykorzystując proporcję liczby wyszukiwań zakończonych sukcesem N_S, do całkowitej liczby przeprowadzonych wyszukiwań N utworzona została miara jakości ontologii Q, określająca jej przydatność w wyszukiwaniu obiektów przez nią opisywanych. Przy takim podejściu błąd pamięci semantycznej E opisany jest wzorem 5.1.

$$E = 1 - Q \tag{5.1}$$

gdzie:

$$Q = \frac{N_S}{N} \tag{5.2}$$

By wartość opisująca jakość pamięci semantycznej odnosiła się do wszystkich obiektów w niej zapisanych, konieczne byłoby zrealizowanie wyszukiwania osobno każdego z obiektów. Dla dużej liczby obiektów wielokrotne przeprowadzanie takiego procesu staje się bardzo pracochłonne. W związku z tym przyjęte zostało, że ocena pamięci semantycznej odbywać się będzie poprzez losowo wybrane 10 obiektów. Prawdopodobieństwo wybrania do testu pamięci semantycznej obiektu związane zostało rozkładem normalnym liczby cech w wektorze CDV, tak że do testów brane były najczęściej obiekty posiadające średnią liczbę cech.

Uśrednienie wartości miary 5.2 uzyskanej dla wyszukanych 10 losowych obiektów utworzyło miarę jakości pamięci semantycznej. Dla bazowej ontologii ζ_B zwierząt, dla 10 średnio dobrze opisanych obiektów jakość pamięci semantycznej wyniosła $Q = 0,8$. W celu potwierdzenia jakości tak przyjętej miary oceny wykonane zostało pięciokrotne powtórzenie wyżej wymienionej procedury: pięciokrotnie przeprowadzony test 10 losowych obiektów, (w sumie dla 43 różnych obiektów – jako że 7 powtórzyło się w losowaniu) dał średni błąd wyszukiwania $E = 0,18$, co nie odbiega znacznie od wartości $Q = 0,8$ otrzymanej podczas pierwszej oceny.

5.2.3 Ocena szybkości uczenia się nowych pojęć

Wprowadzenie do pamięci semantycznej uczenia, opartego na aktywnych dialogach pozwala na korektę już posiadanej wiedzy, jak również na pozyskiwanie wiedzy o nowych obiektach. Obok jakości przestrzeni semantycznej, określanej przez zastosowanie w wyszukiwaniu znanych jej obiektów, ważne jest również oszacowanie średniej liczby wyszukań (powtórzeń gry) koniecznych do poprawnego rozpoznawania zupełnie nieznanych dotychczas obiektów. Podkreślić należy, że wartość ta wyliczana jest dla nieznanych obiektów i przy założeniu poprawnych odpowiedzi podczas wyszukiwania. Dla obiektów już zapisanych w pamięci wartość ta może się różnić, ze względu na możliwość niepoprawnego ich opisu poprzez CDV (wynikającego z automatycznego ich utworzenia ze słowników maszynowych), a do korekty którego potrzebne może być więcej wyszukań.

Szybkość uczenia się nowych pojęć jest drugim istotnym parametrem mającym wpływ na użyteczność pamięci semantycznej. Wartość opisująca go wyznaczona została jako liczba powtórzeń wyszukiwania obiektu początkowo nieznanego pamięci semantycznej obiektu, która była przeprowadzana do

momentu pierwszego poprawnego odnalezienia. Ponieważ, każde wyszukanie kończy się aktualizacją pamięci, przez co zmienia się stan wiedzy pamięci semantycznej, wartość ta będzie jedynie w przybliżeniu określać liczbę gier należy przeprowadzić, by określony obiekt był poprawnie identyfikowany. By zapobiec zarzutowi torowania (polegającego na tym, że system lepiej wyszukuje ostatnio odnalezione obiekty) przyjęto, że każdy z 10 nowo uczonych obiektów będzie wyszukiwany na przemian z dwoma najbardziej do niego podobnymi. W pamięci semantycznej najbardziej podobnymi do wyszukiwanego obiektu będą te dwa, które w podprzestrzeni O(A) utworzonej po końcowym pytaniu mają najmniejszą odległość od wektora ANSW. Procedura ta w pierwszym kroku dokonuje więc 30 wyszukań. W zależności od rezultatu wyszukań pierwszych 10 testowych obiektów ustalony zostaje kolejny zbiór obiektów (które nie zostały odnalezione poprawnie i dla każdego z nich dwa najbardziej podobne), które poszukiwane są w kolejnym kroku procedury testowej. W miarę poprawnego odnajdywania obiektu (czyli zapamiętywania pojęć), w kolejnych krokach procedury testowej liczba wyszukiwanych obiektów się zmniejsza. Dla przeprowadzonych eksperymentów z uczeniem pamięci semantycznej 10 nowych obiektów potrzebne było pięciokrotne powtórzenie opisanej procedury (naprzemiennych wyszukań nieznanego obiektu i dwóch najbardziej do niego podobnych). Zaznaczyć należy, że ostanie 2 powtórzenia procedury testowej trenowały tylko jedną z dziesięciu nowych obiektów.

Średnia liczba wyszukań V_n konieczna do poprawnego rozpoznawania obiektu przez pamięć semantyczną wyznaczona została jako liczba testowych wyszukiwań zakończonych niepowodzeniem N_f (powtarzana do momentu, gdy wyszukiwanie zakończone zostanie sukcesem) do liczby użytych w testach nowych (nieznanych pamięci semantycznej) obiektów N_{kt}. Wartość ta opisana została formułą 5.3.

$$V_n = \frac{N_f}{N_{kt}} \tag{5.3}$$

Dla bazowej przestrzeni semantycznej Ψ_B otrzymana wartość wyniosła $V_n = 2,79$. Określa ona średnią liczbę wyszukań dla nieznanego z początku obiektu, którą należy wykonać, aby była poprawnie (w sensie wyszukiwania) zakodowana w pamięci semantycznej.

Należy mieć na uwadze fakt, że wartość ta jest wartością określającą jedynie w przybliżeniu liczbę wyszukań nowego obiektu, jaką należy wykonać by system ją poprawnie rozpoznawał. Wielkość ta zależy od stopnia wytrenowania całej pamięci Q i średniej liczby cech we wszystkich wektorach CDV_ρ.

Średnia liczba cech w CDV, zwłaszcza dla obiektów podobnych do nowo dodawanego jest tu bardzo istotna, ponieważ miejsce wstawiania nowego obiektu w przestrzeni semantycznej skutkować będzie dla dużej liczby obiek-

tów podobnych do siebie, większą liczbą powtórzeń procesu wyszukiwania. Dla obiektów znacząco różnych od już istniejących w pamięci wartość V_n może być nawet równa jedności. Jeśli już istniejące w pamięci semantycznej obiekty są dobrze opisane (jakość $Q > 0,9$), rozróżnienie będzie bardziej precyzyjne, przez co liczba niepoprawnych wyszukań nieznanego obiektu będzie mniejsza.

5.2.4 Ocena kompletności CDV uzyskanych w aktywnym uczeniu

W paragrafie poprzednim opisane zostały dwa parametry pozwalające ocenić pamięć semantyczną:

Q – określający poprawność danych zapisanych w strukturach kontekstowych,

V_n – określający szybkość nabywania nowej wiedzy.

Obok nich, istotna jest również kompletność otrzymanych wektorów CDV. Przedstawiona w tym paragrafie miara kompletności umożliwia ocenę zastosowania metody aktywnego uczenia do uzyskiwania wektorów opisów zawierających oczekiwane cechy, których zastosowanie ze względu na ogólność reprezentacji wiedzy może być dużo szersze niż tylko gra w 20 pytań. Kompletność wyznaczana jest przez porównanie wektora CDV tworzonego w procesie uczenia do wektora standardu (GS – *Golden Standard*), będącego docelowym zbiorem cech trenowanego obiektu. Wartość kompletności opisuje podobieństwo cech między nowym obiektem (NO – nowy obiekt) powstałym w pamięci semantycznej, poprzez interakcję z człowiekiem, a standardem (GS). Do oceny aktywnego uczenia użyte zostały 4 miary kompletności:

1. S_d – miara podobieństwa oparta na liczbie cech w wektorach CDV.

$$S_d = Dens(GS) - Dens(NO) \tag{5.4}$$

gdzie:

$Dens(GS)$ – oznacza liczbę cech występujących w wektorze standardu

$Dens(NO)$ – oznacza liczbę cech występujących w nowo tworzonym wektorze opisu

Wartość S_d jest liczbą cech, które pojawiły się w wektorze CDV w procesie aktywnego uczenia, odniesiona do standardu. Miara ta pozwala określić, jak odległy w sensie liczby cech jest nowo powstający CDV od przyjętego standardu.

2. S_{GS} – miara podobieństwa dwóch wektorów CDV oparta na pokrywaniu się cech – występowaniu cech wektora NO w GS.

$$S_{GS} = |\cap(\{C_{GS}\}, \{C_{NO}\})| \tag{5.5}$$

gdzie:

|| oznacza liczność zbioru cech

{} zbiór cech wektora CDV (odpowiednio NO i GS)

Wartość S_{GS} oznacza liczbę cech z wektora NO, które znalazły się w wektorze standardu – GS. Miara ta bierze pod uwagę tylko cechy z wektora GS, przez co w dokładniejszy sposób od S_d obrazuje proces, w którym NO zbliża się do GS.

3. Dif_w – jest miarą kompletności, opartą na średniej różnicy wartości cech opisujących wagi w poszczególnych powiązań dla cech, które występują w NO i GS.

$$Dif_w = \frac{1}{n}\sum_i \text{abs}(CDV(NO)_i - CDV(GS)_i) \tag{5.6}$$

Wartość Dif_w pokazuje, jak bardzo różnią się wartości powiązań, dla cech pokrywających się w wektorach NO i GS. Miara ta pozwala ocenić, jak bardzo nowo powstający wektor NO różni się od GS w obszarze wspólnych cech – czy nowe wartości przypisywane powiązaniom pojęcie – cecha nie odbiegają od oczekiwanych. Dif_w jest miarą pozwalającą zaobserwować nieprawidłowe wartości powiązań, podczas gdy poprzednia miara S_{GS} brała pod uwagę jedynie wystąpienie powiązania pojęć z cechą.

4. S_{NO} jest miarą podobieństwa dwóch wektorów CDV opartą na różnicy cech – opisuje ona liczbę cech wektora NO, która nie znalazła się w wektorze GS.

$$S_{NO} = |\Delta(\{C_{NO}\}, \{C_{GS})\}| \tag{5.7}$$

gdzie:

|| oznacza liczność zbioru cech

{} zbiór cech wektora CDV (odpowiednio NO i GS)

Wartość miary S_{NO} opisuje liczbę cech, które pojawiły się w opisie nowego obiektu, a nie zostały zdefiniowane w GS. Różnica cech pojawiająca się między wektorami cech NO i GS wynika z wprowadzenia opisanego w paragrafie 3.4 losowego wybierania cech służących do zadawani pytań. Utworzone w ten sposób pytania pozwalają na pozyskiwanie bogatszej wiedzy – cechy jakie może uzyskać obiekt w trakcie akwizycji wiedzy mogą pochodzić ze zbioru wszystkich cech, a nie tylko są wyznaczone największym IG w $O(A)$.

Zdefiniowane powyżej cztery miary użyte zostały do oceny zastosowania aktywnych dialogów do akwizycji wiedzy. Pomiar ich wartości odbył się z użyciem następującej procedury eksploracji przestrzeni semantycznej:

1. Losowy wybór obiektu do testu, z prawdopodobieństwem proporcjonalnym do liczby cech w wektorze CDV. Wybór taki faworyzuje więc lepiej opisane obiekty.

2. Utworzenie z wektora CDV wybranego obiektu wektora standardu GS.

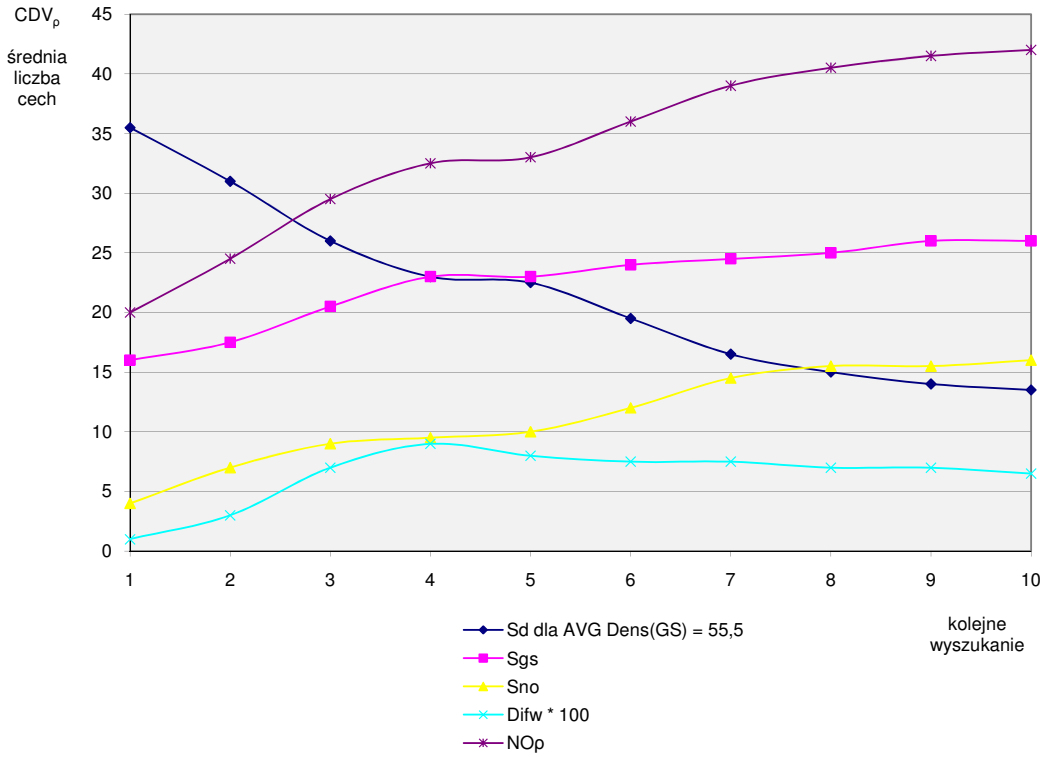
3. Poprzez ręczną korektę GS dodanie cech, które arbitralnie powinny się znaleźć jako typowe dla wybranego obiektu.

4. Usunięcie wybranego oiektu z przestrzeni semantycznej.

5. Rozpoczęcie procesu wyszukiwania i uczenia.

Procedura ta przeprowadzona została dla wylosowanych pięciu obiektów. Uzyskane uśrednione wyniki przedstawiają wykresy na rysunku 5.2, opisujące proces zmiany średniej liczby cech nowo powstających wektorów opisu w odniesieniu do przyjętych standardów. Dynamika tego procesu pokazana została w funkcji kolejnych gier.

Otrzymane wyniki wskazują na użyteczność metody aktywnego uczenia w pozyskiwaniu nowych cech do opisu obiektów w pamięci semantycznej – nowe wektory CDV w dość szybki sposób uzyskują nowe cechy. Obrazuje to wykres wartości NO_ρ, ($NO_\rho = S_{NO} + S_{GS}$) pokazujący średni przyrost cech w nowo powstających wektorach CDV. Wraz z przeprowadzonymi grami zwiększa się liczba cech w nowo powstających wektorach CDV, które pokrywają się z przyjętymi wektorami standardu (wykres S_{GS}). Dla testowego zbioru wyszukiwanych obiektów średnia liczba gier, po której uzyskano poprawne wyszukanie wyniosła $V_n = 2,67$. Nadmienić należy, że po pierwszym



Rysunek 5.2: Zmiana średniej liczby cech dla pięciu nowo powstających wektorów CDV w funkcji kolejnych wyszukań

poprawnym wyszukaniu, w trakcie przeprowadzanych dalszych doświadczeń wszystkie obiekty były poprawnie odnajdywane, co wskazuje na poprawność wektorów CDV w pamięci semantycznej w sensie separowalności. Najprawdopodobniej w sytuacji, gdyby wielu różnych ludzi realizowało tę procedurę, wartość V_n nieznacznie by się zwiększyła, obrazując w ten sposób koszt w postaci liczby wyszukań, jaki musi zostać poniesiony na pozyskanie wypadkowej wiedzy zdroworozsądkowej.

Wartość S_d liczona jest dla wartości średniej cech w wektorze GS wyznaczanej dla wszystkich pięciu testowych obiektów. Dla wybranych obiektów średnia liczba cech w wektorach CDV standardu wyniosła $Dens(GS) = 55,5$. Malejący trend S_d wskazuje na stopniowe zbliżanie się liczby cech w nowo uczonym wektorze do liczby cech w wektorze standardu. Oczekiwać można, że przy dalszym wyszukiwaniu wartość ta będzie zbliżać się do zera, a w dalszej perspektywie może nawet przekroczyć liczbę cech w standardzie. Szybkość spadku jednak na pewno będzie maleć – coraz mniej nowych cech będzie pozyskiwanych podczas pojedynczej gry. W wypadku wyłączenia randomizacji zapytań, wartość ta ustabilizowałaby się wokół pewnej stałej, ponieważ nie

zmieniałaby się liczba cech w wektorach CDV testowych obiektów, w momencie, gdy pamięć semantyczna się ich nauczyła (wszystkie były poprawnie wyszukiwane). Dla testowych obiektów wartość S_d nie miałaby szansy wtedy się zmienić i dopiero, gdy wyszukanie innego pojęcia spowodowałoby aktualizację bazy wiedzy, a przez to zmieniony zostałby ranking zysków informacyjnych dla cech, możliwe by było pozyskanie nowej wiedzy – innych cech.

Naniesione na wykresie wartości S_{NO} opisują liczbę cech, która nie pokrywa się z Golden Standard – obrazuje to sytuację, w której do wektora opisu NO wchodzą cechy niebędące pierwotnie zdefiniowane w standardzie. Wynika to z niepełnego zdefiniowania wektora GS, dla przestrzeni semantycznej (wielkości 475 cech) oraz użycia losowej metody wyboru cech do utworzenia zapytań. Wartość ta powstaje przede wszystkim z cech typu negatywnego, 94% cech w podzbiorze cech nie będących w standardzie otrzymuje wartości powiązania < 0, co wskazuje na dobre zdefiniowanie wektorów standardów (jako wiedzy pozytywnej).

Po przeprowadzeniu testowych gier liczba cech, która nie pokrywa się ze standardem jest bliska 30% wszystkich cech w nowo powstającym wektorze. Nie ma to jednak zasadniczego wpływu na jakość wyszukiwania – co wskazuje brak błędnych wyszukań obiektów po wytrenowaniu pamięci, pozwalającej wyznaczyć wartość $V_n = 2,67$. Pokrycie standardu opisane zostało wykresem S_{GS}. Zmniejszająca się szybkość pozyskiwania wiedzy obrazowana tą miarą wskazuje, że przy dalszych grach więcej cech nabywanych będzie z poza zbioru cech GS.

By wartości parametru Dif_w (opisujące różnice w wartościach w powiązań cech) były widoczne na wykresie pomnożone zostały przez 100. Dla tego wykresu oś Y nie opisuje średniej liczby cech, a średnią różnicę w wartościach powiązań cech (w). Wyraźnie widać zmniejszającą się wielkość różnic, dla pokrywających się cech w wektorach CDV GS i NO, co więcej różnice te są minimalne (musiały być przeskalowane, by można umieścić je na wykresie). Oczekiwać można, że w perspektywie dalszych gier ustabilizują się one wokół niewielkiej wartości – będącej wypadkową różnicy poglądów różnych ludzi na to, jak określone cechy wiążą się ze wskazanymi testowymi obiektami.

Przedstawione w tym rozdziale wyniki zastosowania wyszukiwania kontekstowego wykazują II tezę pracy „algorytm wyszukiwania może zostać użyty do weryfikacji i akwizycji wiedzy semantycznej", demonstrując możliwość pozyskiwania wiedzy semantycznej, poprzez interakcję z człowiekiem z użyciem aktywnych dialogów.

Miara jakości oparta na rezultacie wyszukiwania pozwala ocenić nabywaną przez komputer wiedzę o języku, która jest adekwatna do wiedzy posia-

danej przez człowieka, co potwierdzają przeprowadzone eksperymenty.

5.3 Porównanie wyszukiwania algorytmu kontekstowego i ludzi

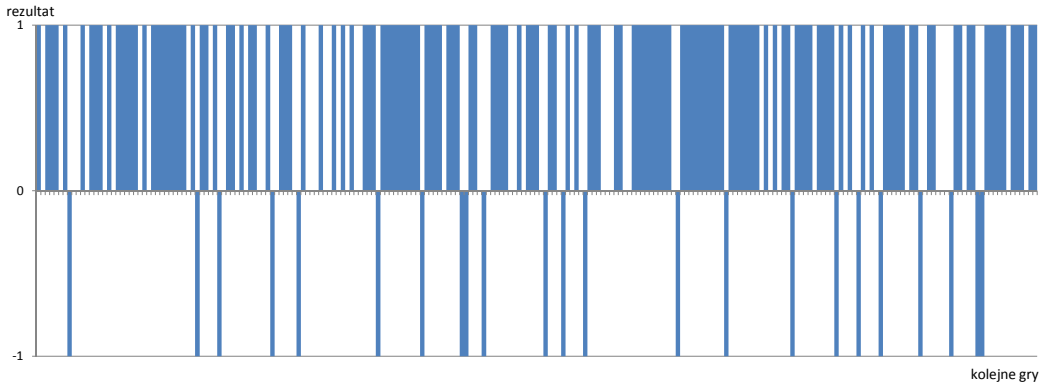
Algorytm wyszukiwania kontekstowego zastosowany do gry w dwadzieścia pytań, w określonej dziedzinie może zostać porównany ze sposobem, w jaki przeprowadziliby takie zadanie ludzie. Porównanie takie pozwala ocenić szybkość wyszukiwania używającego kontekstu V_s mierzonego jako liczba zapytań konieczna do odnalezienia obiektu.

Doświadczenia, których wyniki zaprezentowane zostały w tym rozdziale odbyły się w czterech grupach osób. Osoby biorące udział w eksperymencie, w pierwszej kolejności przeprowadziły grę w 20 pytań między sobą, co pozwoliło ustalić średnią szybkość wyszukiwania kontekstowego ludzi dla określonej dziedziny. W kolejnej fazie eksperymentu, osoby biorące udział w doświadczeniu przeprowadziły gry z pamięcią semantyczną, dzięki czemu określona została jej jakość Q i szybkość wyszukiwania V_s. Otrzymane wartości odniesione zostały do uzyskanych wyników podczas doświadczeń przeprowadzonych z aktywnymi dialogami.

Zamknięty przebieg eksperymentu pozwala prześledzić i ocenić dynamikę procesu zmiany pamięci semantycznej, poprzez interakcję z człowiekiem. Wyniki uzyskane pozwalają porównać algorytm wyszukiwania kontekstowego z wyszukiwaniem, jakiego używają ludzie podczas realizacji podobnego zadania. Przeprowadzone doświadczenia z grupą 86 osób dają jedynie przybliżone wartości. Aby były one bardziej wiarygodne, konieczne byłoby przeprowadzenie doświadczenia na próbce przynajmniej o rząd większej. Rozbicie doświadczenia na grupy pozwala z większym prawdopodobieństwem poprawności przyjąć uśrednione wyniki, jako rezultat eksperymentu.

Pamięć semantyczna na początku doświadczenia zawierała 197 obiektów opisanych 529 cechami, o średniej liczbie cech w wektorze $\mathrm{CDV}_\rho = 50{,}64$. Do oceny jakości pamięci przyjęto miarę 5.2, której wartość dla testowej procedury wyniosła $Q = 0,8$.

Przeprowadzone doświadczenie z grupą osób pozwalają ocanić następujące parametry pamięci semantycznej:



Rysunek 5.3: Wykres opisujący kolejne rezultaty gier

1. Jakość pamięci i ocena miary jakości.

   Przeprowadzone przez grupę osób gry z pamięcią semantyczną pozwalają ocenić zarówno samą pamięć (jakość zapisanej w niej wiedzy), jak i utworzoną miarę jakości, liczoną w testowej procedurze dla 10 średnio dobrze opisanych obiektów (5.2.2). Wyniki gier przeprowadzonych przez badane grupy z pamięcią semantyczną zaprezentowane zostały na wykresie na rysunku 5.3. Na wykresie naniesione zostały rezultaty 227 kolejnych gier przeprowadzonych w trakcie eksperymentu przez osoby z czterech testowych grup. Dla zakodowania uzyskanych rezultatów gier przyjęto następujące wartości: 1 – gra zakończona sukcesem, 0 – gra zakończona porażką (niepoprawianym odnalezieniem obiektu) -1 – gra zakończona porażką, ze względu na brak wyszukiwanego obiektu w pamięci semantycznej.

   Dla przeprowadzonych wszystkich gier średni błąd pamięci semantycznej wyniósł $E = 0,36$, co różni go od wartości uzyskanej w procesie testowej procedury $E = 0,2$. Zauważyć należy jednak fakt, że procedura testowa (losująca średnio dobrze opisane obiekty z pamięci) nie bierze pod uwagę sytuacji, w których wyszukiwane mogą być zupełnie nieznane obiekty. Usuwając nowe obiekty z oceny miary jakości pamięci semantycznej, uzyskanej z gier przeprowadzonych przez osoby biorące udział w eksperymencie wartość jakości $Q = 0,81$ zbliża się do wartości uzyskanej poprzez procedurę testowej oceny. Wskazuje to na jej użyteczność do pomiaru jakości danych w pamięci semantycznej.

   W trakcie eksperymentu pamięć semantyczna pozyskała 46 nowe obiekty, których z oczywistych powodów nie była w stanie poprawnie rozpoznać. Stanowią one większą część błędów wyszukiwania (56,09%).

2. Szybkość wyszukiwania.

   Przeprowadzone eksperymenty z grupą osób wykonane zostały z parametrem algorytmu, wyłączającym możliwość przedwczesnego zgadywania. Dzięki temu pamięć semantyczna otrzymuje więcej danych oraz wzrasta jej jakość w sensie liczby poprawnych gier.

   Przeprowadzone gry są logowane, co pozwala na ich późniejsze odtworzenie: posiadając odpowiedzi użytkownika możliwe jest ponowne przeprowadzenie gry dla zadanego pojęcia, bez konieczności angażowania w ten proces człowieka (którego aktywność jest tu najcenniejszym zasobem), a używając już wcześniej przeprowadzonych schematów wyszukań. Pozwala to na sprawdzenie szybkości wyszukiwania V_s z włączoną heurystyką zgadywania. Logowanie pytań w celu późniejszego odtworzenia gier wymaga wyłączenia randomizacji pytań – tak by w procesie automatycznego odtwarzania gry padały te same pytania. Dopasowanie to nie jest jednak wystarczające, ponieważ pamięć semantyczna zmienia swoją strukturę po każdej grze. Ma to wpływ na ranking zysków informacyjnych cech (IG), wskazujący te, o które najlepiej jest się zapytać. Prowadzi to do minimalnych różnic w wyborze cech wybieranych do utworzenia zadawanych przez system pytań, w kolejnych grach dotyczących tego samego pojęcia. W związku z tym, by możliwe było przeprowadzenie w sposób automatyczny gier (na podstawie już przeprowadzonych przez ludzi) przyjęto, że w sytuacji gdy zadane zostanie przez system pytanie, które nie było użyte w trakcie zalogowanych gier, wykorzystana zostanie odpowiedź wynikająca z wiedzy zapisanej w wektorze CDV wyszukiwanego obiektu. W trakcie przeprowadzania doświadczenia sytuacja ta miała miejsce w mniej niż 5% wszystkich zadanych pytań. Wartość ta pozwala przyjąć, że powyższe dopasowanie nie będzie powodować znacznego zafałszowania wyników. Drugim czynnikiem mogącym mieć wpływ na zniekształcenie wyniku automatycznego testowania jest fakt, że przeprowadzenie gry skutkuje aktualizacją pamięci, przez co staje się ona lepiej wytrenowana. Dlatego pomiar jakości, po przeprowadzeniu gry człowieka z systemem, na podstawie tej samej gry nie jest do końca sprawiedliwy (bada stan pamięci po korekcie, która już zaszła), jednak do oceny szybkości przyjąć można, że wartość średniej liczby kroków V_s zmniejszać się będzie w miarę treningu pamięci semantycznej i uzyskana wartość jest oszacowaniem odgórnym.

   W miejscu tym zaznaczyć należy również fakt pojawiającej się trudności, w pełni obiektywnym porównaniu wyszukiwania, używanego przez ludzi i algorytmu. Wynika ona z różnych baz wiedzy wykorzystywanych

przez algorytm i przez człowieka.

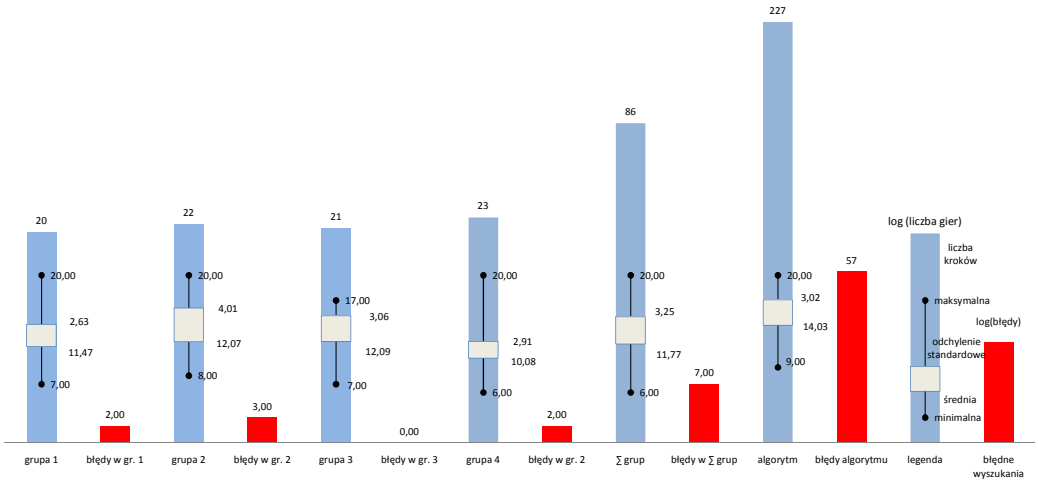
Wiedza w pamięci semantycznej jest niepełna, nawet w sensie powiązań pomiędzy obiektami a cechami je opisującymi (CDV_ρ = 50,64 cech/obiekt, co daje średnio około 10%-owe wypełnienie wektora CDV). Kompletność wiedzy pamięci semantycznej o cechach zadanych obiektów wymagałaby 100% pokrycia wektorów CDV, czego realizacja dla 10 obiektów jest trudna, a już dla 100 ze względu na konieczność zaangażowania znacznych zasobów ludzkich praktycznie niemożliwa. Niekompletność wektorów CDV, jak i zmniejszona liczba cech może zarówno wpływać pozytywnie na zmniejszenie liczby pytań potrzebnych do zakończenia gry, jak również poprzez brak pełnego opisu obiektów na pogorszenie sposobu, w jaki zorganizowana jest informacja w pamięci semantycznej komputera, co skutkować będzie słabszymi wynikami wyszukiwania. Widać tu wyraźnie rozdźwięk między wymogiem kompletności wiedzy, a jej optymalnością w sensie wyszukiwania.

Wiedza dziedzinowa u człowieka nie jest wyodrębniona z wiedzy całościowej. Zanurzenie jej w wiedzy ogólnej, z jednej strony pozwala zadawać wydajniejsze (lepiej separujące) pytania, z drugiej strony liczba potencjalnych pytań jakie może zadać człowiek (nawet dla ograniczonej dziedziny) jest dużo większa niż ta, którą może posiąść system informatyczny (który ograniczony jest liczbą znanych cech).

Przedstawione w tym paragrafie gry (rysunek 5.3) przeprowadzone przez testową grupę osób użyte zostały do oceny szybkości wyszukiwania algorytmu V_s i człowieka. Oszacowanie szybkości wyszukiwania obiektów przez ludzi w określonej dziedzinie przeprowadzony został na czterech grupach ludzi, składających się odpowiednio z: 20, 21, 22 i 23 osób.

W doświadczeniu przeprowadzone zostały 93 gry zrealizowane przez ludzi między sobą. Do oceny szybkości wyszukiwania użyte zostało 86 z nich, które zakończone zostały w nie więcej niż 20 pytaniach. Wystąpienie gier przeprowadzanych przez ludzi między sobą, które trwają dłużej niż 20 pytań, wskazuje na istnienie rozbieżności między pamięciami semantycznymi poszczególnych ludzi, zwłaszcza dla mniej popularnych pojęć. Gry nie zakończone lub przekraczające ten próg wykluczone zostały z badania.

Uzyskane wyniki prezentuje wykres na rysunku 5.4. Pierwsze cztery grupy słupków (grupa – niebieski i czerwony słupek) obrazują uśrednioną liczbę pytań koniecznych do odnalezienia obiektu, uzyskaną w każdej z grup badanych osób, w których gry przeprowadzane były między ludźmi. Sumaryczne otrzymane wyniki (piąty, niebieski słupek) porów-



Rysunek 5.4: Porównanie liczby pytań stawianych przez algorytm 20q i ludzi

nane zostały do wyszukania przeprowadzonego z pamięcią semantyczną (niebieski słupek szósty). Na wykresie czerwonym kolorem zaznaczone zostały wartości określające liczby nie odnalezionych obiektów w każdym z przypadków. Automatyczne gry przeprowadzone w celu uzyskania wartości naniesionych w grupie szóstej słupków wykorzystywały gry przeprowadzone przez testową grupę (przedstawione na rysunku 5.3). Naniesiona na rysunku 5.4 ostatnia grupa słupków stanowi legendę do wykresów.

Otrzymane rezultaty pozwalają ocenić, że człowiek potrzebuje do odnalezienia obiektu w obrębie dziedziny zwierząt około 12 pytań, algorytm dla heurystyki (3.14) pozwalającej na wcześniejsze niż po 20 pytaniach zgadywanie – ok. 14. Wartość ta obarczona jest jednak błędem $E = 0,25$, wynikającym zarówno z jakości wiedzy, jak i z przyjęcia heurystyki zgadywania pozwalającej na ograniczenie liczby stawianych pytań. Możliwa jest minimalizacja tego błędu poprzez lepsze wytrenowanie pamięci semantycznej i/lub zwiększenie liczby pytań do odnajdywania obiektu. Przeprowadzone doświadczenie, którego celem była ocena jakości wyszukiwania uzyskanego w 20 pytaniach, zrealizowane zostało poprzez automatyczne gry (realizowane na podstawie wcześniejszych interakcji z ludźmi). Otrzymana w nim wartość błędu pamięci $E = 0,19$ określa, że około 80% obiektów jest poprawnie zapisanych w pamięci.

Uzyskane tu wyniki jakości wyszukiwania są lepsze niż wyniki uzyskane w grach pierwotnie przeprowadzonych przez testową grupę osób, nawet

pomimo wprowadzenia heurystyki zgadywania. Wynika to z faktu, że automatyczne gry przeprowadzane są dla obiektów znanych już pamięci semantycznej oraz częściowo już wytrenowanej przez testową grupę osób.

Wskazuje to na szybkie uczenie się nowych pojęć i poprawne kodowanie nowych obiektów w pamięci semantycznej. Widać to porównując wyniki automatycznego pomiaru błędu pamięci semantycznej, z oceną opartą na przeprowadzanych grach przez użytkowników. Poprawa jakości wyszukiwania wynika również z wyłączenia użycia randomizowanych zapytań, co ma wpływ na poprawę jakości drzewa klasyfikującego, odbywa się to jednak kosztem ilości pozyskiwanych danych. Obserwacja ta, spowodowała modyfikację wdrożonej wersji gry, w której randomizacja pytań włączana jest dopiero po przekroczeniu progu wytrenowania pamięci $E = 0,1$ liczonego dla 100 ostatnich gier.

Wynik porównania algorytmu i człowieka wypadł na niekorzyść tego pierwszego. Wynika to przede wszystkim ze stopnia wytrenowania pamięci semantycznej oraz z bogatszej bazy wiedzy posiadanej przez ludzi, którzy używają do gry dodatkowych informacji, nie tylko wiedzy o cechach obiektów, ale również wiedzy o np. popularnościach pojęć, które mogą być wyszukiwane.

Wiedzę taką można również wprowadzić do algorytmu wyszukiwania zmieniając sposób wyboru cechy. W algorytmie używana jest miara tworząca ranking cech na podstawie entropii związanej z każdą cechą (3.4). Możliwą zmianą jest obliczenie dla każdej cechy c entropii warunkowej:

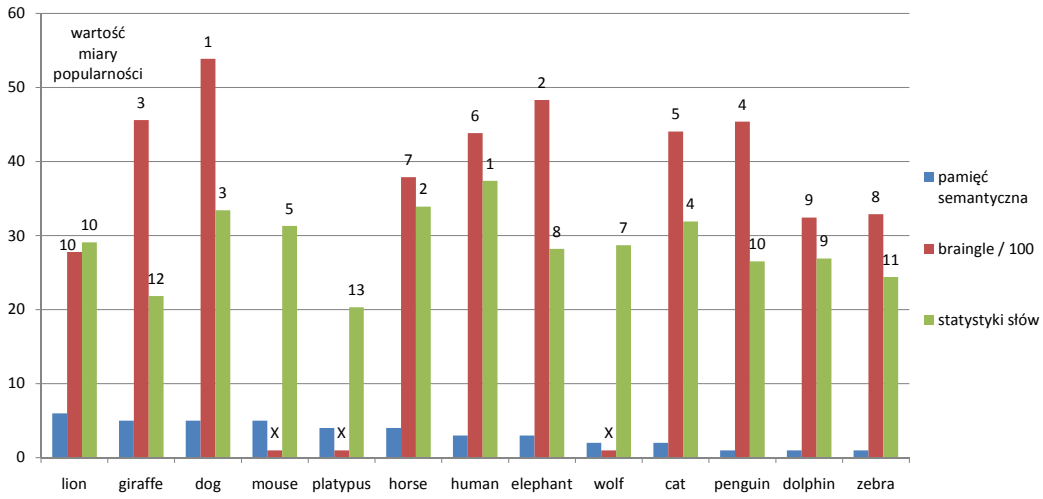
$$H(c/o) = -\sum_{i=0}^{N}\sum_{j=0}^{M} p(c,o)\log\frac{p(c,o)}{p(o)} \tag{5.8}$$

gdzie:

N jest liczbą wszystkich cech

M jest liczbą obiektów jakie mają cechę, dla której liczona jest entropia

$p(o)$ jest dane rozkładem prawdopodobieństwa, będącego popularnością obiektu o



Rysunek 5.5: Popularność słowa reprezentującego obiekt, dla którego najczęściej przeprowadzane były gry

3. Typowość wyszukiwania obiektów.

   W sytuacji rozszerzenia algorytmu o wybór cechy oparty na wartości entropii warunkowej użyte zostało prawdopodobieństwo $p(o)$ – wybrania do gry określonego obiektu. Prawdopodobieństwo to wyznaczane jest ze wszystkich przeprowadzonych gier i w miarę, jak określony obiekt jest częściej przez ludzi poszukiwany – rośnie.

   Rozszerzenie takie pozwala uwzględnić oczywisty fakt, że w grze niektóre obiekty są częściej wyszukiwane od innych. Popularność pojęcia pozwala włączyć do procesu wyszukiwania dodatkową wiedzę – o typowości obiektu.

   Interesująca jest charakterystyka tej częstości wskazująca typowość najpopularniejszych pojęć. Na rysunku 5.5 przedstawiono wysokościami niebieskich słupków liczby wyszukań dla najpopularniejszych obiektów w pamięci semantycznej. Wartości te uzyskane zostały dla rozróżnialnego, pojedynczego użytkownika – wykluczono sytuację, gdy ta sama osoba wyszukiwała ten sam obiekt. Na wykresie zamieszczono również miarę popularności słowa (opisaną miarą 4.1) – słupek zielony, oraz częstość (przeskalowaną/100) wyszukiwania najpopularniejszych obiektów w Braingle Zoo (słupek czerwony). Wartości nad słupkami podają pozycję w rankingu popularności, uzyskaną przez każde ze słów w zależności od przyjętej miary.

   Pomimo, że kolejność pojęć w rankingu popularności utworzonego przez

te trzy podejścia nie pokrywają się w pełni, zauważyć można powtórzenie się 70% najbardziej popularnych obiektów w grach przeprowadzonych z pamięcią semantyczną i z Braingle Zoo. Powtórzenie to zaszło pomimo różnicy dwóch rzędów w liczbie przeprowadzanych wyszukań w obu systemach.

Opracowane miary częstości słów (4.1) potencjalnie mogłyby zostać zweryfikowane, poprzez częstość wyszukiwania obiektu w pamięci semantycznej. Wymagałoby to jednak przeprowadzenia zdecydowanie większej liczby gier z pamięcią semantyczną. Porównując do Braingle Zoo widać pewną zależność pomiędzy popularnością słowa a popularnością wyszukiwania obiektu.

Analizując częstości wyszukiwanych obiektów udostępniane przez Braingle Zoo wyciągnąć można wniosek, że oparta na interpretowanych typach relacji reprezentacja wiedzy semantycznej daje bardzo dobre rezultaty akwizycji wiedzy: < 1000 przeprowadzonych wszystkich gier do osiągnięcia $Q > 0,8$ jakości, w porównaniu do > 3000 wyszukań dla pojedynczych (najpopularniejszych) obiektów w Braingle.

4. Stopień podobieństwa pytań zadawanych przez maszynę i człowieka.

   Obok oceny szybkości wyszukiwania obiektów w pamięci semantycznej V_s interesujący jest również sposób, w jaki jest to realizowane. Podobieństwo wykonania takiego zadania przez komputer do działalności człowieka jest jednym z istotniejszych elementów na drodze do opanowania przez maszyny kompetencji językowych.

   Tabele 5.1 i 5.2 prezentują przykładowe wyszukania – kolejne pytania użyte do odnalezienia obiektu *lion*, zrealizowane przez: Braingle Zoo, q20.net, człowieka i pamięć semantyczną.

   Podobnie jak prezentacja popularności pojęć, ocena porównania jakości zadawanych pytań (w sensie wiedzy zdroworozsądkowej) może zostać dokonana jedynie subiektywnie. Subiektywizm w ocenie kompetencji językowych programu komputerowego jest ogólnym problemem w ocenie poziomu realizacji języka naturalnego w maszynach. Widoczny jest on również w powszechnie ustalonym teście Turinga [91], jak również w organizowanych konkursach o nagrodę Loebnera[2], gdzie weryfikacja odbywa się poprzez jakościową ocenę dokonywaną przez człowieka, a nie zdefiniowane kryterium ilościowe.

   Z porównania przedstawionych 4 podejść do odnajdywania obiektów zauważyć można różnice w podejściu opartym na wypełnianiu macie-

[2] http://www.loebner.net/Prizef/loebner-prize.html ,XII 2007

lp	Braingle zoo	q20.net
1	Is it from Harry Potter?	It is classified as Animal?
2	Does it usually jump to get around?	Is it a vegetarian?
3	Does it have a long tongue	Do you clean it regularly?
4	Does its name contain an evil creature?	Does it have striped fur?
5	Is it an insect?	Can you see it in a zoo?
6	Does it live in the ocean?	Is it black?
7	Can it fit in a shoe box when fully grown?	Is it very large?
8	Does it have wings?	Is it bigger than a sofa?
9	Does it have paws?	Is it larger than a microwave oven?
10	Is it native to North America?	Does it like to play?
11	Does it live in Africa?	Is it soft?
12	Is it a feline?	Does it have tough skin?
13	Is it golden colored?	Does it like to run?
14	Can it be colored black?	Is it a type of dog?
15	Does it have stripes?	Can you lift it?
16	Is it colorful?	Is it multicoloured?
17	Does it have spots?	Does it live in the jungle?
18	Does it live primarily in central Africa?	Can it be heard?
19	Does it use its tail to balance?	I guessed that it was a cougar?
20	Does it have stripes?	I guessed that it lion?

Tabela 5.1: Porównanie realizacji gry dla obiektu *lew* przez Braingle Zoo i q20.net

rzy pytań i obiektów (tabela 5.1), i użycia bazy wiedzy zwierającej powiązania między pojęciami (tabela 5.2). Brak takiej organizacji danych wyraźnie jest widoczny dla 20q.net, gdzie część pytań wydaje się nie być związana z wyszukiwanym obiektem, a nawet dziedziną. Również kolejność zadawanych pytań odbiega znacznie od sposobu w jaki człowiek realizowałby takie zadanie.

Na minus interfejsu w języku naturalnym pamięci semantycznej zaliczyć należy pojawianie się niegramatycznych pytań. Wynikają one z generowania pytań z atomów wCRK, a nie używania wcześniej przygotowanych zdań. Można zrobić znacznie lepszy program do generacji pytań, jednak jest to bardzo pracochłonne i do demonstracji elementarnych kompetencji językowych nie jest konieczne. Drugim problemem wynikającym z umożliwienia użytkownikowi wprowadzania wiedzy, poprzez aktywne dialogi do pamięci semantycznej są przekłamania (literówki) w wyrazach będącymi składowymi CRK. Słowa te oznaczają

lp	człowiek	wyszukiwanie kontekstowe
1	czy jest ssakiem?	Is it a mammal?
2	czy jest lądowy?	Can it jump?
3	czy żyje w Polsce?	Does it have a tail?
4	czy używany w rolnictwie?	Does it eat meat?
5	czy mięsożerny?	Does it have a fur?
6	czy żyje w lesie?	Is it solitary?
7	czy żyje w wodzie?	Can it bite?
8	czy się na niego poluje?	Does it live in China?
9	czy żyje na łące?	Can it swim?
11	czy jest w zoo?	Does it have a night vision?
12	czy jest kotowaty?	Is it predator?
13	czy żyje w afryce?	Is it a type of dog?
14	zgaduje: lew	Is it cat?
15		Is it wild?
16		Does it live in africa?
17		Does it have a mane? : I guess it is lion

Tabela 5.2: Porównanie wyszukiwania człowieka i algorytmu wyszukiwania kontekstowego ze zgadywaniem dla obiektu *lew*

to samo, ale przez to, że reprezentowane są różnymi ciągami znaków traktowane są jako różne elementy pamięci semantycznej. Zaistnienie powyższej sytuacji oraz wystąpienia wulgaryzmów wymagają moderacji wiedzy, jaka napływa do pamięci semantycznej w procesie interakcji z człowiekiem.

Pojawianie się nienaturalnych pytań dla sposobu wyszukiwania człowieka, zaobserwować można również w podejściu stosowanym przez Braingle Zoo, gdzie dopiero po uzyskaniu odpowiedzi pozwalającej zawęzić zbiór najbardziej prawdopodobnych obiektów widoczne jest wyraźniejsze zstępowanie pytań w stronę coraz bardziej szczegółowych cech, jakie może posiadać poszukiwany obiekt. Zauważyć można również, że w obu systemach występują pytania, które raczej nie zostałyby zadane przez człowieka, a pomimo to pozwalają rozdzielać obiekty od siebie.

W przypadku algorytmu pamięci semantycznej widoczne jest wystąpienie po sobie pytań, których człowiek nie zadałby ze względu na dodatkową posiadaną wiedzę, znajdującą się w powiązaniach między cechami np. „Does it eat meat?" i później „Is it predator?". Wiedza taka może zostać jednak wprowadzona do pamięci semantycznej, poprzez dodanie powiązania typu *entail* („predator–entail–eat meat"). Użyteczność ty-

pów powiązań dyskutowana była w rozdziale 5, przy opisie struktury powiązań w pamięci semantycznej (rysunek 5.1). W chwili obecnej relacje określonego typu (który może być proceduralnie interpretowany) pomiędzy cechami wprowadzane są ręcznie. Brak aktywnych dialogów pozwalających na zdobywanie wiedzy o powiązaniach miedzy cechami stanowi waży element dalszego rozwoju pamięci semantycznej. Zaznaczyć tu należy również, że pojawianie się w kolejnych pytaniach treści o wiedzę, która może z siebie wynikać pozwala algorytmowi wyszukiwania uzyskać potwierdzenie, że obiekty znajdujące się w kolejnej podprzestrzeni O(A) są prawidłowe.

Przeprowadzenie doświadczenia, którego celem było porównanie szybkości wyszukiwania kontekstowego z wyszukiwaniem ludzi, poprzez uczenie oparte na aktywnych dialogach zmieniło w następujący sposób strukturę pamięci semantycznej: pierwotnie zapisanych było 197 obiektów, opisanych 529 cechami, o średniej liczbie cech w wektorze $\mathrm{CDV}_\rho = 50{,}64$. Na zakończenie doświadczenia w pamięci semantycznej były 243 obiekty, opisane 602 cechami, o średniej liczbie cech $\mathrm{CDV}_\rho = 48{,}23$. Przeprowadzona końcowa testowa procedura oceny jakości (5.2) dała wartość $Q = 0,8$, nie pogarszając tym samym jakości wiedzy w pamięci semantycznej, a opisując więcej obiektów większą liczbą cech. Zwiększenie liczby cech używanych do opisu obiektów pamięci semantycznej nie spowodowało pogorszenia wyników wyszukiwania, spowodowało jednak zmniejszenie się CDV_ρ, co wskazuje na mniejszy stopień kompletności wiedzy zapisanej w pamięci semantycznej.

W perspektywie rozwoju pamięci semantycznej konieczna jest implementacja pamięci rozpoznawczej [38], pozwalającej odnajdywać słowa, które oznaczają to samo. Istnienie identycznych, a różnie interpretowanych cech i obiektów wynika z faktu, że dla komputera słowo jest ciągiem znaków.

Rozdział 6

Podsumowanie

Przedstawione w pracy wyniki przeprowadzonych eksperymentów wykazały możliwość zastosowania komputerowej realizacji pamięci semantycznej do wyszukiwania opartego na kontekście. Posłużyły one wykazaniu tez postawionych w pracy:

Teza I: „Reprezentacja wiedzy semantycznej stanowi podstawę dla algorytmu wyszukiwania kontekstowego" udowodniona została przez utworzenie algorytmu wyszukiwania kontekstowego na bazie reprezentacji wiedzy vwCRK, która pozwala składować powiązania między pojęciami językowymi. Działanie algorytmu zademonstrowane zostało w obszarze wiedzy medycznej oraz w grze słownej w dziedzinie zwierząt, dzięki której zademonstrowane zostały elementarne możliwości lingwistyczne semantycznej reprezentacji wiedzy.

Teza II: „Algorytm wyszukiwania może zostać użyty do weryfikacji i akwizycji wiedzy semantycznej" udowodniona została przez wyniki przeprowadzonych doświadczeń, z użyciem algorytmu do wyszukiwania kontekstowego, który rozszerzony o tzw. aktywne dialogi wykazał możliwość pozyskiwania wiedzy dla pamięci semantycznej. Ocena tak pozyskiwanej wiedzy zrealizowana została przez utworzoną miarę jakości, opartą na rezultacie wyszukiwania w pamięci semantycznej.

Przeprowadzone doświadczenia na danych medycznych opierają się na założeniu, że wyszukiwanie kontekstowe jest elementem procesu diagnostycznego i posłużyły jedynie demonstracji właściwości algorytmu, a nie realizacji systemu wspomagania decyzji medycznych. Otrzymane rezultaty diagnozowania opartego na wskazywaniu cech charakterystycznych jednostek chorobowych wskazują na lepszą użyteczność algorytmu opartego na wyborze cech, poprzez zysk informacyjny, od przeprowadzania testów kolejnych węzłów drzewa diagnostycznego.

Algorytm wyszukiwania kontekstowego opracowany został na bazie zna-

nej z indukcji drzew decyzyjnych metody liczenia zysku informacyjnego. Posłużył on do implementacji gry w dwadzieścia pytań, która jest demonstracją kompetencji lingwistycznych przyjętego sposobu reprezentacji wiedzy składowanej w pamięci semantycznej. Realizacja wyszukiwania kontekstowego wymagała wprowadzenia rozszerzeń, które nie są oferowane przez typowe algorytmy drzew decyzyjnych: dopuszczenia niezgodności między bazą wiedzy a rezultatami testów pochodzących od użytkownika oraz uczenia w procesie interakcji z człowiekiem. Wyszukiwanie w pamięci semantycznej jest przykładem zadania dla uczenia maszynowego, które w tej formie nie było postawione: zwykle szukamy jednej z kilku klas wielu obiektów, a tu mamy bardzo wiele klas reprezentujących pojedyncze obiekty (przez co można utożsamić klasę i obiekt).

Automatyczne pozyskiwanie wiedzy kontekstowej z zasobów tekstowych, dla pamięci semantycznej, weryfikowanej w procesie wyszukiwania okazało się zadaniem trudnym, które dało tylko częściowo zadowalające rezultaty. Przeprowadzone eksperymenty pokazały dwie podstawowe rzeczy: niedostatki istniejących cyfrowych zasobów semantycznych oraz duże wyzwanie dla przetwarzania języka naturalnego, jakim jest prosta dla człowieka gra w 20 pytań.

Kompensacja braków i korekta wiedzy pochodzącej z zasobów cyfrowych nastąpiła przez realizację aktywnego uczenia pamięci semantycznej. Podejście to obrazuje możliwości pozyskiwania i wykorzystywania wiedzy językowej w systemach komputerowych z użyciem relatywnie prostej reprezentacji wiedzy vwCRK, w stosunku do bardziej wyszukanych metod np.: ram, czy logiki, za pomocą których takie zastosowania nie zostały pokazane. Przyjęta reprezentacja wiedzy w elastyczny sposób umożliwia realizacje różnych typów relacji wiążących pojęcia językowe, nie zawężając się jedynie do powiązań taksonomicznych. Wprowadzona aktualizacja bazy wiedzy, na podstawie interakcji z człowiekiem pokazuje możliwość częściowej implementacji podstawowych aspektów języka: zmienności, zależności znaczenia od kontekstu, rozmycia pojęć i występujących sprzeczności (w rozumieniu logiki), które powodują, że algorytmiczne ujęcie języka naturalnego jest zadaniem wybitnie trudnym. Użyta do realizacji pamięci semantycznej ogólna metoda reprezentacji wiedzy vwCRK nie wiąże danych językowych z jednym zastosowaniem, a pozwala na zastosowanie ich w innych projektach lingwistycznych. Zaproponowana w pracy numeryczna reprezentacja pojęć CDV posłużyła do demonstracji elementarnych kompetencji językowych w maszynach, opartych nie na szablonach imitujących rozumienie, a na algorytmach operujących na kontekstowej organizacji danych. Reprezentacja pojęć w postaci CDV może znaleźć szereg zastosowań przy przetwarzaniu języka naturalnego, np. w wyszukiwarkach internetowych, jednak do użycia jej na szeroką skalę wymaga

utworzenia sieci semantycznej dla całego języka.

Obok opracowania algorytmu wyszukiwania kontekstowego, zastosowanego do akwizycji i oceny jakości wiedzy semantycznej, do osiągnięć pracy zaliczyć należy utworzenie niezależnego agenta funkcjonującego w sieci, potrafiącego pozyskiwać zdroworozsądkową wiedzę w postaci powiązań między pojęciami. Demonstracja aplikacji pamięci semantycznej dla ograniczonej dziedziny stanowi podstawę do dalszych badań i rozwoju projektu w kierunku wielkoskalowej pamięci semantycznej, której realizacja stanowić może podstawę implementacji rozumienia pojęć w systemach komputerowych. Zasadniczym problem jest tu brak trenerów pamięci, od których w znacznym stopniu zależy jakość gromadzonych w niej danych.

Gra w 20 pytań demonstrująca kompetencje lingwistyczne pamięci semantycznej dla ograniczonej dziedziny, może zostać użyta do postawienia szerzej tego problemu: zwycięstwa maszyny w takiej grze językowej, dla nieograniczonego obszaru, co stanowić może etap pośredni na drodze do przejścia przez komputer testu Turinga. Utworzony algorytm wyszukiwania kontekstowego wprowadza również możliwość ilościowej oceny gromadzonych zasobów językowych, co powszechnie oceniane jest jakościowo.

Dalszy rozwój pamięci semantycznej, poprzez implementację nowych aktywnych dialogów i potencjalną możliwość integracji z szablonami AIML [15] prowadzić będzie do odejścia od pierwotnej koncepcji gry w stronę systemu pozwalającego prowadzić dialog w języku naturalnym, opartego na wiedzy systemu informatycznego o języku.

W trakcie pracy nad rozprawą zaimplementowano szereg programów, które potrzebne były do zrealizowania badań. Z najważniejszych wymienić tu należy:

1. Aplikacja automatycznego generowania zapytań do wyszukiwarki Google dla pojęć ze słownika WordNet.

2. Parser projektu Microsoft MindNet, ponieważ dane udostępniane są tylko on-line.

3. Programy do drążenia tekstu w języku naturalnym oparte na gramatycznej analizie zdań, wymagające zastosowania szeregu metod wstępnego przetwarzania języka naturalnego: identyfikacji słów (tokenizacji), sprowadzania słów do formy podstawowej (*stemming*), oznaczana części mowy (*part of speech tagging*), ujednoznaczniania słów i parserów składniowych.

4. Portal pamięci semantycznej, którego implementacja wymagała integracji różnych platform i języków programowania. Z punktu widze-

nia wytwarzania oprogramowania, interaktywna wizualizacja na stronie WWW, współpracująca z serwerem danych kontekstowych używa zaawansowanych rozwiązań technologicznych, których opracowanie wymagało przetestowania szeregu podejść.

5. Programy do pozyskiwania i konwersji danych lingwistycznych do relacyjnej bazy danych z projektów: BNC, WordNet, ConceptNet, Sumo/Milo ontology, Word association, encyklopedia Britannica. Wynikiem ich zastosowania jest utworzenie dużych korpusów językowych, które są cennymi zasobami lingwistycznymi i materiałem do przeprowadzania eksperymentów z wydobywaniem informacji z tekstu.

6. Wizualizacja słownika WordNet zrealizowana z użyciem komponentu do interaktywnego przeglądania grafów.

7. Aplikacja do ekstrakcji wiedzy z definicji słownikowych z użyciem algorytmu Active Search.

8. Aplikacja do generowania zagadek odnośnie obiektów pamięci semantycznej, demonstrująca możliwość innego zastosowania zaproponowanej reprezentacji wiedzy.

9. Aplikacja do zamiany dokumentów na reprezentację HAL i przetwarzające ją algorytmy: PCA i SVD.

10. Wizualizacja[1] ontologii medycznych UMLS (*Unified Medical Language System*)[2].

Rezultaty uzyskane w trakcie pracy opublikowane zostały w następujących artykułach:

1. W. Duch, J. Szymański, T. Sarnatowicz. *Concept description vectors and the 20 question game*; Intelligent Information Processing and Web Mining, Advances in Soft Computing, Springer LNAI, str. 41–50, 2005.

2. J. Szymański, T. Sarnatowicz, W. Duch. *Semantic memory for avatars in cyberspace*; Proceedings of the International Conference on Cyberworlds, IEEE Computer Society, str. 165–171, 2005.

3. J. Szymański. *Bazodanowy system WordNet jako słownik języka angielskiego*; Wydawnictwo Politechniki Gdańskiej KASKBook, str. 155–163, 2006.

[1] http://semanticspaces.eti.pg.gda.pl/ ,VI 2008
[2] http://www.nlm.nih.gov/research/umls/ ,VI 2008

4. J. Szymański. *Pamięć semantycza i Awatar grający w 20 pytań*; Proceedings of the 2nd Polish and International PhD Forum-Conference on Computer Science, str. 1–15 ,2006.

5. J. Szymański, B. Kamiński, O. Tomczak *Wontougo – kooperacyjny edytor WordNetu*; Zeszyty naukowe Wydziału ETI PG, str. 231–240, 2007.

6. J. Szymański, K. Dusza, Ł. Byczkowski *Cooperative Editing Approach for Building Wordnet Database*; Proceedings of the XVI International conference on system science, str. 448–457, 2007.

7. J. Szymański, T. Sarnatowicz, W. Duch *Towards avatars with artificial minds: Role of semantic memory*; Journal of Ubiquitous Computing and Intelligence, American Scientific Publishers (w druku), 2007.

8. J. Szymański, W. Duch *Semantic Memory Knowledge Acquisition Through Active Dialogues*; Proceedings of the 20th International Joint Conference on Neural Networks, IEEE Press, str. 3110-3115, 2007.

9. J. Szymański, W. Duch *Semantic Memory Architecture for Knowledge Acquisition and Management*; Proceedings of the 6 International Conference on Information and Management Sciences, California Polytechnic State University, str. 342-348, 2007

10. J. Szymański, A. Mizgier, M. Szopinski, P. Lubomski *Ujednoznacznianie słów przy użyciu słownika WordNet*; Zeszyty naukowe Wydziału ETI PG, str. 421–427, 2008.

11. J. Szymański. *Portal do kooperacyjnej pracy nad ontologiami dziedzinowymi*; Wydawnictwo Politechniki Gdańskiej KASKBook, (w druku), 2008.

12. P. Majewski, J. Szymański. *Text categorisation with semantic common sense knowledge: first results*; Proceedings of 14th Int. Conference on Neural Information Processing (ICONIP07), Springer Lecture Notes in Computer Science, str. 285-294, 2008

13. W. Duch, J. Szymański. *Knowledge Representation and Acquisition for Large-Scale Semantic Memory*; World Congress of Computational Intelligence, IEEE Press, str. 3117–3124, 2008.

14. W. Duch, J. Szymański. *Semantic Web: Asking the Right Questions*; Proceedings of the 7 International Conference on Information and Management Sciences, California Polytechnic State University, (w druku), 2008

Do nie wymienionych wcześniej osiągnięć związanych z pracą wymienić jeszcze należy:

- Opis projektu pamięci semantycznej ukazał się w publikacji Ministerstwa Nauki „Kalejdoskop nauki" 2007.
- Wyniki pracy w postaci portalu pamięci semantycznej demonstrowane były w programie telewizyjnym „Nauka bez kompleksów", jako część projektu badawczego „*Neurocognitive approach to language*"[3]
- Autor pracy był uczestnikiem projektu badawczego realizowanego z grantu KBN N516 035 31/3499 „Strategie i procedury tworzenia i negocjacji ontologii dziedzinowych".
- Od 07.2006 do 06.2008 (z półroczną przerwą) autor pracy otrzymywał stypendium doktorskie Politechniki Gdańskiej.

Ze względu na niejednoznaczność pojęć i złożoność procesu, jakim jest myślenie abstrakcyjne, przetwarzanie języka naturalnego jest trudne. Formalizacja tego zjawiska, a nawet jedynie jego pewnych, wyizolowanych aspektów w postaci programu jest wyzwaniem dla informatyki. Przedstawione w pracy podejście, oparte na reprezentacji pojęć w postaci wektorów CDV pozwala zademonstrować, w ograniczonej dziedzinie języka, możliwość implementacji w maszynie prostych zdolności lingwistycznych. Przejawiające się w bardzo uproszczonym dialogu, w języku zbliżonym do naturalnego, możliwości maszyn operowania na symbolach semantycznych oparte nie na imitowaniu rozumienia, a na modelu znaczenia są projektem badawczym[4], którego pierwsze rezultaty są obiecujące.

Nabywanie wiedzy o języku wymaga interakcji maszyny z człowiekiem, który jest w tym wypadku jej nauczycielem. Konieczność zaangażowania człowieka w proces nabywania zdolności językowych przez maszynę powoduje, że proces ten jest dość powolny. Próba automatyzacji, poprzez statystyczne drążenie danych tekstowych nie dała oczekiwanych rezultatów prawdopodobnie z faktu, że nawet dla człowieka nie jest możliwe nauczenie się języka obcego tylko, poprzez czytanie.

Zastosowanie podejścia, opartego na wektorowej reprezentacji pojęć CDV do przetwarzania tekstu jest propozycją mogącą znacznie poprawić jakość przetwarzania języka naturalnego, zwłaszcza w porównaniu do podejścia BOW, w którym słowa nie posiadają znaczeń. Wymaga ono jednak utworzenia dobrej sieci semantycznej dla pojęć języka naturalnego.

[3]http://www.is.umk.pl/projects/ncl.html ,VI 2008
[4]http://www.is.umk.pl/projects/ssd.html ,VI 2008

Dodatek A

Architektura pamięci semantycznej

Rozdział poniższy przedstawia przyjęte rozwiązania użyte do implementacji portalu internetowego, stanowiącego interfejs do pamięci semantycznej. Zaprezentowany opis ma na celu przedstawienie funkcjonalności zrealizowanych w portalu pamięci semantycznej oraz przyjętych rozwiązań technologicznych.

Komunikacja z pamięcią semantyczną odbywa się przez strony WWW udostępniające na zewnątrz funkcje, pozwalające operować poprzez interfejs, zbliżony do języka naturalnego na wiedzy składowanej w strukturach danych, opartych na opisanej w paragrafie 3.1 reprezentacji wiedzy vwCRK. Pamięć semantyczna, jako składnica pojęć językowych, powiązanych ze sobą określonym typem relacji zrealizowana została z użyciem relacyjnej bazy danych. System udostępnia szereg różnorodnych interfejsów: zarówno dla użytkownika końcowego, jak i programisty. Umożliwiają one implementację rozszerzeń funkcjonalnych, kooperacyjny dostęp do pamięci semantycznej, równoczesną pracę wielu użytkowników nad semantyką w środowisku rozproszonym oraz na przeprowadzanie dedykowanego przetwarzania danych.

Zadaniem utworzonego systemu jest powstanie niezależnego agenta, mogącego funkcjonować w sieci WWW, potrafiącego gromadzić dane w postaci powiązań między słowami w obrębie wybranego zagadnienia. Portal pamięci semantycznej jest platformą umożliwiającą przeprowadzanie eksperymentów z danymi kontekstowymi, jak i z interfejsami do komunikacji z maszyną cyfrową, w sposób bardziej zbliżony do naturalnego człowiekowi. Portal użyty został również do demonstracji kompetencji lingwistycznych pamięci semantycznej.

A.0.1 Architektura portalu pamięci semantycznej

Implementacja pamięci semantycznej zrealizowana została w oparciu o technologie Microsoft .Net[1]. W skład systemu chodzi bazodanowy serwer MSSQL, serwer aplikacji *IIS – Internet Information Server* oraz aplikacje interfejsów. Składniki te odpowiadają trójwarstwowej architekturze sytemu przedstawionej na rysunku A.1.

Dostęp do bazy danych realizowany jest poprzez *semAPI* – zaimplementowanej jako zbiór bibliotek warstwę zamieniającą relacyjne dane na struktury semantyczne i dostarczającą logicznych funkcjonalności, przeprowadzających fizyczne operacje na danych kontekstowych, składowanych w relacyjnej bazie danych.

Warstwa prezentacji wykorzystuje szereg technologii do udostępnienia funkcjonalności pamięci semantycznej użytkownikowi:

- dostęp poprzez Awatara demonstrującego poprzez grę słowną kompetencje lingwistyczne pamięci semantycznej (rysunek A.3 i A.4),
- interaktywny graf do wizualizacji i edycji bazy wiedzy (rysunek A.2),
- panel administratora z dostępem do bardziej zaawansowanych funkcjonalności z poziomu strony ASP (rysunek A.5).

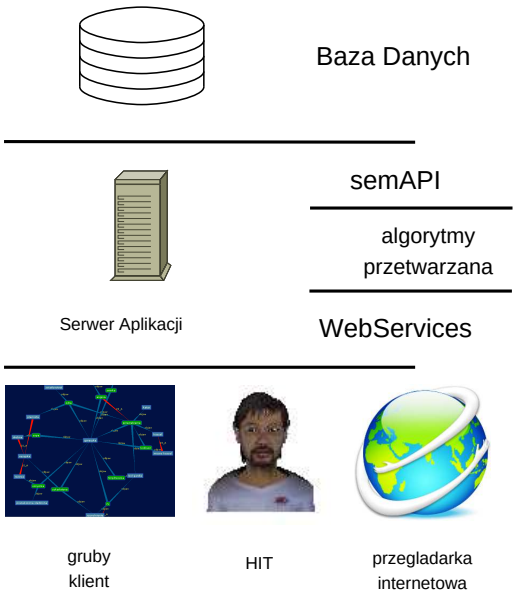
Podejście wydzielające warstwy odpowiedzialne za realizacje poszczególnych zadań, na określonych poziomach systemu pozwala na zachowanie skalowalności rozwiązania, jasne wydzielenie warstw funkcjonalnych, oraz proste dla programisty operowanie na złożonych strukturach danych, a dla użytkownika końcowego intuicyjny sposób interakcji ze złożonym systemem.

Przedstawiona na rysunku A.1 architektura systemu pokazuje podział funkcjonalności, realizowanych odpowiednio w poszczególnych warstwach:

1. Serwer bazy danych – zapewnia kontener dla składowania informacji.
2. Serwer aplikacji – realizuje dostęp do danych z heterogenicznych środowisk, oferuje interfejs WWW oraz WebServces do kooperacyjnego używania pamięci semantycznej.

   W warstwie tej, zaimplementowana została biblioteka SemAPI – zapewniająca jednolity dostęp do danych.
3. Interfejs użytkownika:
   - Gruby klient – zewnętrzne aplikacje komunikujące się z systemem:

[1] http://www.microsoft.com/net/ ,VII 2007



Rysunek A.1: Schemat architektury pamięci semantycznej

interaktywny edytor prezentujący przestrzeń semantyczną w postaci grafu,

aplikacja konsolowa do gry w dwadzieścia pytań i generowania zagadek.

- HIT – ***H****umanised* ***I****n****T****erfaces* – interfejs realizujący komunikację z pamięcią semantyczną, za pomocą interakcji z Awatarem – trójwymiarową głową, posiadającą możliwość syntezy mowy (TTS - text to speach)
- Przeglądarka internetowa – interfejs udostępniający funkcjonalności pamięci semantycznej przez strony WWW.

A.0.2 Baza danych

Jako kontener do składowania danych kontekstowych użyta została relacyjna baza danych. Stanowi ona repozytorium wiedzy pamięci semantycznej. Wybór relacyjnej bazy danych podyktowany został potrzebą wysokiej wydajności koniecznej do przetwarzania i zapisu dużej ilości danych, zwłaszcza powiązań wiele do wiele, które podczas importu danych z zewnętrznych źródeł wiedzy mogą sięgać rzędu milionów rekordów. Użycie relacyjnej bazy

danych zapewnia wydajność przetwarzania dużej ilości informacji, możliwość skalowania repozytorium, jak również umożliwia elastyczne operowanie na danych, które nie jest w prosty sposób osiągalne przy użyciu innych sposobów składowania informacji np.: plików XML, czy zwykłych plików tekstowych.

Jako rozwiązanie najlepiej integrujące się z Microsoft Visual Studio[2] użyta została baza danych tego samego producenta – SQL Server[3].

Relacyjna baza, jako miejsce składowania wiedzy semantycznej zawiera struktury danych, mogące przechowywać dane kontekstowe, opisane w postaci vwCRK. Użycie bibliotek realizujących fizyczny dostęp do składowanych w tych strukturach danych odbywa się, poprzez udostępnienie w logicznych klasach funkcjonalności semantycznych. Zapewnia to rozdzielenie warstwy fizycznej i logicznej aplikacji, co w znaczny sposób ułatwia programowanie.

A.0.3 Warstwa serwera aplikacji i logiki biznesowej

Osadzone na serwerze aplikacji zaimplementowane funkcjonalności podzielone zostały na trzy grupy:

1. API bazy danych – *semAPI* ujednolicające dostęp do bazy danych z poziomu aplikacji

2. Algorytmy realizujące przetwarzanie danych – stanowiące warstwę rozszerzeń funkcjonalnych pamięci semantycznej

3. WebServices – udostępniające określone funkcjonalności pamięci semantycznej na zewnątrz

Biblioteki *semAPI* zapewniają wydajny i elastyczny dla programisty interfejs do operowania na danych w strukturach relacyjnych. Warstwa ta zamienia krotki bazodanowe na obiekty aplikacji, które używane są w programach i poprzez odpowiadające tabelom klasy realizują fizyczny dostęp do danych. Podejście to pozwala na utworzenie logicznych struktur przetwarzania danych ponad strukturami fizycznego składowania. Klasy warstwy semAPI mogą zostać podzielone na dwie grupy:

1. Klasy realizujące fizyczny dostęp do tabel bazy. Wszystkie klasy tego typu zawierają metody *Insert*, *Update*, *Delete*, *SelectEntity* i *SelectCollection* wykonujące określoną przez parametry operację wskazanych, poprzez identyfikator, bądź nazwę danych (na krotce lub na całej kolekcji). Klasy tej grupy używane są do niskopoziomowych operacji na

[2]http://msdn2.microsoft.com/pl-pl/vstudio/default(en-us).aspx ,VII 2007
[3]http://www.microsoft.com/sql/default.mspx ,VII 2007

relacyjnych danych. Operacje realizowane przez klasy dostępowe obejmują: zapis, edycje, uaktualnienie oraz wydobycie wskazanej krotki, bądź określonej kolekcji. Przeprowadzane są za pomocą poleceń SQL zaszytych w strukturze klasy, dzięki czemu operacje na bazie danych są przeźroczyste dla programisty. W wypadku zmiany struktury bazy zapewniona jest integralność jej z *semAPI*, poprzez mechanizmy automatycznego generowania kodu dostępu do tabel. Zapewnia to jednorodność realizowanych funkcji, przez klasy dostępu do danych i jest bardzo wygodne podczas procesu wytwarzania oprogramowania.

2. Logiczne klasy używają opisanych wcześniej bazowych klas dostępu do tabel danych, w celu hermetyzowania funkcjonalności – grupowania w pojedynczej funkcji określonego ciągu operacji przetwarzania danych. Klasy tego typu nie mają swoich odpowiedników w bazie danych, a operacje na nich skutkują aktualizacją wielu tabel.

Z funkcji dostarczanych przez semAPI do operacji na bazie danych korzystają wszystkie zaimplementowane w warstwie rozszerzeń funkcjonalnych algorytmy przetwarzające dane. Oprogramowane są one jako osobne biblioteki realizujące funkcje, będące bardziej złożonymi operacjami na danych kontekstowych, takie jak np. wnioskowanie. Ponieważ dane w bazie wiedzy systemu składowane są w postaci najprostszych faktów otrzymanie dodatkowych informacji, które nie są w niej bezpośrednio zapisane, wymaga utworzenia algorytmu wnioskującego. Zrealizowane w postaci zamkniętej w bibliotece funkcjonalności algorytmy zwracają klientowi już przetworzone dane. Przykładem takich funkcjonalności jest udostępnianie określonych danych w wyspecyfikowanym formacie np.: w formie macierzy wektorów CDV lub obliczenie zysku informacyjnego dla określonej przestrzeni semantycznej.

W celu zachowania możliwości integracji z innymi systemami w warstwie logiki biznesowej istnieje możliwość użycia bibliotek, odpowiedzialnych za wymianę danych z innymi środowiskami. Standardowym sposobem wymiany danych między systemami operującymi semantyką jest format RDF. Wymaga on zamiany informacji zapisanej w bazie danych, w postaci vwCRK do RDF XML. Klasy eksportu i importu danych oferują możliwość napisania własnego konwertera, umożliwiającego wymianę danych praktycznie z każdym systemem zawierającym dane kontekstowe. Zaimplementowane w tym miejscu konwertery pozwalają pozyskiwać wiedzę z takich zasobów, jak: Mnex, MindNet, SumoMiloOntology, WordAssociations czy WordNet, które w celu efektywnego przetwarzania zapisane zostały w relacyjnej bazie danych.

Trzecim zadaniem serwera aplikacji jest udostępnienie na zewnątrz funkcjonalności systemu pamięci semantycznej. Odbywa się ono, poprzez moduł sterowania WebServices oferujący uruchamianie funkcji poprzez protokół

SOAP, bądź przez strony ASP, których sterowanie interfejsem użytkownika posadowione jest w osobnym module warstwy biznesowej.

Moduły sterujące udostępnianiem danych korzystają z funkcji opisanych wcześniej w SemAPI, stanowiących komponenty do zrealizowania bardziej złożonych funkcjonalności. Interfejs dostępu zrealizowany poprzez protokół HTTP zapewnia dostęp z heterogenicznych środowisk, co w pozwala na podłączenie różnorodnych aplikacji do bazy wiedzy. Przykładem jest wizualizacja danych zapisanych w pamięci semantycznej (rysunek A.2). By zapewnić równoległy dostęp do systemu, każdy z wywoływanych modułów składuje zmienne, z których korzysta do obsługi poszczególnych wywołań w obiekcie sesji serwera aplikacji. Może to prowadzić przy dużej liczbie równolegle korzystających z systemu użytkowników do całkowitego wykorzystania pamięci operacyjnej serwera.

A.0.4 Interfejsy

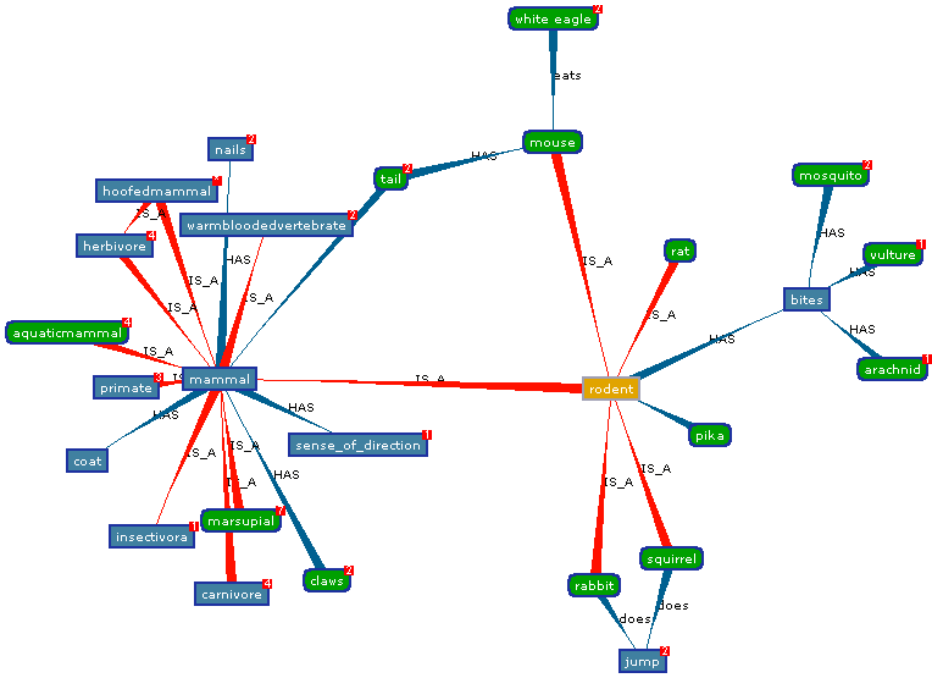
Z punktu widzenia wytwarzania oprogramowania, warstwa prezentacji danych, zapisanych w pamięci semantycznej końcowemu użytkownikowi jest najbardziej złożonym elementem systemu. Wynika to z konieczności integracji wielu różnych platform i języków programowania. Zrealizowane interfejsy umożliwiają różnorodne metody dostępu do pamięci semantycznej. Poniżej przedstawione zostaną poszczególne z nich:

1. Wizualizacja przestrzeni semantycznej przy użyciu interaktywnego grafu.

   Do przeglądania danych składowanych w pamięci semantycznej użyty został interfejs, oparty na interaktywnym grafie. Umożliwia on poruszanie się po sieci powiązań między pojęciami zapisanymi w pamięci semantycznej oraz ich ręczną modyfikację. Interfejs ten, oferuje takie funkcjonalności, jak: dodawanie i usuwanie obiektów i cech oraz definiowanie i zamianę relacji określonego typu.

   Wizualizacja zrealizowana została jako aplikacja grubego klienta osadzona w przeglądarce jako *applet java*. Do wykonania jej użyty został komponent TouchGraph[4] współpracujący z *WebServices* pamięci semantycznej, które dla wskazanego węzła dostarczają informacji o sąsiadujących z nim powiązaniach, pozwalając w ten sposób nawigować nawet po bardzo złożonych grafach. Na rysunku A.2 przedstawiony został przykład wizualizacji fragmentu sieci. Do uruchomienia apletu

[4]http://www.touchgraph.com ,VII 2007



Rysunek A.2: Wizualizacja przestrzeni semantycznej przy użyciu interaktywnego grafu

z panelu administracyjnego konieczne jest zainstalowanie po stronie klienta Java JRE w wersji 1.6[5].

2. HIT - Humanized Interfaces.

   Multimedialny interfejs w postaci trójwymiarowej głowy, zrealizowany został jako obiekt ActiveX[6], osadzony na stronie WWW. Jest on prezentacją możliwości zrealizowania humanoidalnego interfejsu do systemu informatycznego, zawierającego możliwość wizualnej prezentacji, jak i użycia kanału głosowego do komunikacji. Wersja WWW zawiera jedynie możliwość syntezy mowy prostych zdań, generowanych poprzez szablony pytań, z atomów wiedzy vwCRK składowanych w pamięci semantycznej. Werbalizacja dokonana jest przy użyciu syntezatora mowy wbudowanego w komponent ActiveX. Istniejące obecnie rozwiązania technologiczne pozwalają również na rozpoznawanie mowy w procesie komunikacji ze stroną WWW[7]. Technologia ta kładzie duży nacisk

[5]http://java.sun.com/javase/downloads/index.jsp ,VII 2007
[6]http://activex.microsoft.com/activex/activex/ ,VII 2007
[7]http://www.microsoft.com/downloads/details.aspx?FamilyID=23977e00-e369-42a9-b5dc-2f40f6e3f816&DisplayLang=en ,VII 2007

Rysunek A.3: Awatar aplikacji konsolowej pamięci semantycznej

na wymagania sprzętowe, które są niezbędne do zrealizowania takiej funkcjonalności w środowisku internetu, dlatego interfejs taki zrealizowany został jedynie w postaci aplikacji konsolowej, która pozwala na przeprowadzenie aktywnych dialogów z pamięcią semantyczną z użyciem komunikacji, opartej o komendy wydawane głosem. Rozpoznawania mowy zrealizowane zostało w oparciu o komponent Microsoft (SAPI – SpeechAPI)[8].

Awatar w postaci mówiącej głowy zbudowany został przy użyciu komponentów Haptek[9], osadzonych w aplikacji konsolowej lub będącej grubym klientem zagnieżdżonym na stronie WWW i komunikującym się z pamięcią semantyczną, poprzez WebServices. Przykład interfejsu, aplikacji konsolowej, typu HIT został pokazany na rysunku A.3.

3. Interfejs gry słownej poprzez Awatara

   Interfejs gry w dwadzieścia pytań, prezentujący kompetencje językowe pamięci semantycznej przedstawiony został na rysunku A.4. Interfejs demonstruje kompetencje lingwistyczne pamięci semantycznej w zastosowaniu jej do gry słownej w dwadzieścia pytań.

[8]http://www.microsoft.com/speech/speech2007/default.mspx ,VII 2007
[9]http://www.haptek.com ,VII 2007

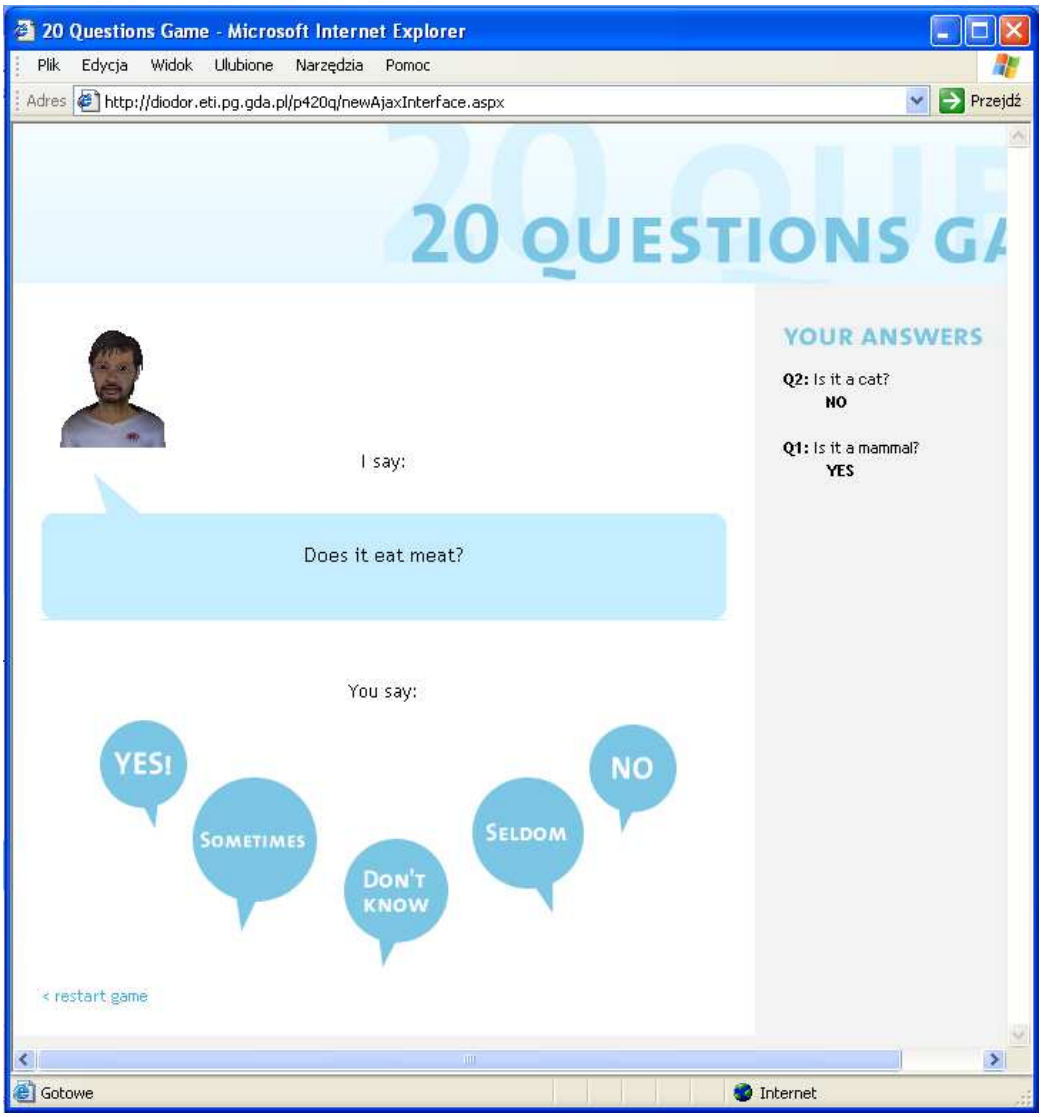
Strona zrealizowana została w technologii Ajax[10]. Zaznaczyć należy, że Awatar jest obiektem ActiveX, którego obsługa realizowana jest jedynie w przeglądarkach Internet Explorer i wymaga doinstalowania po stronie klienta komponentu Haptek. Wersja gry dostępna jest pod adresem http://diodor.eti.pg.gda.pl/p423q/Site/AjaxIf.aspx [11]. Dla innych przeglądarek niż IE uruchamiana jest wersja strony bez Awatara, w technologii Flash[12].

4. Interfejs w technologii ASP

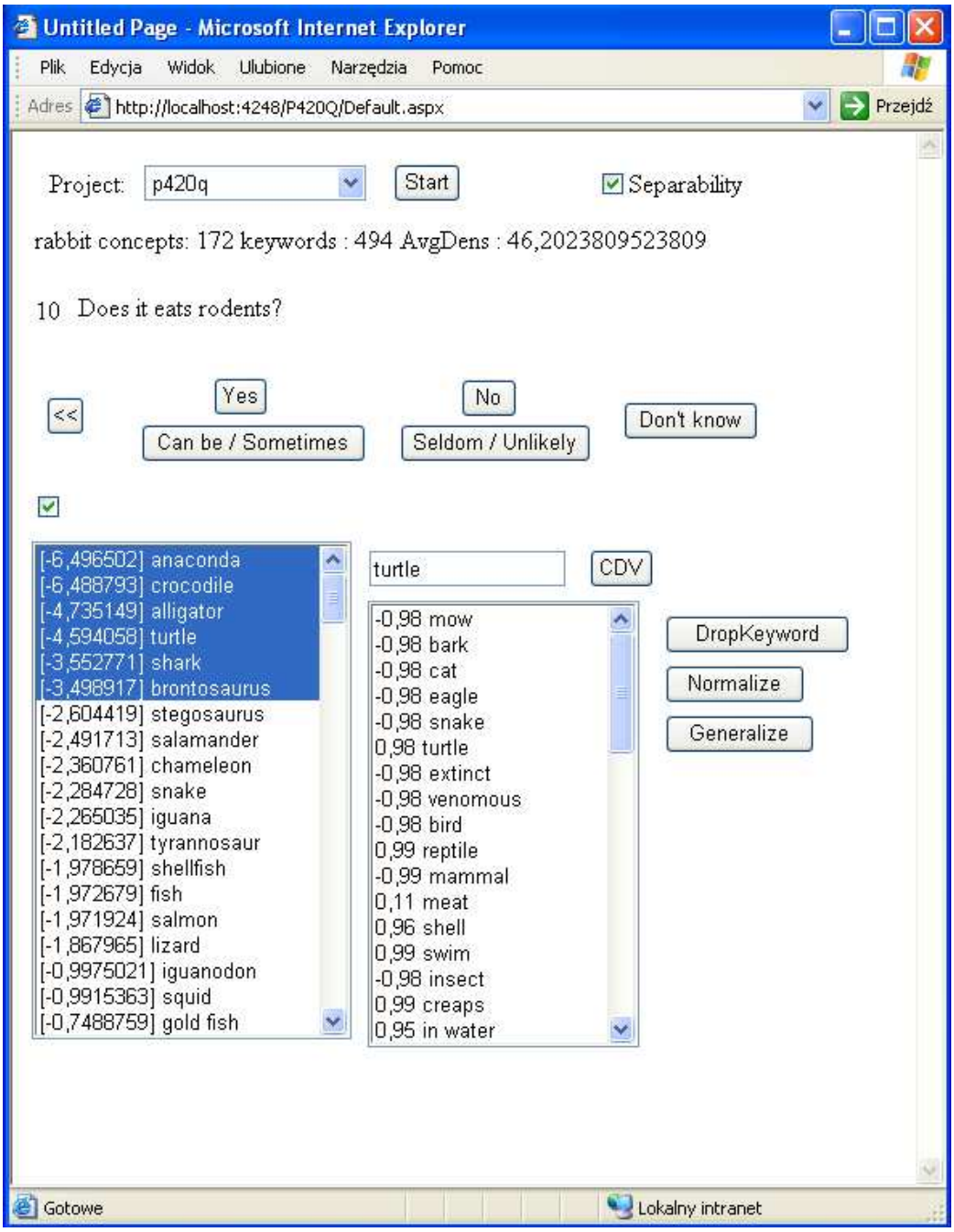
   Interfejs WWW zrealizowany w technologii ASP stanowi panel administracyjny systemu. W miejscu tym zrealizowany jest dostęp do funkcjonalności pamięci semantycznej, niedostępny domyślnym użytkownikom. Realizowane są tu operacje eksportu, importu, wywołania algorytmów semantycznego przetwarzania danych, modyfikacje parametrów opracowanych algorytmów, jak również możliwe jest podejrzenie kolejnych podprzestrzeni O(A). Wygląd panelu administracyjnego przedstawiony został na rysunku A.5. Interfejs ten, pozwala przeprowadzić grę z użyciem uproszczonego w stosunku do HIT interfejsu użytkownika, jak również umożliwia śledzenie procesu wyszukiwania.

   Uproszczona wersja interfejsu gry dostępna jest pod adresem http://diodor.eti.pg.gda.pl/p423q/Site/StandardIf.aspx[13], oraz http://diodor.eti.pg.gda.pl/p423q/Site/Default.aspx[14], będącego panelem administracyjnym portalu. Pod adresem http://diodor.eti.pg.gda.pl/p423q/Site/dsm.aspx[15] udostępniony został interfejs do pamięci semantycznej stanowiący platformę, na której przeprowadzone zostały eksperymenty z danymi medycznymi opisanymi w rozdziale 3.3.

[10] http://asp.net/ajax/ ,VII 2007
[11] VII 2007
[12] http://www.adobe.com/products/flashplayer/ ,VII 2007
[13] VII 2007
[14] VII 2007
[15] VII 2007



Rysunek A.4: Wygląd interfejsu gry słownej pod IE



Rysunek A.5: Panel administracyjny zrealizowany w technologii ASP

Dodatek B

Symbole użyte w pracy

ζ	sieć semantyczna	62
Ψ	przestrzeń semantyczna	62
CDV	Concept Description Vector – wektor opisu obiektu w przestrzeni cech	63
CRK	atom wiedzy semantycznej: obiekt – relacja – cecha	64
w	typowość powiązania obiektu z cechą	64
v	pewność powiązania CRK	64
ANSW	wektor odpowiedzi użytkownika	73
O(A)	podprzestrzeń najbardziej prawdopodobnych obiektów	74
IG	zysk informacyjny cechy w przestrzeni semantycznej Ψ	72
CDV_ρ	średnia liczba cech w wektorach CDV w przestrzeni semantycznej Ψ	109
Q	jakość pamięci semantycznej	135
E	błąd pamięci semantycznej	134
V_s	szybkość wyszukiwania (zbieżności do wyszukiwanego obiektu) – liczba testów cech potrzebnych do odnalezienia obiektu	79, 142
V_n	szybkość uczenia się nowych obiektów	136
NO	wektor CDV nowego obiektu	137
GS	wektor CDV standardu	137
S_d	różnica liczby cech w wektorach CDV	137
S_{GS}	liczba cech wektora NO (nowego obiektu) w wektorze GS (złotego standardu)	138
S_{NO}	liczba cech wektora NO, które nie znalazły się w GS	138
Dif_w	średnia różnica wartości w cech NO występujących GS	138
$Dens(GS)$	liczba cech wektora CDV złotego standardu	137
$Dens(NO)$	liczba cech w wektorze CDV dla nowo tworzonego obiektu	137

Spis rysunków

Bibliografia

[1] J. Allen. *Natural language understanding*. Benjamin-Cummings Publishing Co., Inc. Redwood City, CA, USA, 1995.

[2] J. Anderson. *Language Memory Thought*. John Wiley & Sons Inc, 1976.

[3] J. Anderson. *The architecture of cognition*. Lawrence Erlbaum Associates, 1996.

[4] J. Anderson. *Uczenie się i pamięć. Integracja zagadnień.* WSiP, Warszawa, 1998.

[5] R. Baayen. *Word Frequency Distributions*. Springer, 2002.

[6] M. Blachnik, W. Duch, T. Wieczorek. Selection of prototypes rules–context searching via clustering. *Lecture Notes in Artificial Intelligence*, 4029:573–582, 2006.

[7] P. Blackburn, J. Bos. Representation and Inference for Natural Language: A First Course in Computational Semantics. *Computational Linguistics, MIT Press*, 32(2):283–286, 2006.

[8] L. Bolc, M. Cichy, L. Różańska. Przetwarzanie języka naturalnego. *Wydawnictwo Naukowo-Techniczne, Warszawa*, 1982.

[9] R. Brachman, H. Levesque. *Knowledge Representation and Reasoning*. Morgan Kaufmann, 2004.

[10] L. Breiman. *Classification and Regression Trees*. Chapman & Hall/CRC, 1998.

[11] M. Brzozowska. Etymologia a konotacja wybranych nazw kamieni. *Etnolingwistyka. Problemy języka i kultury*, 12:265–278, 2000.

[12] M. Buckland, F. Gey. The relationship between Recall and Precision. *Journal of the American Society for Information Science*, 45(1):12–19, 1994.

[13] P. Buitelaar. *CoreLex: Systematic Polysemy and Underspecification*. Praca doktorska, Brandeis University, 1998.

[14] D. Burke, D. MacKay, J. Worthley, E. Wade. On the tip of the tongue: What causes word finding failures in young and older adults. *Journal of Memory and Language*, 30(5):542–579, 1991.

[15] N. Bush. Artificial Intelligence Markup Language (AIML) Ver. 1.0.1. *http://www.alicebot.org/TR/2001/WD-aiml*, 2001.

[16] T. Chklovski. *Using Analogy To Acquire Commonsense Knowledge from Human Contributors*. Praca doktorska, Massachusetts Institute of Technology, 2003.

[17] Z. Chlewiński, A. Hankała, M. Jagodzińska, B. Mazurek. *Psychologia pamięci*. Wiedza Powszechna, Warszawa, 1997.

[18] P. Cichosz. *Systemy uczące się*. Wydawnictwa Naukowo-Techniczne, Warszawa, 2000.

[19] A. Collins, E. Loftus. A spreading-activation theory of semantic processing. *Psychological Review*, 82(6):407–428, 1975.

[20] A. Collins, M. Quillian. Retrieval time from semantic memory. *Journal of Verbal Learning and Verbal Behaviour*, 8:240–247, 1969.

[21] G. Curtis. *Business Information Systems*. Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA, 1995.

[22] R. Davis, H. Shrobe, P. Szolovits. What Is a Knowledge Representation? *AI Magazine*, 14(1):17–33, 1993.

[23] S. Deerwester, S. Dumais, G. Furnas, T. Landauer, R. Harshman. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6):391–407, 1990.

[24] DSM. *Diagnostic and Statistical Manual of Mental Disorders*. American Psychiatric Association, 1994.

[25] W. Duch. Modele pamięci i przestrzenie semantyczne. Tekst projektu dla Centrum Doskonałości. *http://www.is.umk.pl/projects/ssda.html*, 2008.

[26] W. Duch, P. Matykiewicz, J. Pestian. Towards Understanding of Natural Language: Neurocognitive Inspirations. *Springer Lecture Notes in Computer Science*, 4669:953–962, 2007.

[27] W. Duch, P. Matykiewicz, J. Pestian. Neurolinguistic approach to natural language processing with applications to medical text analysis. *Neural Networks (in print)*, 2008.

[28] W. Duch, P. Matykiewicz, J. Pestian. Neurolinguistic approach to vector representation of medical concepts. *Proc. of the 20th Int. Joint Conference on Neural Networks*, strony 3110–3115, August 2008.

[29] W. Duch, R. Setiono, J. Zurada. Computational Intelligence Methods for Rule-Based Data Understanding. *Proceedings of the IEEE*, 92(5):771–805, 2004.

[30] W. Duch, J. Szymański. Semantic web: Asking the right questions. *Proceedings of the 7 International Conference on Information and Management Sciences, California Polytechnic State University*, strony 1–8, 2008.

[31] W. Duch, J. Szymański, T. Sarnatowicz. Concept description vectors and the 20 question game. *Intelligent Information Processing and Web Mining, Advances in Soft Computing, Springer Verlag (Eds. Klopotek, M.A., Wierzchon, S.T., Trojanowski, K.)*, strony 41–50, 2005.

[32] S. Dumais, T. Letsche, M. Littman, T. Landauer. Automatic cross-language retrieval using latent semantic indexing. *AAAI Spring Symposuim on Cross-Language Text and Speech Retrieval*, strony 115–132, 1997.

[33] Encyklopedia. *Wielka encyklopedia PWN*. Wydawnictwo Naukowe PWN, Warszawa, 2004.

[34] K. Forster. Category size effects revisited: Frequency and masked priming effects in semantic categorization. *Brain and Language*, 90(1-3):276–286, 2004.

[35] D. Goldberg. *Algorytmy genetyczne i ich zastosowania*. Wydawnictwo Naukowo Techniczne, Warszawa, 1995.

[36] S. Grimm, P. Hitzler, A. Abecker. Knowledge Representation and Ontologies. *Rudi Studer, Stephan Grimm, Andreas Abecker, Semantic Web Services: Concepts, Technology and Applications, Springer, Berlin*, strony 37–81, 2007.

[37] M. Hearst. What is text mining. *http://people.ischool.berkeley.edu/∼hearst/*, 2003.

[38] R. Henson, M. Rugg, T. Shallice, R. Dolan. Confidence in Recognition Memory for Words: Dissociating Right Prefrontal Roles in Episodic Retrieval. *Journal of Cognitive Neuroscience*, 12(6):913–923, 2000.

[39] S. Hunston. Word frequencies in written and spoken english: Based on the british national corpus. *Language Awareness*, 11(2):152–157, 2001.

[40] P. Johnson-Laird. *Mental models: towards a cognitive science of language, inference, and consciousness*. MA: Harvard University Press, Cambridge, 1986.

[41] H. Kamp, U. Reyle. *From Discourse to Logic: Introduction to Model-theoretic Semantics of Natural Language, Formal Logic and Discourse Representation Theory*. Kluwer Academic, 1993.

[42] A. Kao, S. Poteet. *Natural Language Processing and Text Mining*. Springer-Verlag, New York, 2006.

[43] G. Kass. Exploratory technique for investigating large quantities of categorical data. *Applied Statistics*, 29(2):119–127, 1980.

[44] S. Kirkpatrick, C. Gelatt, M. Vecchi. Optimization by Simulated Annealing. *Science*, 220(4598):671–680, 1983.

[45] G. Lakoff. *Women, fire, and dangerous things*. University of Chicago Press Chicago, 1987.

[46] S. Lamb. *Pathways of the Brain: The Neurocognitive Basis of Language*. J. Benjamins, 1999.

[47] A. Langville, C. Meyer. Deeper Inside PageRank. *Internet Mathematics*, 1(3):335–380, 2004.

[48] O. Lassila, R. Swick, i in. Resource Description Framework (RDF) Model and Syntax Specification. *W3C Recommendation*, 1999.

[49] H. Liu, P. Singh. ConceptNet. A Practical Commonsense Reasoning Tool-Kit. *BT Technology Journal*, 22(4):211–226, 2004.

[50] K. Lund, C. Burgess. Producing high-dimensional semantic spaces from lexical co-occurrence. *Behavior Research Methods, Instruments & Computers*, 28(2):203–208, 1996.

[51] P. Majewski, J. Szymański. Text categorisation with semantic common sense knowledge: first results. *Springer Lecture Notes in Computer Science, Proceedings of 14th Int. Conference on Neural Information Processing (ICONIP07)*, 4985:285–294, 2008.

[52] A. Martin, L. Chao. Semantic memory and the brain: structure and processes. *Current Opinion in Neurobiology*, 11(2):194–201, 2001.

[53] T. Maruszewski. *Psychologia poznania*. Gdańskie Wydawnictwo Psychologiczne, Gdańsk, 2001.

[54] P. Matykiewicz, J. Pestian, W. Duch, N. Johnson. Unambiguous Concept Mapping in Radiology Reports: Graphs of Consistent Concepts. *AMIA Annual Symposium Proceedings*, 2006:1024–1031, 2006.

[55] J. McClelland, T. Rogers. The parallel distributed processing approach to semantic cognition. *Nature Reviews Neuroscience*, 4(4):310–322, 2003.

[56] T. McNamara. *Semantic Priming: Perspectives From Memory and Word Recognition*. Psychology Press (UK), 2005.

[57] G. Miller. The magical number seven, plus or minus two: Some limits on our capacity for processing information. *Information.The psychological review*, 63:81, 1956.

[58] G. A. Miller, R. Beckitch, C. Fellbaum, D. Gross, K. Miller. *Introduction to WordNet: An On-line Lexical Database*. Cognitive Science Laboratory, Princeton University Press, 1993.

[59] J. Mulawka. *Systemy ekspertowe*. Wydawnictwo Naukowo-Techniczne, Warszawa, 1996.

[60] S. Murthy, S. Kasif, S. Salzberg. A System for Induction of Oblique Decision Trees. *Journal of Artificial Intelligence*, 2:1–33, 1994.

[61] B. R. Nan, B. G. Jean. *Psycholingwistyka*. Gdańskie Wydawnictwo Psychologiczne, Gdańsk, 2005.

[62] A. Newell, H. Simon. *Human problem solving*. Prentice-Hall, 1972.

[63] I. Niles, A. Pease. Towards a standard upper ontology. *Proceedings of the international conference on Formal Ontology in Information Systems-Volume 2001*, strony 2–9, 2001.

[64] C. Ogden, I. Richards. *The Meaning of Meaning: A Study of the Influence of Language Upon Thought and of the Science of Symbolism*. Kegan Paul, Trench, Trübner, 1930.

[65] A. Paivio. *Mental Representations: A Dual Coding Approach*. Oxford University Press, USA, 1986.

[66] J. Pearl. *Causality: Models, Reasoning, and Inference*. Cambridge University Press, 2000.

[67] M. Posner, S. Keele. On the genesis of abstract ideas. *Journal of Experimental Psychology*, 77(3):353–63, 1968.

[68] D. Prerau. Knowledge acquisition in expert system development. *AI Mag.*, 8(2):43–51, 1987.

[69] G. Qian, S. Sural, Y. Gu, S. Pramanik. Similarity between Euclidean and cosine angle distance for nearest neighbor queries. *Proceedings of the 2004 ACM symposium on Applied computing*, strony 1232–1237, 2004.

[70] Quillian, M.R. The teachable language comprehender: a simulation program and theory of language. *Communications of the ACM*, 12(8):459–476, 1969.

[71] J. Quinlan. Induction of decision trees. *Machine Learning*, 1(1):81–106, 1986.

[72] J. Quinlan. Generating production rules from decision trees. *Proceedings of the Tenth International Joint Conference on Artificial Intelligence*, 30107:304–307, 1987.

[73] R. Richens. Preprogramming for mechanical translation. *Mechanical Translation*, 3(1):25–27, 1956.

[74] E. Rosch. Natural categories. *Cognitive Psychology*, 4(3):328–350, 1973.

[75] D. Ruchkin, J. Grafman, K. Cameron, R. Berndt. Working memory retention systems: A state of activated long-term memory. *Behavioral and Brain Sciences*, 26(06):709–728, 2003.

[76] R. Schank. *Conceptual Information Processing*. Elsevier Science Inc. New York, NY, USA, 1975.

[77] R. Schank. Goal-Based Scenarios: Case-Based Reasoning Meets Learning by Doing. *Case-Based Reasoning: Experiences, Lessons & Future Directions, pages*, strony 295–347, 1996.

[78] R. Schank, R. Abelson. Scripts, plans and knowledge. *Advance papers 4th Intl. Joint Conf. Artificial Intelligence*, strony 151–158, 1975.

[79] W. Scoville, B. Milner. Loss of Recent Memory after Bilateral Hippocampal Lesions. *Cognitive Neuroscience: A Reader*, 2000.

[80] L. Shastri. *Semantic Networks: An Evidential Formalization and Its Connectionist Realization.* Morgan Kaufmann Publishers Inc. San Francisco, CA, USA, 1987.

[81] P. Singh. The public acquisition of commonsense knowledge. *Proceedings of AAAI Spring Symposium: Acquiring (and Using) Linguistic (and World) Knowledge for Information Access*, 2002.

[82] P. Singh, T. Lin, E. Mueller, G. Lim, T. Perkins, W. Zhu. Open Mind Common Sense: Knowledge acquisition from the general public. *Proceedings of the First International Conference on Ontologies, Databases, and Applications of Semantics for Large Scale Information Systems*, 2519, 2002.

[83] E. Smith, E. Shoben, L. Rips. Structure and process in semantic memory: A featural model for semantic decisions. *Psychological Review*, 81(3):214–241, 1974.

[84] J. Sowa. *Principles of Semantic Networks: Explorations in the Representation of Knowledge.* Morgan Kaufmann, Series in Representation and Reasoning, San Mateo, CA, 1991.

[85] G. Sperling. *The information available in brief visual presentations.* American Psychological Association, 1960.

[86] G. Ssali, T. Marwala. Estimation of Missing Data Using Computational Intelligence and Decision Trees. *Arxiv preprint arXiv:0709.1640*, strony 1–14, 2007.

[87] J. Szymański. Bazodanowy system WordNet jako słownik języka angielskiego. *KASKBOOK, Wydawnictwo Politechniki Gdańskiej*, strony 155–163, 2006.

[88] J. Szymański, K. Dusza, Ł. Byczkowski. Cooperative Editing Approach for Building Wordnet Database. *Proceedings of the XVI International conference on system science*, strony 448–457, 2007.

[89] J. Szymański, A. Mizgier, M. Szopiński, L. P. Ujednoznacznianie słów przy użyciu słownika wordnet. *Wydawnictwo Naukowe PG TI 2008*, 18:89–195, 2008.

[90] E. Tulving, G. Bower, W. Donaldson. *Organization of Memory*. Academic Press, New York, 1972.

[91] A. Turing. Computing machinery and intelligence. *Mind*, 59(236):433–460, 1950.

[92] M. Ullman. A neurocognitive perspective on language: the declarative/procedural model. *Nature Reviews Neuroscience*, 2(10):717–726, 2001.

[93] L. Vanderwende, G. Kacmarcik, H. Suzuki, A. Menezes. MindNet: an automatically-created lexical resource. *Proceedings of HLT/EMNLP on Interactive Demonstrations*, strony 8–19, 2005.

[94] J. Vetulani. Pamięć: Podstawy neurobiologiczne i możliwości wspomagania. *Farmakoterapia w Psychiatrii i Neurologii*, 22(1):7–18, 2006.

[95] Z. Vetulani. *Komunikacja człowieka z maszyną. Komputerowe modelowanie kompetencji językowej*. Akademicka Oficyna Wydawnicza Exit, Warszawa, 2004.

[96] P. Voss. Essentials of General Intelligence: The Direct Path to Artificial General Intelligence. *Springer, Artificial General Intelligence*, strony 131–157, 2005.

[97] M. E. Wall, A. Rechtsteiner, L. M. Rocha. *Singular Value Decomposition and Principal Component Analysis*, strony 91–109. Kluwel, Norwell, MA, Mar 2003.

[98] R. Warren. Time and the Spread of Activation in Memory. *Journal of Experimental Psychology: Human Learning and Memory*, 3(4):458–466, 1977.

[99] J. Weizenbaum. ELIZA – a computer program for the study of natural language communication between man and machine. *Communications of the ACM*, 9(1):36–45, 1966.

[100] B. Whorf, J. Carroll. *Language, thought, and reality: selected writings of Benjamin Lee Whorf*. MIT Press, Cambridge, 1956.

[101] P. Winston. *Artificial intelligence*. Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA, 1984.

[102] J. Wolff. Computing, Cognition and Information Compression. *AI Communications*, 6(2):107–127, 1993.

[103] L. Zadeh. Fuzzy logic = computing with words. *IEEE Transactions on Fuzzy Systems*, 4(2):103–111, 1996.