Na prawach rękopisu

POLITECHNIKA WROCŁAWSKA
INSTYTUT INFORMATYKI

Metody analizy spójności i zgodności kolekcji dokumentów WWW

(rozprawa doktorska)

Marek Kopel

Promotor: dr hab. inż. Aleksander Zgrzywa, prof. PWr

Słowa kluczowe: wyszukiwanie informacji,
kolekcje dokumentów WWW,
poprawa relewancji wyników,
grupowanie, semantyka danych

Wrocław 2009

Podziękowania

Pragnę podziękować Promotorowi pracy, Panu prof. Aleksandrowi Zgrzywie za opiekę merytoryczną, a także za wielką życzliwość, cierpliwość i zaufanie. Równocześnie dziękuję Panu prof. Czesławowi Daniłowiczowi, mojemu pierwszemu promotorowi, za stałe motywowanie mnie do pracy.

Pracownikom, doktorantom i studentom Instytutu Informatyki, a zwłaszcza Zakładu Systemów Informacyjnych dziękuję za cenne dyskusje i wsparcie.

Mojej rodzinie dziękuję serdecznie za wytrwałość w wierze we mnie.

Spis treści

Wykaz ważniejszych oznaczeń

A	Zbiór węzłów typu autor
a_j	Węzeł typu autor
Ath	Autorstwo dokumentu
C	Zbiór węzłów typu pojęcie
CC	Współczynnik zgrupowania grafu
$\overline{CC}$	Średni współczynnik zgrupowania grafu
CC_w	Współczynnik zgrupowania grafu ważonego
c_k	Węzeł typu pojęcie
cq	Współczynnik dla klik w grafie
D	Zbiór węzłów typu dokument
DA	Podgraf DAC, powstały przez ograniczenie węzłów tylko do typów D i A
DAC	Graf dokument-autor-pojęcie
DC	Podgraf DAC, powstały przez ograniczenie węzłów tylko do typów D i C
d_i	Węzeł typu dokument
E	Zbiór krawędzi grafu
e_k	Krawędź grafu
F	Dopasowanie pertynencji do grup rankingu
G	Graf
g	Gęstość grafu
$G[W]$	Podgraf grafu G powstały na podstawie podzbioru węzłów W
$P@n$	Dokładność na n pozycjach
$R@n$	Kompletność porządkowa
$R(Ci)$	Ocena grupy C_i w rankingu
R^{agn}	Ocena agenta
R^{rsp}	Odpowiedź agenta n/t oceny
R^{usr}	Ocena użytkownika
T^{agn}	Zaufanie agenta
T^{rsp}	Odpowiedź agenta n/t zaufania
T^{usr}	Zaufanie użytkownika
V	Zbiór węzłów grafu
v_i	Węzeł grafu
w_s	Waga składowa związku

λ_k	Współczynnik istotności zaufania agenta względem zaufania użytkownika
ρ	Współczynnik Spearmana korelacji rankingów
τ	Współczynnik Kendalla korelacji rankingów
τ_k	Współczynnik istotności oceny agenta względem oceny użytkownika
φ	Formuła
Φ	Zbiór formuł

1 Wprowadzenie

Na obecnym etapie rozwoju WWW i – ogólniej – Internetu dostęp do informacji nie jest problemem. Coraz częściej zachodzi potrzeba obrony przed nadmiarem informacji z różnych źródeł. Za pomocą wyszukiwarki internetowej można szybko znaleźć konkretną informację zamieszczoną w WWW, czyli ogólnoświatowej sieci informacyjnej. Problemem natomiast jest ocena jakości tej informacji w kontekście innych pozycji w wynikach zwróconych przez wyszukiwarkę. Coraz częściej użytkownik WWW staje przed zadaniem analizy wyników z wyszukiwarki w celu uzyskania informacji zbiorczej, potwierdzenia wiarygodności jednego źródła informacji lub znalezienia informacji w odpowiedniej formie. W niniejszej pracy zaproponowano metody wspomagające użytkownika WWW w rozwiązaniu takich problemów. Narzędziem tym jest analiza spójności i zgodności.

1.1 Plan rozprawy

Niniejsza rozprawa podzielona została na siedem rozdziałów. Pierwszy rozdział przedstawia semantykę[1] pojęcia „spójność" i „zgodność" w różnych dziedzinach naukowych. Okazuje się bowiem, że te dwa pojęcia zależnie od kontekstu mogą oznaczać zupełnie różne rzeczy. Pierwszy rozdział obejmuje również zdefiniowanie konkretnego celu pracy i tezy, którą praca ma potwierdzić. Rozdział kończy się propozycją dwustopniowego rozwiązania postawionego problemu.

W drugim rozdziale, przytaczając definicje, zbadano semantykę pojęć związanych z WWW. Dało to obraz jak szerokim i pojemnym pojęciem może być „dokument WWW" czy „link". W tym rozdziale pokazano również, na przykładach znanych algorytmów, że najczęściej przyjmowanym modelem sieci WWW jest graf.

Trzeci rozdział przedstawia stan zaawansowania przekształcania się WWW w Sieć Semantyczną. Semantyka jest bowiem kluczowym elementem metod analizy proponowanych w pracy. Dostępność informacji semantycznych pozwala na nowe podejście do analizy tradycyjnie przeprowadzanej metodami statystycznymi.

Kolejnym kluczowym aspektem proponowanych metod analizy jest grupowanie. Czwarty rozdział opisuje aktualne trendy związane z grupowaniem w kontekście WWW. Grupowanie to bazuje przede wszystkim na semantyce i rozwijane jest wraz z coraz większą dostępnością zasobów Sieci Semantycznej. Czwarty rozdział zamyka przegląd stanu badań w dziedzinie, który jest punktem wyjścia dla sformułowanego problemu. Dalszy ciąg rozprawy dotyczy ściśle autorskich badań.

Najważniejszy wkład merytoryczny autora zawarty jest w rozdziale piątym. W rozdziale omówione są wybrane typy informacji semantycznej dostępnej w WWW, oraz wykorzystujące je metody analizy spójności i zgodności. W prezentowanych metodach zaproponowano konkretne algorytmy działające w oparciu o autorski model WWW – graf DAC.

Proponowane metody analizy weryfikowane są na przykładzie autorskiej, analitycznej wyszukiwarki WWW, która w porównaniu do tradycyjnej wyszukiwarki WWW wzbogacona została o funkcjonalności analizy spójności i zgodności. Aplikacja ta, skonstruowana specjalnie na potrzeby weryfikacji, wykorzystuje konkretne typy informacji semantycznej zawartej w systemach blogów w celu poprawy jakości wyszukiwania.

W rozdziale szóstym zaproponowano miary do oceny rerankingu. Reranking jest empiryczną weryfikacją efektywności metody wybranej z prezentowanych metod analizy. Następnie przedyskutowano wyniki eksperymentu badającego jakość rerankingu.

Ostatni rozdział podsumowuje dokonania opisane w pracy i prezentuje perspektywy dalszych badań w dziedzinie analizy spójności i zgodności.

1 Sam termin „semantyka" został tutaj użyty do określenia znaczenia, istoty, która opisywana jest przez pojęcie

1.2 Pojęcia spójności i zgodności w pracach naukowych

Spójność i zgodność to pojęcia często stosowane w nauce. Spójność to zjawisko jednoznaczności i adekwatności informacji. Spójność opisuje zjawisko bycia na wskroś jednakowym. W PWN 2008) przy definicjach „spójność" i „spójny" pojawiają się terminy „logicznie powiązany, harmonijny, konsekwentny". Z kolei „zgodny" tłumaczone jest jako „niesprzeczny z czymś", „jednomyślny, jednakowy, harmonijny". W języku angielskim „spójność" określana jest terminami *consistency*, *cohesion* lub *coherence* czy *coherency*. Wszystkie te terminy mogły by być tłumaczone jako „spójność", jednak używane są w różnych kontekstach. Termin *consistency* używany jest głównie w naukach ścisłych, jak matematyka i fizyka. W informatyce termin używany jest najczęściej w stosunku do integralności danych i modeli spójności pamięci (Mosberger 1993) (Steinke i Nutt 2004). Termin *cohesion* wywodzi się z chemii, a w informatyce stosowany jest w odniesieniu do metodologii programowania modułowego i używany jako miara. Antonimem tego terminu jest *coupling*. Czyli *high cohesion* (wysoka spójność) związana jest z *low coupling* (niska rozdzielność) i odwrotnie (Yourdon i Constantine 1979). Termin *coherence* używany jest w naukach kognitywnych, lingwistyce oraz zamiennie z *consistency* w informatyce, zwłaszcza w odniesieniu do pamięci podręcznej (ang. *cache*) (Handy 1998). Należy również zauważyć, że termin *consistent* może być tłumaczony jako „zgodny".

1.2.1 Spójność w pracach teoretycznych

Podstawowym zastosowaniem pojęcia „spójności" jest logiczne wnioskowanie i dowodzenie oraz matematyczny „dowód spójności" dowodzący, że system logiczny jest „spójny". „Spójność" systemu logicznego oznacza „niesprzeczność" jego formuł (czyli „zgodność").

Definicja 1.2.1. Spójność (ang. *consistency*)

„(a) Φ jest *spójne* (pisane: Con Φ) wtedy i tylko wtedy, gdy nie istnieje formuła φ taka, że $\Phi \vdash \varphi$ i $\Phi \vdash \neg\varphi$.

(b) Φ jest *niespójne* (pisane: Inc Φ) wtedy i tylko wtedy, gdy Φ nie jest spójne (to jest, jeśli istnieje formuła φ taka, że $\Phi \vdash \varphi$ i $\Phi \vdash \neg\varphi$)." - (Ebbinghaus, Flum, i Thomas 1996, 72)

Definicja 1.2.1 mówi, że zbiór formuł Φ jest spójny, jeśli nie można z niego wyprowadzić (oznaczane $\vdash$) sprzecznych formuł.

W statystyce „spójność" jest własnością estymatorów i szacowania. „Spójna sekwencja estymatorów" to taka, która jest probabilistycznie zbieżna do prawdziwej wartości parametru. Często sekwencję taką nazywa się „spójnym estymatorem". Sekwencja jest „silnie spójna" jeśli jest „prawie na pewno" zbieżna do właściwej wartości (Lehmann i Casella 1998).

1.2.2 Spójność w zastosowaniach

Badanie spójności jest kluczowym elementem w przetwarzaniu danych. W bazach danych wymagana jest spójność typu danych oraz niesprzeczności informacji niesionych przez te dane. Spójność jest jednym z warunków transakcji. Transakcje wraz z pozostałymi trzema warunkami są uznawane za kluczowe rozwiązanie stosowane w rozproszonych systemach typu klient-serwer. Transakcje, a więc wszystkie kilku etapowe modyfikacje informacji w systemie, muszą podlegać regule ACID (ang. *Atomicity, Consistency, Isolation and Durability*) (Haerder i Reuter 1983) (Bernstein, Hadzilacos, i Goodman 1987). Reguła ta mówi, że w stanach bazy przed i po modyfikacji zachowane muszą być:

- atomowość (niepodzielność) (ang. *atomicity*) – transakcja nie może być wykonana w części; jeśli w trakcie wykonywania okazuje się, że nie może zostać ukończona w całości, system musi ponownie zostać przywrócony do stanu przed rozpoczęciem wykonywania transakcji;

- spójność (ang. *consistency*) – transakcja musi przekształcać system ze stanu spójnego w inny stan, również spójny;
- izolacja (ang. *isolation*) – modyfikacje w trakcie wykonywania transakcji są niewidoczne dla pozostałych transakcji do momentu jej zakończenia;
- trwałość (ang. *durability*) – jeśli transakcja kończy się pomyślnie modyfikacje wprowadzone przez nią są trwałe, do momentu kolejnych modyfikacji wprowadzonych przez kolejne transakcje.

W przypadku baz danych stan bazy nazywamy spójnym, gdy w tym stanie spełnione są więzy integralności (ang. *integrity constraints*) definiujące zależności między danymi (np. data urodzin pracownika nie może być późniejsza od daty jego zatrudnienia). Autorzy (Moerkotte i Lockemann 1991) twierdzą, że baza danych jest nazywana spójną, gdy jest prawdziwym odzwierciedleniem zadanego miniświata. Jak zauważono w Hanna Mazur i Zygmunt Mazur 2004, 63): transakcja powinna przekształcać bazę ze stanu spójnego w spójny, ale w trakcje transakcji stan bazy może być chwilowo niespójny. W bazach danych o równoległym dostępie powoduje to potrzebę blokowania dostępu do danych przetwarzanych w transakcji. Ponieważ w trakcie transakcji stan bazy może być niespójny, dane, których dotyczy transakcja, nie powinny być dostępne do zakończenia transakcji. Ten fakt generuje nowe problemy z dostępem współbieżnym. Zamiast blokowania dostępu do danych, w Zygmunt Mazur 2006), zaproponowano metody optymistyczne do zarządzania transakcjami. Jedną z metod jest zastosowanie sag. W kontekście ACID, sagi są transakcjami, w których nie zawsze wszystkie cztery własności są ściśle zachowane. Spójność jest jedyną z tych własności, która w sagach jest zachowywana bez uproszczeń. Pozostałe proponowane metody zarządzania transakcjami, czyli: metoda znaczników czasowych i algorytm wielowersyjny, nie gwarantują zachowania spójności. Poprzez transakcje kompensacyjne sagi rozwiązują również problem z wycofywaniem transakcji.

W złożonych bazach danych nie ma możliwości ograniczenia się do sprawdzania spójności jedynie podczas transakcji, jak to zostało zaproponowane w Eswaran et al. 1976). Ponieważ przy dużej liczbie transakcji nie jest optymalnym rozwiązaniem cofanie się do poprzedniego stanu, gdy nowy stan nie zachowuje spójności. Istnieje potrzeba sprawdzania spójności w całym systemie poprzez zachowanie więzów integralności. W Teniente i Olivé 1995) przedstawiono alternatywne, do sprawdzania więzów integralności, rozwiązanie, w odniesieniu do bazy wiedzy, polegające na „naprawianiu" spójności poprzez dodatkowe modyfikacje w systemie (ang. *integrity constraints maintenance*). Jeszcze innym rozwiązaniem zachowania spójności w systemie jest uaktualnianie pochodnych faktów (ang. *updating derived facts*) znane również jako uaktualnianie widoku. Metoda ta polega na znalezieniu przekształcenia żądania uaktualnienia widoku w odpowiednią modyfikację przechowywanych w systemie faktów. Ostatecznie autorzy prezentują metodę, opartą na zdarzeniach i regułach wnioskowania, będącą złożeniem metod utrzymania warunków spójności i uaktualniania widoku. „Naprawianie" spójności w systemie tą metodą polega na uaktualnianiu: faktów w systemie, reguł dedukcyjnych, warunków integralności i widoków. W Decker et al. 1991) zaprezentowano algorytm oparty na procedurze zachowania spójności w bazach dedukcyjnych w odniesieniu do planowania liniowego. W ten sposób cel zapisany jako ograniczona formuła może zostać osiągnięty unikając problemów powstających w tradycyjnym podejściu do planowania. Metoda polega na uzupełnieniu dedukcyjnej bazy danych o mechanizm naprawiający spójność bazy poprzez tworzenie zmian, którym musi zostać poddana baza, aby zachować spójność. Zmiany te służą dwóm celom. Ponieważ opisują one możliwe światy, w których może być zawarty cel, mogą zostać użyte do rozważania tych światów i do odrzucania światów o niepożądanych atrybutach. Dodatkowo zmiany mogą być używane, do kierowania aktualnej procedury planowania tak, aby generowany plan prowadził do wybranych, możliwych stanów świata. Jeszcze innym sposobem przywracania spójności replikowanych danych w rozproszonych bazach danych są metody consensusu. W pracach (Daniłowicz i Nguyen 2000) (Daniłowicz i Nguyen 2003) przedstawiono metody znajdowania consensusu przywracające spójność w rozproszonych

systemach informacyjnych.

Badanie spójności danych jest również problemem w obiektowych bazach danych, gdzie istotne jest zachowanie spójność wersji bazy danych. W Cellary i Jomier 1990) przedstawiono metodę zachowania spójności wersji w obiektowej bazie danych. Wielowersyjna baza danych jest tutaj rozpatrywana jako zbiór logicznie niezależnych i identyfikowalnych wersji bazy. Autorzy rozwiązują problem niespójności wersji poprzez zastosowanie semantyki znaczków czasowych identyfikujących obiekty.

Modele spójności. Bardziej ogólnym aspektem spójności baz danych jest spójność dzielonych pamięci rozproszonych. Na tym poziomie abstrakcji badanie spójności ma wymiar bardziej teoretyczny, pomimo że wywodzi się z tak praktycznych problemów, jak optymalizacja wydajności systemu. *DSM* (ang. *Distributed Shared Memories*) to pamięci zbudowane w oparciu o systemy rozproszone. Na przełomie ósmej i dziewiątej dekady zeszłego wieku powstało wiele modeli spójności dla pamięci dzielonych (Steinke i Nutt 2004). Najważniejsze to: spójność sekwencyjna, spójność PRAM, spójność pamięci podręczej i spójność procesora. Wraz z rozwojem DSM ewoluowały modele spójności. W Gharachorloo et al. 1990) zdefiniowany został model spójności *release consistency*, ekwiwalentny dla modelu spójności sekwencyjnej dla programów równoległych z wystarczającą synchronizacją. W Mosberger 1993) wprowadzone zostały inne modele spójności, między innymi: *atomic* i *casual*. W Raynal i Schiper 1996) zaprezentowano zbiór formalnych definicji spójności opartych na protokołach.

W (Kumar 1992) techniki zachowania spójności zostały wykorzystane do poprawy wydajności rozpoznawania obrazów. Techniki spójności mają na celu rozwiązanie problemu zaspokojenia więzów integralności, w skrócie CSP (ang. *constraint satisfaction problem*). Zachowanie spójności polega na ograniczaniu domen poszukiwanego rozwiązania. Należy tu zauważyć, że o ile przeszukiwanie (ang. *search*) jest niedeterministyczne, o tyle techniki *spójności* są deterministyczne. Jednakże techniki spójności rzadko są stosowane jako jedyna metoda rozwiązywanie CSP.

CSP często dotyczy spójności lokalnej . W Marriott i Stuckey 1998) omówiono podstawowe warunki spójności. Są to warunki spójności:

- węzła (ang. *node consistency*), z zachowaniem warunku dla jednej zmiennej;
- łuku (ang. *arc consistency*), z więzami binarnymi (dwóch zmiennych);
- ścieżki (ang. *path consistency*), gdy więzy binarne zachowują więcej niż dwie zmienne (każda z każdą).

Spójność lokalna może zostać utrzymana poprzez transformacje problemu zwane propagacją więzów (ang. *constraint propagation*). Istnienie wiele klas spójności lokalnej, na przykład:

- pierwotna spójność lokalna, której warunki zakładają, że każde zadanie może zostać spójnie rozszerzone na kolejną zmienną;
- spójność kierunkowa (ang. *directional consistency*), która zakłada spełnienie warunku, jeżeli kolejna zmienna przy rozszerzaniu będzie większa od tych w zadaniu zachowując określony porządek;
- spójność relacyjna (ang. *relational consistency*), która zakłada rozszerzanie zadania o wiele zmiennych, ale przy zachowaniu tylko pewnego podzbioru więzów.\

W aspekcie rozpoznawania obrazów, w Leclerc, Luong, i Fua 2000), autorzy zdefiniowali pojęcie samo-spójności (ang. *self-consistency*). Samo-spójność to własność systemu wizyjnego, która jest spełniona, gdy wnioski percepcyjne jednego punktu widzenia są takie same jak wnioski percepcyjne z innego punktu widzenia.

Kolejnym przykładem zastosowania spójności są systemy multi-agentowe, w których głównym problemem jest uzgadnianie wiedzy agentów. W pracy (Daniłowicz, Nguyen, i Jankowski 2002) przedstawione zostały definicje miar spójności wiedzy agentów. Podstawową metodą na osiąganie spójności wiedzy agentów w systemach rozproszonych jest konsensus (Nguyen 2002).

Jeszcze inną dziedziną, w której pojawia się problem spójności są interfejsy użytkownika.

W Mahajan i Shneiderman 1995) zaprezentowano metryki, które ułatwiają wykrywanie anomalii w kolorach, wielkości, kroju i rodzaju czcionek oraz w rozłożeniu przycisków. Jednak, jak pokazuje (Grudin 1989), w przypadku interfejsów użytkownika kwestia spójności może być bardzo niejednoznaczna.

W (Kopel 2004) spójność została wykorzystana jako kryterium dla filtrów antyspamowych.

1.2.3 Spójność w odniesieniu do WWW

Spójność w odniesieniu do WWW ma szczególne zastosowanie w utrzymywaniu sieci pamięci podręcznych w systemach buforujących dokumenty WWW. W Cao 1998) przedstawiono metodę utrzymywania **słabej spójności** pamięci podręcznej: *adaptive TTL*. Metoda TTL (ang. *Time to Live*) polega na przypisywaniu każdemu dokumentowi znacznika jak długo treść tego dokumentu jest ważna. Jeśli podczas dostępu do dokumentu w pamięci podręcznej okazuje się, że czas ważności dokumentu minął pobierana jest jego aktualna wersja ze źródła. Dla porównania pokazano, że **mocna spójność** w metodach *polling every time* i *invalidation* może zostać zachowana bez lub z niewielkim udziałem dodatkowych kosztów. Metoda *polling every time* polega na sprawdzaniu, przy każdym dostępie do dokumentu w cachu, czy dokument źródłowy nie zmienił się od daty ostatniego buforowania. W metodzie *invalidation* serwer źródłowy przechowuje listę serwerów buforujących dany dokument i w momencie modyfikacji tego dokumentu informacja o tym fakcie jest rozsyłana do serwerów buforujących.

Problem spójności dokumentów w sieci WWW rozważa również (Kermarrec i Soleto 1997). Autorzy porównują spójność dokumentów ze spójnością pamięci dzielonych i podają wskazówki zarządzania dokumentami sieciowymi, aby pogodzić 2 rzeczy: lokalny dostęp do dokumentu i spójność dokumentu pierwotnego z lokalnymi kopiami w czasie.

Podobnym problemem zajęto się w Santos, Sampaio, i Courtiat 1999). W pracy przedstawiono metodę usuwania tymczasowych niespójności w odniesieniu do dokumentów hipertekstowych. Niespójności powodować mogą wewnętrzne lub zewnętrzne niedeterministyczne zdarzenia. Wewnętrzne niedeterministyczne zdarzenia dotyczą elastyczności czasu prezentacji mediów. Powiązane to jest między innymi z dopuszczalną jakością usług, w skrócie QoS (ang. *Quality of Service*). Zewnętrzne niedeterministyczne zdarzenia powiązane są ze zdarzeniami, takimi jak: interakcja użytkownika, opóźnienia sieciowe i przetwarzaniem wyników z zapytań do baz danych, naukowych symulacji itp. Autorzy identyfikują tymczasową niespójność spowodowaną wewnętrznym niedeterminizmem, zewnętrznym niedeterminizmem oraz oboma na raz.

1.2.4 Zgodność w pracach teoretycznych

Zgodność dwóch elementów w informatyce rozumiane jest jako harmonijne istnienie, dopasowanie czy niesprzeczność tych elementów. W odniesieniu do sprzętu komputerowego bardzo modny swego czasu był termin **kompatybilność**, który również miał oznaczać zgodność – w tym przypadku – danego podzespołu z innym. Jedna z definicji kompatybilności w WordNet brzmi: „zdolny do bycia używanym lub podłączonym do drugiego urządzenia lub komponentu bez modyfikacji” (George A. Miller 2006). Kompatybilność można też rozumieć jako zgodność jednego urządzenia ze specyfikacją współdziałania drugiego urządzenia. Można więc powiedzieć, że kompatybilność to przestrzeganie norm podczas interakcji między dwoma urządzeniami.

1.2.5 Zgodność w odniesieniu do WWW

Jeśli chodzi o pojęcie zgodności w odniesieniu do WWW, to głównie używa się go w znaczeniu stosowania się do wymagań i wytycznych. Wymagania i wytyczne, to głównie specyfikacje standardów. Najczęściej dyskutowane i najbardziej krytyczne problemy zgodności w tematyce

WWW dotyczą raportów technicznych W3C[2] i IETF[3].

Definicja 1.2.2. Zgodność (ang. *conformance*)

> „Wypełnienie przez produkt, proces, system lub usługę wyszczególnionego zbioru wymagań.” - (Rosenthal et al. 2005)

Definicja 1.2.3. Ścisła zgodność (ang. *strict conformance*)

> „Zgodność implementacji, która obejmuje jedynie wymagania i/lub funkcjonalność zdefiniowaną w specyfikacji i żadnych więcej (t.j. nie są implementowane żadne dodatkowe rozszerzenia specyfikacji).” - (Rosenthal et al. 2005)

Definicje 1.2.2 i 1.2.3 pochodzą z rekomendacji W3C standaryzującej wytyczne do pisania specyfikacji. Przytoczone pojęcia zgodności dotyczą przestrzegania i wypełniania wymogów specyfikacji, które stają się rekomendacjami W3C. Taka zgodność dotyczy więc najczęściej trzymania się wytycznych dla formatu dokumentu danego typu, składni protokołu czy języka zapytań.

Walidacja. Proces sprawdzania zgodności badanego obiektu ze specyfikacją nazywa się walidacją.

Definicja 1.2.4. Walidacja, walidować, walidowanie (ang. *validation, validate, validating*)

> „Proces niezbędny do przeprowadzenia testów **zgodności** według predefiniowanych procedur i oficjalnych zbiorów testowych.” - (Dubost i Skall 2005)

Definicja 1.2.5. Walidacja

> „Walidacja to proces, w którym dokumenty są weryfikowane względem dołączonego DTD[4], zapewniając, że struktura, użycie elementów i użycie atrybutów jest **spójne** z definicjami w DTD. ” - (Steven Pemberton 2002)

Definicja 1.2.4 wiąże się z programem zapewniania jakości W3C. Jeśli dany obiekt przechodzi pomyślnie walidację, wtedy inne dokumenty i usługi bazujące na tym obiekcie mają zagwarantowaną bezproblemową interakcję. Dzięki temu można pominąć implementacje wyjątków związanych z sytuacjami nieprzewidzianymi przez specyfikację, z którą zgodny jest rzeczony obiekt. Termin walidacji staniał od początku prób standaryzowania WWW, jednak jego pojęcie traktowane było intuicyjnie. Nieco starsza definicja 1.2.5, dotyczy uszczegółowienia pojęcia walidacji na potrzeby standardu XHTML. Co ciekawe, przy porównaniu obu definicji widać jak luźno w środowisku zajmującym się WWW interpretowane są pojęcia zgodności i spójności. Definicja 1.2.4 mówi o walidacji, jako procesie badania zgodności, natomiast 1.2.5 przedstawia walidację jako proces badania wymaganej spójności.

Najpopularniejszymi narzędziami testującymi zgodność dokumentów z rekomendacjami W3C są oficjalne walidatory dla (X)HTML i CSS. Webmasterzy dbający o zgodność swoich dokumentów z normami najczęściej umieszczają informację o pozytywnym przejściu walidacji w stopce danego

2 W3C (ang. *World Wide Web Consortium*) – założona przez Tima Bernersa-Lee międzynarodowa organizacja zajmująca się rozwijaniem standardów dla WWW. Forma konsorcjum oznacza, że firmy związane z WWW mają w W3C pełnoetatowych przedstawicieli, którzy wspólnie opracowują standardy. W3C zostało założone w 1994 roku. W 2008 roku liczyło 434 członków.

3 IETF (ang. *Internet Engineering Task Force*) - to nieformalne, międzynarodowe stowarzyszenie osób zainteresowanych ustanawianiem standardów technicznych i organizacyjnych w Internecie. IETF ściśle współpracuje z W3C.

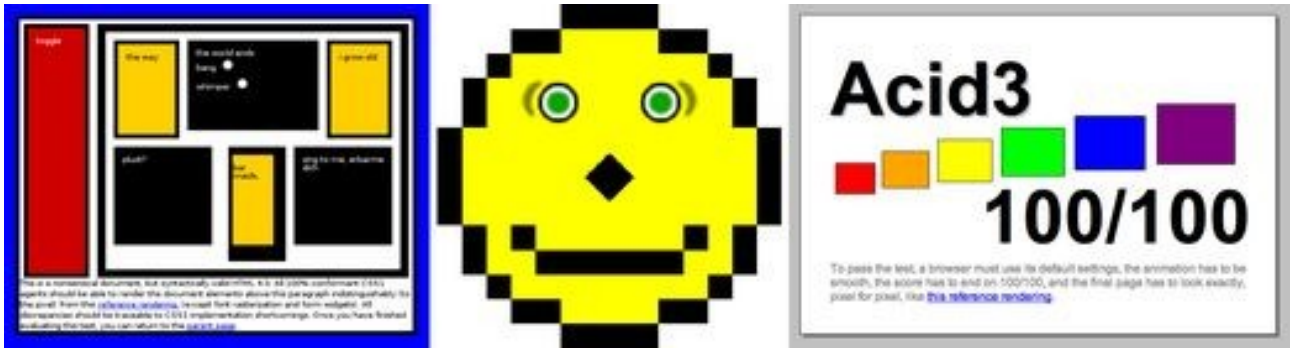
4 DTD (ang. *Document Type Definition*), czyli definicja typu dokumentu, to język pozwalający określić formalną strukturę dokumentów tworzonych według konkretnych języków znaczników, takich jak HTML czy XML. Termin DTD może również oznaczać sam dokument lub fragment dokumentu, który opisuje tą strukturę w języku DTD.

dokumentu. Taka informacja, w postaci oficjalnego logo walidacji W3C jest jednocześnie linkiem do walidatora, aby każdy odwiedzający stronę sam mógł sprawdzić jej walidację. Nie jest to jednak częste, ponieważ przeciętny użytkownik WWW najczęściej nie wiem czym są rekomendacje W3C i dlaczego należy się do nich stosować.

Trzymanie się jednolitych standardów podczas tworzenia WWW, czyli udostępniania nowych serwisów, tworzenia blogów czy aplikacji używających HTTP jako warstwy komunikacyjnej pozwala nie tylko na prostsze łączenie i współpracę tych elementów WWW. Sam problem uzgadniania formy dokumentów WWW stał się krytyczny w momencie powstania konkurencji na rynku przeglądarek internetowych. Ponieważ producenci przeglądarek rozszerzali możliwości wyświetlania dokumentów w sposób niestandardowy, a webmasterzy chętnie wykorzystywali nowe możliwości, często zdarzało się tak, że strona napisana pod jedną przeglądarkę, nie dała się zupełnie wyświetlić w drugiej. Działo się to nie tylko przez wykorzystanie niestandardowych funkcjonalności jednej przeglądarki, ale również dlatego, że same przeglądarki często nie były zgodne lub nie w pełni implementowały rekomendacje W3C.

Przez długi czas, webmasterzy dbający o dostępność swojej strony z poziomu każdej przeglądarki musieli testować i najczęściej pisać alternatywne wersje swoich dokumentów dla różnych przeglądarek. Oczywistym rozwiązaniem tego problemu było ujednolicenie sposobu obsługi dokumentów WWW przez przeglądarki, co W3C starało się promować od początku swojego istnienia. Ponieważ jednak weryfikowalność zgodności przeglądarek z W3C była domeną informatyków zajmujących się WWW, a niektórzy producenci przeglądarek byli głusi na problemy niezgodności, powstały testy umożliwiające każdemu użytkownikowi proste sprawdzenie zgodności jego przeglądarki ze standardami.

Testy Acid. Podobnie jak w przypadku walidatorów testujących zgodność dokumentów z rekomendacjami W3C, tak testy Acid pozwalają pod tym kątem sprawdzić zgodność agenta użytkownika, czyli najczęściej przeglądarki. Zbieżność nazw testów Acid i reguły transakcji ACID w bazach danych jest przypadkowa. W tym przypadku Acid nie jest skrótem, ale nazwą zapożyczoną od testów kwasowych używanych w połowie XIX wieku przez poszukiwaczy złota. Testy Acid polegają na zadaniu przeglądarce dokumentu WWW wykorzystującego testowany zakres funkcjonalności i porównaniu wyrenderowanego[5] (wyświetlanego) dokumentu ze wzorcem w postaci obrazka, jak na rys. 1.1.



Rysunek 1.1. Poprawnie wygenerowane elementy (strony) w testach (od lewej) Acid, Acid2, Acid3

źródło: (Hickson 2009)

Do tej pory powstały 3 testy przygotowane w ramach The Web Standards Project przez Iana 'Hixie' Hicksona, obecnie pracownika Google i członka Web Hypertext Application Technology

5 Renderowanie – wizualizacja informacji zawartych w dokumencie elektronicznym dokonywana w formie właściwej dla wskazanego środowiska, w tym przypadku przeglądarki internetowej

Working Group zajmującej się rozwojem HTML 5. Pierwszy z testów powstał w październiku 1998 roku i weryfikował zgodność przeglądarek ze specyfikacją CSS1 według (Håkon Wium Lie i Bos 2008). Opublikowany w kwietniu 2005 roku test Acid2 był o wiele szerszym testem zgodności. Testował nie tylko kolejną wersję arkuszy stylów: CSS2, obecnie wypieraną przez CSS 2.1 (Bos et al. 2007), ale również użycie znaczników HTML, obrazów PNG[6] oraz identyfikatorów data URI[7] [8]. W marcu 2008, po roku opracowywania, światło dzienne ujrzał test Acid3. Test ten skupia się na standardach wykorzystywanych głównie we współczesnych, wysoce interaktywnych, serwisach Web 2.0. W głównej mierze testowana zgodność dotyczy ECMAScript'u, czyli JavaScript'u ustandaryzowego według (ECMA International 1999) i DOM Level 2 według sześciu rekomendacji W3C. Poza tym Acid3 testuje protokół HTTP 1.1 (R Fielding et al. 1999), języki znaczników: HTML 4.0 i 4.01 (Raggett, Le Hors, i I. Jacobs 1999), XHTML 1.0 (Steven Pemberton 2002), SMIL 2.1 (Mullender et al. 2005), grafikę wektorową SVG[9], kodowanie Unicode 5: UTF-8 i UTF-16 oraz, ponownie, data URI. W teście uwzględnione zostały również specyfikacje raportów technicznych CSS3 w wersjach szkiców roboczych (ang. *working draft*), które nie zdążyły jeszcze zostać rekomendacjami.

Pozytywny wynik w testach Acid ma na celu pokazanie użytkownikom, że przeglądarka stosuje się do zaleceń standardów i jeżeli twórca strony czy usługi również stosował się do tych zaleceń, to użytkownik może być pewien, że widzi stronę i korzysta z usługi zgodnie z intencją autora. Czyli testy mają gwarantować, że narzędzia użytkownika dają mu nieprzekłamany dostęp do treści WWW. Aby jednak odbiór treści WWW był właściwy, to sama treść też musi był poprawnie (standardowo) sformatowana. To sformatowanie testują wspomniane wcześniej walidatory. Zarówno te oficjalne dostępne na stronach W3C, jak i te wbudowane w narzędzia developerskie, pozwalające śledzić zgodność ze standardami w czasie tworzenia treści dla WWW. Niestety jak wykazały badania przeprowadzone przez firmę Opera, według (Brian Wilson 2008a) w styczniu 2008 tylko 4.13% z 3,509,180 weryfikowanych URLi[10] przeszło walidację, co i tak jest dwukrotną poprawą w stosunku do dwóch wcześniejszych lat. Użyta próbka może wydawać się mała, gdy Google ogłosił w Alpert i Hajaj 2008), że w lipcu 2008 zaindeksował trylion (10^{12}) różnych adresów URL, jednak autorzy potwierdzają, że jest ona reprezentatywna.

WAI. Dodatkowym aspektem dodającym wartość walidacji treści WWW jest dostosowywanie formy tych treści dla potrzeb niepełnosprawnych. W W3C powstała inicjatywa WAI (ang. *Web Accessibility Initiative*), która ma na celu promowanie dobrych praktyk i standaryzowaniem technik pozwalających na dostęp do treści WWW jak najszerszej grupie odbiorców. Podstawowe wytyczne, standaryzowane przez (Chisholm, Ian Jacobs, i Vanderheiden 1999), dotyczą zarówno autorów treści WWW jak i autorów narzędzi do tworzenia tych treści. Całą inicjatywę WAI tworzą grupy robocze skupione wokół konkretnych zagadnień dostępności. Trzy najważniejsze to:

- WCAG (ang. *Web Content Accessibility Guidelines*) - wytyczne dotyczące dostępności treści internetowych
- ATAG (ang. *Authoring Tool Accessibility Guidelines*) – wytyczne dotyczące oprogramowania służącego do tworzenia stron internetowych
- UAAG (ang. *User Agent Accessibility Guidelines*) – wytyczne dotyczące przeglądarek internetowych

6 PNG (ang. *Portable Network Graphics*) – rastrowy format plików graficznych używający bezstratnej kompresji danych - (Duce 2003)

7 URI (ang. *Uniform Resource Identifier*) to łańcuch znaków umożliwiający nazwanie i identyfikację zasobów w sieci Internet - (L. Masinter, T. Berners-Lee, i R. Fielding 1998)

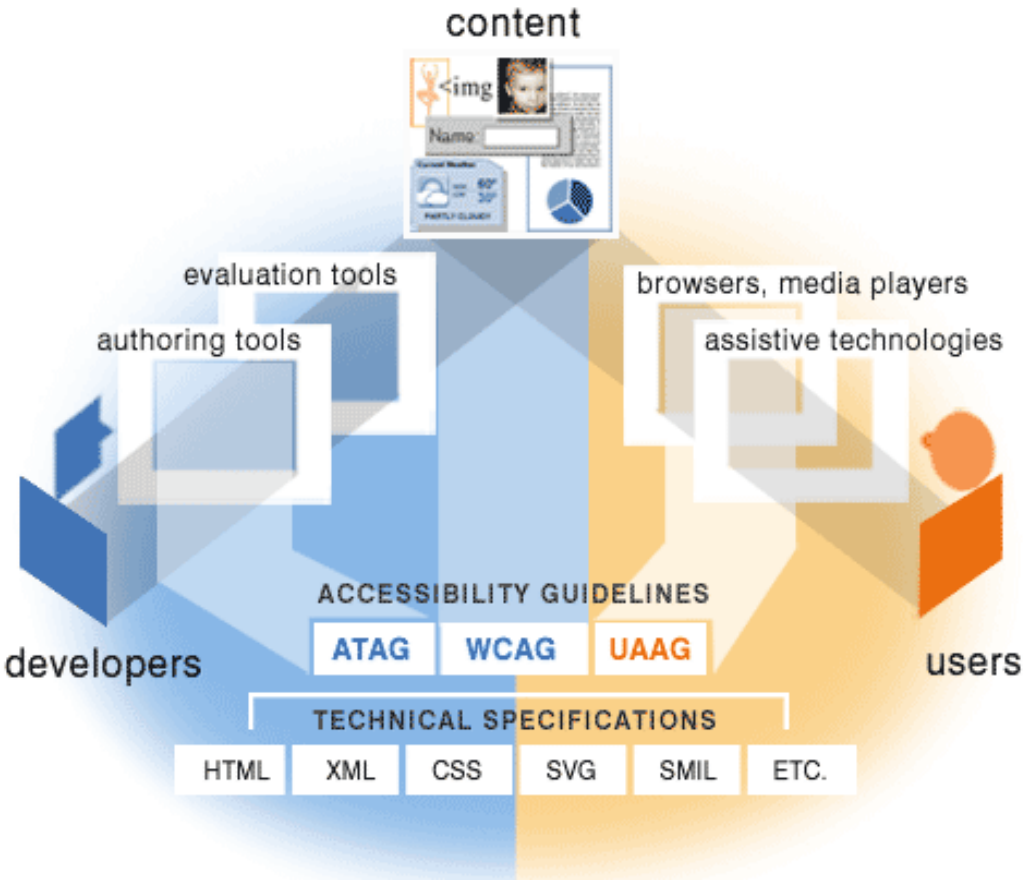
8 Data URI – format identyfikatorów URI pozwalający dołączać dane wewnątrz dokumentu w taki sposób, jak gdyby były linkiem do zdalnego zasobu - (L. Masinter 1998)

9 SVG (ang. *Scalable Vector Graphics*) – standard dokumentów XML służących do opisu dwuwymiarowej grafiki statycznej lub animowanej - (Jun, Ferraiolo, i Jackson 2003)

10 URL (ang. *Uniform Resource Locator*) to podzbiór URI, który definiuje gdzie znajduje się identyfikowany zasób oraz sposób jego pobrania - (T. Berners-Lee, L. Masinter, i McCahill 1994)

Jak widać na rysunku 1.2 te trzy grupy wytycznych bazują na standardach W3C i doprecyzowują jak należy używać tych standardów, aby zapewnić odbiór tworzonych treści WWW (ang. *content*) osobom niedosłyszącym czy niedowidzącym. Pierwsze dwie koncentrują się na autorach narzędzi (po lewej), a trzecia na samych twórcach i odbiorcach treści, czyli użytkownikach (po prawej). W związku z tym ATAG i WCAG dotyczy narzędzi autorskich (ang. *authoring tools*) i weryfikujących (ang. *evaluation tools*). Natomiast UAAG dotyczy pośrednictwa między użytkownikiem i treścią w postaci wspomagania (ang. *assistive technologies*) i narzędzi prezentacji, jak przeglądarki (ang. *browsers*) i odtwarzacze (ang. *media players*). Nacisk na zgodność ze standardami WAI nabiera tu nowego wymiaru: społecznego, a nie jak do tej pory jedynie technologicznego.

Zarówno WAI jak i testy Acid mają na celu zagwarantowanie czytelności WWW dla użytkownika. Z drugiej strony jednak równie ważną, o ile nie ważniejszą, sprawą jest zagwarantowanie czytelności WWW dla maszyn. Ten aspekt staje się szczególnie widoczny w kontekście Sieci Semantycznej, gdzie aby pomóc maszynom automatycznie przetwarzać dane zmieszczone w WWW dodaje się do nich metadane. Oczywistym jest, że skoro metadane są przeznaczone wyłącznie dla maszyn, to muszą być przez nie interpretowane, a to możliwe jest tylko, jeśli z góry wiadomo według jakiego standardu są zapisane. Jednak same metadane na nic się nie zdadzą, gdy nieczytelne będą same dane. Stąd widać jak krytyczne znaczenie dla rozwoju sztucznej inteligencji na platformie WWW ma trzymanie się standardów podczas rozbudowywania i dodawania treści do WWW. Już dziś wiele usług korzysta z metadanych umieszczanych z dokumentach oraz polega na prawidłowej składni tych dokumentów według zadeklarowanego DTD. W przypadku, gdy dokument nie spełnia zadeklarowanych wymagań, najczęściej staje się dla usługi nieczytelny.



Rysunek 1.2. Hierarchiczna struktura aktywności grup roboczych WAI

źródło: (WAI 2009)

Ponieważ dziś każdy użytkownik może tworzyć WWW, a najczęściej nie ma on świadomości istnienia standardów dokumentów WWW, dlatego zgodne ze standardami strony WWW są ciągle w mniejszości. Dzisiejsze masowe budowanie serwisów, najczęściej blogów, oparte jest o przygotowany szablon. Zakładając, że sam szablon strony jest zgodny ze specyfikacjami W3C, co też nie jest częste, to użytkownik dostosowując taki szablon może go bardzo łatwo zepsuć. Najczęściej budowanie i dostosowywanie strony do własnych potrzeb polega na dodawaniu do strony elementów w postaci gotowych fragmentów kodu (X)HTML lub CSS. Wystarczy, że kod taki zostanie wstawiony w niewłaściwe miejsce – nawet o jeden znak za wcześnie, czy za daleko – i już strona przestaje spełniać standardy, co nie jest w żaden sposób sygnalizowane użytkownikowi. Jedyną weryfikacją dla niego jest wygląd strony w przeglądarce, który z kolei nawet jeśli jest poprawny nie musi przekładać się na poprawność kodu strony. Dzieje się tak dlatego, że silniki przeglądarek odpowiadające za wyświetlanie stron są obecnie najbardziej zaawansowanymi agentami WWW, jeśli chodzi o przetwarzanie dokumentów. Chcąc wyjść naprzeciw zwykłym użytkownikom silniki przeglądarek potrafią w dużym stopniu zgadywać intencje użytkownika i poprawiać niezgodny ze standardami kod, tak, aby strona wyświetlała się poprawnie. Takie działanie jest marketingowo uzasadnione jednak wpływa negatywnie na rozwój nowych narzędzi z zakresu przetwarzania danych z WWW. Dbając jedynie o wygląd przeciętny użytkownik najczęściej nie ma świadomości, że niepoprawny kod jego strony uniemożliwia automatyczne przetwarzanie danych na niej zawartych. Chcąc poprawić ten stan, producenci usług opierających się głównie na przetwarzaniu WWW, takich jak wyszukiwarki internetowe, starają się promować i tworzyć świadomość użytkowników odnośnie stosowania się do standardów. Na przykład, jak wynika z testu SEO (Anderson 2008) wyszukiwarka Google przyznaje wyższy PageRank dokumentom spełniającym normy W3C.

Co więcej, serwis Blogger, którego Google jest właścicielem, posiada narzędzie sprawdzania poprawności kodu HTML. Jeśli użytkownik prowadzący blog w tym serwisie, próbuje dodać nowy wpis, który nie jest poprawnym kodem HTML jest o tym informowany. Jednocześnie nowy wpis nie może zostać dodany, dopóki nie zostanie poprawiony i nie będzie zgodny ze standardami. Podobną funkcjonalność ma WordPress, który jednak nie dręczy użytkowników ostrzeżeniami, tylko sam próbuje automatycznie poprawić kod. Na przykład sam automatycznie domyka niezamknięte znaczniki HTML.

1.3 Spójność i zgodność kolekcji dokumentów

W niniejszej pracy interpretacja tytułowych pojęć spójność i zgodność związana jest bardziej z ich semantyką słownikową. W PWN 2008) jedna z definicji pojęcia „spójny” brzmi: „logicznie **powiązany**, **harmonijny**, konsekwentny”. Natomiast przy pojęciu „zgodny” znajdziemy wyjaśnienia „**niesprzeczny z** czymś”, „jednomyślny, jednakowy, **harmonijny**”. Definicje obu pojęć częściowo się pokrywają, co widać na przykład przez powtórzenie w obu terminu „harmonijny”. W niniejszej pracy również oba pojęcia traktowane są podobnie, ponieważ oba dotyczą pewnego stopnia „jednakowości” dokumentów w kolekcji. To czym różni się semantyka obu pojęć w kontekście rozwiązywanego w pracy problemu to rodzaj informacji brany pod uwagę przy badaniu podobieństwa dokumentów. Spójność dotyczy związków między dokumentami wynikających z powiązań, głównie na poziomie metadanych. Z kolei zgodność to podobieństwo na poziomie danych, wynikające np. z ich niesprzeczności. Rozpatrzmy dwa przykłady.

Przykład 1: Słońce. Mamy dwie prognozy pogody na dany dzień. Ich spójność jest wysoka, ponieważ wynika z typu informacji i czasu, którego dotyczy. Natomiast, zgodność ich będzie niewielka, gdy jedna prognoza będzie przewidywać słońce, a druga deszcz.

Z drugiej strony: weźmy pod uwagę 2 komunikaty prasowe. Ich zgodność jest wysoka, ponieważ oba dotyczą słońca. Jednak spójność komunikatów będzie niewielka, ponieważ jeden jest komunikatem meteorologicznym mówiącym o pogodzie, a drugi astronomicznym dotyczącym centralnego punktu układu słonecznego.

Przykład 2: Album muzyczny. Załóżmy, że mamy recenzje dwóch albumów muzycznych. Recenzje są spójne, ponieważ mają jednego autora, ale nie są zgodne, ponieważ jedna recenzja jest pozytywna, a druga negatywna. Alternatywnie możemy mieć dwie opinie na temat albumu, które będą zgodne, ze względu na pozytywną ocenę. Jednak ich spójność będzie niewielka, gdy jedna opinia to autoryzowana recenzja, a druga to średnia ocen klientów sklepu internetowego.

W przykładzie 1 wysoka spójność wynika z metadanych dotyczących prognozy. Natomiast wysoka zgodność z danych, czyli treści dotyczącej słońca. W przykładzie 2 jeden autor obu recenzji wskazuje na wysoką spójność metodologii i stylu literackiego użytego w ocenach. Z kolei jednoznaczna pozytywność dwóch ocen albumu wpływa na ich wysoką zgodność, jednak pochodzenie tych ocen wskazuje na niewielką ich spójność.

1.4 Sformułowanie problemu

Tak, jak w większości prac dotyczących wyszukiwania informacji, tak i w tej pracy problemem wymagającym rozwiązania jest potrzeba informacyjna użytkownika systemu wyszukiwania informacji. Aby tą potrzebę użytkownika sprecyzować należy najpierw przytoczyć pojęcia relewancji i pertynencji.

1.4.1 Relewancja i pertynencja

Używając w wyszukiwaniu informacji pojęcia relewancja najczęściej mamy na myśli stopień w jakim otrzymana w odpowiedzi informacja pasuje do zadanego pytania. Gdy wyszukujemy dokumenty relewancja dotyczy zgodności tematyki dokumentów z tematyką zawartą w pytaniu. To najpopularniejsze rozumienie relewancji w Saracevic 2007) nazwane jest relewancją tematyczną. Powołując się na (Swanson 1986) Saracevic zaznacza dwoistość relewancji dzieląc ją na **obiektywną** i **subiektywną**. Relewancja obiektywna jest kluczowa dla projektowania i testowania bibliograficznych systemów wyszukiwania, a relewancja subiektywna jest krytyczna podczas używania takich systemów. Czyli relewancja subiektywna zakłada scenariusz: cokolwiek użytkownik powie, że jest relewantne – jest uznawane za relewantne. Użytkownik jest ostatecznym sędzią. Relewancja obiektywna natomiast wynika z logicznego związku między dokumentami. Ocena użytkownika nie ma tu znaczenia. Ta dwoistość relewancji tematycznej według podziału Saracevic'a dotyczy **relewancji słabej** (systemowej) i **silnej** (użytkownika). Jednak podział relewancji tematycznej jest punktem wyjścia dla innych relewancji badanych przez naukowców na przestrzeni ostatnich kilku dziesięcioleci. Próbując je klasyfikować Saracevic układa relewancje na osi użycia informacji w kierunku od komputera do użytkownika. Stronę komputera i stroną użytkownika dzieli interfejs. Po stronie komputera mamy do czynienia z **relewancjami**: **treści**, **przetwarzania** i **inżynierii**. Dalej od interfejsu w kierunku użytkownika znajdują się **relewancje**: **zapytania**, **kognitywna**, **afektu** i **sytuacyjna**. Dodatkowo wszystkie te relewancje zależą od kontekstu, np. społecznego czy kulturowego. Zamiast używania przydawki precyzującej o którą relewancję chodzi, niektórzy autorzy wprowadzają inne terminu dla określenia różnych typów relewancji. Takimi terminami są np.: „**pertynencja**” oznaczająca „**relewancję kognitywną**” czy „**użyteczność**” oznaczająca „**relewancję sytuacyjną**”. Z kolei w Soergel 1994) autor proponuje zagnieżdżenie pojęć. W tym kontekście: obiekt informacyjny (np. dokument):

- jest **relewantny tematycznie**, jeśli odpowiada na pytanie użytkownika;
- jest **pertynentny**, jeśli jest relewantny tematycznie i właściwy dla użytkownika – użytkownik rozumie i może użyć zwróconą informację;
- jest **użyteczny**, jeśli jest pertynentny i daje użytkownikowi nową informację.

Na przykład: jeśli użytkownik poszukuje informacji na dany temat i znajduje dokument pasujący tematycznie swojego autorstwa, to dokument jest relewantny i pertynentny, ale nie jest użyteczny.

W dalszej części pracy, relewancja w kontekście kolekcji dokumentów zwróconej przez

wyszukiwarkę, będzie rozumiana jako relewancja subiektywna, silna, czyli użytkownika. Według dalszej klasyfikacji relewancja, o której dalej będzie mowa to relewancja kognitywna, dlatego zamiennie używanym terminem będzie „pertynencja".

1.4.2 Problem badawczy

Rozważany i rozwiązywany w pracy problem ma być odpowiedzią na potrzebę informacyjną użytkownika. Tak jak we wszystkich systemach wyszukiwania informacji potrzebą użytkownika jest najszybsze i najwygodniejsze znalezienie najpełniejszej i najtrafniejszej informacji. Mówiąc bardziej technicznie: użytkownik chce móc jak najprościej zdefiniować zakres poszukiwanych informacji i otrzymać jak najprecyzyjniejszą odpowiedź w postaci kolekcji obiektów (w tym przypadku dokumentów). Problem ze zrealizowaniem tego wynika z faktu, że użytkownik najczęściej nie jest ekspertem w dziedzinie, w której poszukuje odpowiedzi. Przez to nie zna terminologii i trudno jest mu sformułować swoje zapytanie w języku zrozumiałym przez wyszukiwarkę.

Najbardziej intuicyjną i najczęściej używaną metodą odpytywania systemu wyszukiwawczego jest wyrażenie boolowskie, które w najprostszej postaci może być ciągiem słów kluczowych. Użytkownik wprowadza jedynie słowa, które mają identyfikować interesujące go dokumenty, a wyszukiwarka wstawiając między te słowa operatory AND lub OR dostaje wyrażenie, które może być zapytaniem w modelu boolowskim (Salton i McGill 1983, 202-203). Problem braku możliwości uporządkowania dokumentów w odpowiedzi w takim modelu był genezą powstania modelu wektorowego. Tu co prawda funkcja wyszukiwawcza zwraca wartości niebinarne, dzięki którym można posortować dokumenty w odpowiedzi względem relewancji, ale problemem staje się pytanie, które jest wektorem niebinarnym. Taki, nieintuicyjny sposób zadawania zapytań został wyeliminowany przez model oparty na logice rozmytej, który łączył dobre cechy obu swoich poprzedników. Z modelu boolowskiego przejął przyjazne zapytania – w postaci wyrażenia logicznego, a z modelu wektorowego dziedzinę funkcji wyszukiwawczej, dzięki której użytkownik mógł otrzymać najpierw najbardziej relewantne dokumenty.

Ze względu na fakt zderzenia tradycyjnych metody wyszukiwania z potrzebą przeszukiwania Internetu, powstało wiele metod ułatwiania użytkownikowi otrzymania relewantnej odpowiedzi, jako adaptację tych metod dla WWW. Według (Manning, Raghavan, i Schütze 2008) kluczowymi metodami poprawiającymi wyniki tradycyjnego, wyszukiwania opartego wyłącznie na słowach kluczowych są metody bazujące na analizie linków. Wspomniana praca dzieli potrzeby informacyjne użytkownika, czyli potrzeby zapytań na trzy rodzaje:

1. Zapytania informacyjne – ogólne zapytania dotyczące szerokiej tematyki, jak „białaczka" czy „Prowansja". Wszystkie poszukiwane informacje najczęściej nie znajdują się na jednej stronie WWW i użytkownik musi przejrzeć ich kilka.
2. Zapytania nawigacyjne – dotyczą znalezienia serwisu lub strony domowej pojedynczego obiektu, np. linie lotnicze Lufthansa. W takim wypadku użytkownik spodziewa się, że znajdzie szukaną stronę na pierwszej pozycji wyników.
3. Zapytania transakcyjne - które mają być wstępem do wykonania przez użytkownika pewnej transakcji, np.: zakup produktu, ściągnięcie pliku, zrobienie rezerwacji. W taki przypadkach wyszukiwarka powinna zwrócić strony usług dających interfejs do przeprowadzenia takich transakcji.

Problem badawczy pracy to: opracowanie metod analizy spójności i zgodności kolekcji dokumentów WWW, umożliwiających uzyskanie polepszenia jakości wyszukiwania w porównaniu z wyszukiwaniem za pomocą tradycyjnej wyszukiwarki opartej na ważeniu terminów. Użytkownikowi zasiadającemu do terminala wyszukiwarki będą proponowane odpowiedzi o relewancji zwiększonej dzięki wykorzystaniu miar spójności i zgodności. Formalnie rzecz ujmując: graf modelujący przeszukiwaną kolekcję dokumentów będzie analizowany za pomocą tych miar tak, aby przedstawić odpowiedź w sposób w danej chwili żądany przez użytkownika

wyszukiwarki. Na strukturę odpowiedzi wpływać może reranking[11] oraz poziom ziarnistości grupowania dokumentów w odpowiedzi wynikające z żądanej spójności i zgodności wyników. A sama analiza dotycząca struktury zależna jest od chwilowej perspektywy użytkownika, czyli kontekstu przeglądanych wyników.

1.5 Cel pracy

Celem pracy jest rozwiązanie postawionego powyżej problemu. Genezą problemu analizy kolekcji dokumentów jest potrzeba informacyjnej użytkownika WWW. Dzięki zdefiniowaniu spójności i zgodności – wynikająca z potrzeby użytkownika – analiza może dotyczyć różnych aspektów dokumentów w kolekcji. Taka analiza wyników wyszukiwania motywuje wprowadzenie nowego typu zapytań.

1.5.1 Teza

Relewancja odpowiedzi i intuicyjność to podstawowe cechy dobrej wyszukiwarki. Spełnienie tych kryteriów to najczęściej wystarczający wymóg przeciętnego użytkownika szukającego prostej odpowiedzi, zawartej w jednym z wielu dokumentów. Najpopularniejszy scenariusz wyszukiwania to zapytanie informacyjne, czyli wpisanie kilku słów kluczowych i znalezienie odpowiedzi zawartej w większości z kilkunastu pierwszych dokumentów odpowiedzi. Taki scenariusz jest w stanie obsłużyć typowa wyszukiwarka oparta na metodach ważenia terminów. Jednak często zdarza się, że potrzebny jest szerszy aspekt analizy kolekcji dokumentów. Często nie chodzi o znalezienie konkretnej informacji zawartej dokumentach czy nawet konkretnego dokumentu, ale o analizę ilościową i jakościową dokumentów spełniających zadane kryteria. Taka potrzeba nie mieści się w klasyfikacji zapytań z poprzedniego podrozdziału zaproponowanej przez (Manning, Raghavan, i Schütze 2008).

Przykład 1: samouczek. Wyjaśniając to na przykładzie: można wyobrazić sobie użytkownika poszukującego w WWW samouczka dotyczącego pewnego standardu. Skorzystanie z pierwszego znalezionego nie jest dobrym rozwiązaniem. Z drugiej strony stwierdzenie czy dany samouczek odpowiada użytkownikowi wymaga bardziej szczegółowego zagłębienia się w treść, niż w przypadku próby stwierdzenia czy dokument jest na temat czy nie. Zakładamy, że wszystkie znalezione samouczki są na temat, ale nadal pozostaje problem, który wybrać. Sprawdzenie (wypróbowanie) każdego nie wchodzi w rachubę, ponieważ jest zbyt kosztowne. W takich okolicznościach istnieje potrzeba zestawienia znalezionych dokumentów w kolekcję i przeanalizowanie jej w innych aspektach. Dzięki analizom grupującym według pewnego stopnia spójności i/lub zgodności użytkownik może dokonać ostatecznego wyboru samouczka na przykład w oparciu o informacje:

- czy jest to część większej serii samouczków,
- czy jest on napisany w kontekście innych standardów, technik, języków programowania używanych przez użytkownika,
- jak bardzo źródło samouczka jest autorytatywne (czy np. został on przygotowany przez twórców standardu, którego dotyczy)
- ile i jakie komentarze otrzymał od innych użytkowników
- czy zawiera dodatkowe dokumenty (np. gotowe kody źródłowe, działające online przykłady, itd.)

Takie zapytanie można by próbować podciągnąć pod zapytanie informacyjne, ale w rzeczywistości jest to bardziej szczegółowy proces. Powyższy przykład może być mało sugestywny dla szerszego grona użytkowników WWW, dlatego przyjrzyjmy się przykładowi sytuacji częściej spotykanej w Internecie.

11 Reranking – ranking ustalający nowy porządek (kolejność) wyników wyszukiwania stworzony ponownie na podstawie pierwotnego rankingu oraz dodatkowych informacji – w tym przypadku informacji semantycznej

Przykład 2: aukcja. Wyobraźmy sobie użytkownika, który poszukuje pewnego przedmiotu na popularnej platformie aukcji internetowych. Choć poszukiwanie dotyczy przedmiotu, to tak naprawdę bezpośrednio szukany jest dokument opisujący aukcję wraz z opisem tego przedmiotu. Czyli potrzebą informacyjną użytkownika najczęściej nie jest znalezienie pierwszej z brzegu aukcji poszukiwanego przedmiotu, ale wybranie najbardziej mu odpowiadającej. Załóżmy, że znaleziona kolekcja dokumentów to aukcje jednakowego, z punktu widzenia użytkownika, przedmiotu. Aby wybrać najlepszą dla siebie ofertę, dzięki użyciu faset[12], grupowania i sortowania, użytkownik może przeanalizować dostępne aukcje np. pod kątem:

- popularności przedmiotu
- ceny i rodzajów przesyłki
- wiarygodności sprzedawcy
- odległości od siedziby sprzedawcy (mający wpływ na czas transportu)
- dodatkowych opcji (płatności, promocji)

Dodatkowo umożliwiając użytkownikowi analizę spójności i zgodności kolekcji aukcji mógłby on uzyskać np. informację o:

- wiarygodności sprzedawcy niezależnej od platformy aukcyjnej
- weryfikacji czy w kontekście innych platform/sklepów/porównywarek cen dana cena aukcji jest podejrzanie niska
- testach i recenzjach dotyczących przedmiotu aukcji z niezależnych źródeł
- popularności przedmiotu aukcji w zestawieniu z nowszymi/starszymi wersjami (jeśli jest to przedmiot produkowany seryjnie)
- popularności przedmiotu aukcji w zestawieniu z produktami konkurencyjnymi
- weryfikacji promocji aukcji

Wszystkie dodatkowe możliwości wynikające z analizy spójności i zgodności opierają się o rozszerzenie kolekcji dokumentów opisujących aukcje o dokumenty w innych serwisów WWW. Ten przykład ma wiele wspólnego z zapytaniem transakcyjnym, ale głównym problemem nie jest znalezienie usługi umożliwiającej rozpoczęcie transakcji licytacji, ale wybranie odpowiedniej aukcji, który ma być obiektem transakcji. W związku z tym wydaje się, że do zaproponowanej klasyfikacji należałoby dodać „zapytanie analityczne". Taki rodzaj zapytań spełniałby założenia obu powyższych przykładów.

Istotnym aspektem przedstawionej w przykładach analizy jest grupowanie, pozwalające ogarnąć wyszukaną kolekcję dokumentów na wyższym poziomie ziarnistości. Grupowanie powinno pozwalać na regulowanie wielkości i liczby grup. Specyficznym rodzajem grupowania, jest wspomniane używanie faset dające ogólny pogląd na liczbę i liczność grup tworzonych według predefiniowanego kryterium. Z drugiej strony fasety pozwalają na zawężanie wyszukiwania (ang. *drill-down*) i analizę na wyższym poziomie szczegółowości poprzez dodatkowe filtrowanie wyników.

Grupowanie i filtrowanie kolekcji dokumentów to kluczowe elementy pozwalające na analizę wyników na różnych poziomach spójności i zgodności. Dzięki sprecyzowaniu oczekiwanej spójności i zgodności możliwe jest zawężanie lub poszerzanie zakresu poszukiwań niezależnie od tradycyjnych kryteriów, takich jak słowa kluczowe czy autor. Regulacja ziarnistości wyników za pomocą spójności i zgodności możliwa jest dzięki wydobyciu z dokumentów informacji semantycznej i określeniu związków pomiędzy dokumentami, użytkownikami tworzącymi te dokumenty i pojęciami używanymi do ich opisywania.

12 Klasyfikacja fasetowa pozwala na wielokrotną klasyfikację obiektów, dzięki czemu można je porządkować na różne sposoby, a nie tylko według predefiniowanego porządku. Przykładowo: kolekcję dokumentów można grupować i przeglądać według autora, tematyki, daty, itp. mając jednocześnie informację o liczności tych grup dla danej klasyfikacji

Teza: **Dzięki wykorzystaniu spójności i zgodności, wynikających z informacji semantycznej na temat obiektów WWW i ich związków, możliwe jest poprawienie dokładności i kompletności oraz sposobu prezentacji wyników wyszukiwania.**

1.5.2 Propozycja rozwiązania

W celu rozwiązania problemu, czyli opracowania algorytmów wyznaczania i stosowania spójności i zgodności dla kolekcji dokumentów na różnych poziomach, niezbędne jest rozwiązanie następujących problemów składowych:

1. analiza semantyki kolekcji dokumentów WWW
2. analiza związków między obiektami: dokument - autor - pojęcie (DAC)
3. modelowanie grafu DAC
4. określenie miar spójności i zgodności dla różnych poziomów związków
5. grupowanie grafu DAC ze względu na spójność i zgodność
6. określenie perspektyw prezentacji i interakcji z pogrupowanymi wynikami

Oryginalnym i najważniejszym problemem składowym pracy jest modelowanie związków za pomocą grafu DAC. Związki te to nie tylko bezpośrednie powiązania pomiędzy dokumentami, ale również związki wynikające z powiązań pomiędzy autorami dokumentów oraz związków ontologicznych pomiędzy pojęciami opisującymi dokumenty. Określenie miar spójności i zgodności dotyczy skupienia się na konkretnych poziomach związków. Oznacza to, że ze względów praktycznych nie można stworzyć jednej miary uwzględniającej wszystkie poziomy związku. Na przykład, chcąc wyrazić liczbowo siłę związku między dwoma autorami dokumentów możemy wziąć pod uwagę:

- deklarowanie faktu znajomości za pomocą formatu FOAF
- deklarowanie jakości związku za pomocą formatu XFN
- współautorstwo dokumentów
- współuczestnictwo w zdarzeniach, np. konferencjach
- wzajemne cytowanie/komentowanie dokumentów
- podobieństwo preferencji (profili z serwisów społecznościowych), np. gusta muzyczne (last.fm), zakładki WWW (del.icio.us), listy życzeń, itp.
- aspekty demograficzne (lokalizacja, język, ...)
- itd.

Poza tym, że w praktyce rzadko mamy jednoczesny dostęp do danych na temat wielu poziomów związków, to ujecie ich wszystkich w jednej mierze byłoby nieefektywne choćby ze względu na trudność w interpretacji takiej miary. Konkretne algorytmy wyznaczania miary związków muszą uwzględniać charakterystykę danego poziomu związków w połączeniu z innym, dlatego w tej pracy zaproponowane zostaną miary jedynie dla kilku przypadków/kombinacji poziomów związków. Ostatecznie algorytmy mierzenia i wykorzystania spójności i zgodności podzielone zostały na dwa podejścia:

1. podejście ogólne (teoretyczne), uwzględniające szeroki zakres informacji semantycznej możliwej do wykorzystania;
2. podejście szczegółowe (praktyczne), uwzględniające wybrane aspekty (poziomy) związków, pozwalające przeprowadzić eksperyment weryfikujący metodę.

2 Analiza semantyki dokumentów WWW

Pierwszy krokiem w kierunku analizy WWW jest stworzenie jej modelu. Aby model był jak najbardziej uniwersalny, a jednocześnie możliwie najwierniej odzwierciedlał cechy obiektów ze świata rzeczywistego należy przyjąć definicje, które uściślą modelowany świat i jego atrybuty. W pracy używany będzie model kolekcji dokumentów WWW, więc najistotniejsze dla modelu są przyjęte definicje pojęć „dokument WWW" i „kolekcja".

2.1 Dokument WWW

Definicja „dokumentu WWW" (ang. *Web document*) jest kluczowym zagadnieniem leżącym u podstaw przetwarzania danych zawartych w „Ogólnoświatowej Pajęczynie". Zależnie od przyjętej definicji, wyniki przeprowadzanych analiz WWW mogą dawać wyniki znacznie różniące się, zarówno ilościowe, jak i jakościowe.

W słowniku W3C (Thereaux 2008) znajduje się 7 definicji pojęcia „dokument". Mnogość tych definicji wynika z liczby kontekstów, w jakich są używane, a więc liczby źródeł, w których pojęcie „dokument" zostało niezależnie zdefiniowane. Źródła to głównie rekomendacje W3C, a więc specyfikacje uznanych standardów używanych przy budowaniu i analizie WWW. Przytoczone poniżej definicje pokazują jak różna może być semantyka pojęcia „dokument".

Definicja 2.1.1. Dokument

„Dokument to strumień danych, który po złożeniu z każdym innym strumieniem, do którego się odwołuje, ma taką strukturę, że przechowuje informacje zawarte w elementach zorganizowanych tak, jak zdefiniowano w powiązanym DTD. [...]" - (Steven Pemberton 2002)

Definicja 2.1.2. Obiekt dokumentu, Model Obiektu Dokumentu (DOM)

„W ogólnym użyciu, termin „obiekt dokumentu" odnosi się do reprezentacji danych (np. dokumentu) przez agenta użytkownika. Dane ogólnie pochodzą ze źródła dokumentu, ale mogą również być wygenerowane (np. z arkuszy stylów, skryptów lub transformacji), stworzone w wyniku ustawień preferencji w agencie użytkownika lub dodane jako wynik naprawy przeprowadzonej automatycznie przez agenta użytkownika. Niektóre dane, będące częścią źródła dokumentu są rutynowo renderowane (np. w HTMLu, to, co pojawia się pomiędzy znacznikami początkowymi i końcowymi elementów oraz wartości atrybutów, takich jak *alt*, *title*, i *summary*). Inne części obiektu dokumentu są ogólnie przetwarzane przez agenta użytkownika bez świadomości użytkownika, takie jak DTD lub zdefiniowane przez schemat nazwy typów i atrybutów, oraz inne wartości atrybutów takich jak *href* i *id*. Większość wymagań tego dokumentu (czyli specyfikacji, której pochodzi niniejsza definicja - *przyp. tłum.*) odnosi się do obiektu dokumentu po jego skonstruowaniu. Jednakże, kilka punktów [...] może wpływać na konstruowanie obiektu dokumentu." - (Ian Jacobs, Gunderson, i Hansen 2002)

Definicja 2.1.3. Dokument

„Dokument odnosi się do drzewa, którego korzeniem jest Węzeł Dokumentu." - (Malhotra et al. 2007)

Definicja 2.1.4. Dokument

„Dokument to ciąg elementów zdefiniowanych w języku znaczników (np. HTML4 lub aplikacji XML)." - (Treviranus et al. 2000)

Definicja 2.1.5. Węzeł

„Jednostka informacji. Znana również jako rama (KMS), karta (Hypercard, Notecards). Używana z tym specjalnym znaczeniem w kręgach hipertekstowych: nie pomylić z „węzłem" oznaczającym „komputer w sieci". Na korzyść użytkownika, używamy terminu „dokument", ponieważ jest to najbliższy termin poza światem hipertekstu." - (Girschweiler 1995)

Definicja 2.1.6. Dokument

„Termin dla węzła (patrz definicja powyżej – *przyp. tłum.*) w niektórych systemach (np. Intermedia). Czasami używany przez innych jako termin dla kolekcji węzłów na powiązane tematy, umożliwiająca przechowywanie lub rozpowszechnianie jako jedną całość." - (Girschweiler 1995)

Definicja 2.1.7. Dokument

„Na potrzeby tej specyfikacji, „dokument" odnosi się do zawartości dostarczonej w odpowiedzi na zapytanie. Używając tej definicji, „dokument" może być kolekcją mniejszych „dokumentów", które, z kolei, są częścią większego „dokumentu"." - (Hjelm et al. 2004)

Definicja 2.1.8. Dokument

„Każde dane, które mogą być zaprezentowane w formie cyfrowej." - (Brown i Haas 2004)

Definicja 2.1.1 pochodząca z rekomendacji XHTML 1.0 – najpopularniejszego obecnie formatu publikowania danych w WWW – uściśla, że pojęcie „dokument" ma oznaczać ustrukturyzowany (według DTD) strumień danych połączony z innymi strumieniami. Czym są „inne strumienie" określa dokładniej definicja 2.1.2. W tej definicji „strumień danych" to „źródło dokumentu", a „inne strumienie" to np. arkusze *CSS*, skrypty *JavaScript* czy transformacje *XSL*. Są to strumienie, które wpływają na wygląd dokumentu. Dodatkowo na stronę wizualną dokumentu, według definicji 2.1.2, mogą wpływać ustawienia przeglądarki, czy automatyczna korekta analizowanego przez przeglądarkę kodu.

Zamiast „przeglądarki" w definicji używane jest bardziej ogólne pojęcie „agenta użytkownika", jednak w ogromnej większości przypadków korzystania z dokumentów WWW używana jest właśnie przeglądarka internetowa. Dlatego w pracy, w tym znaczeniu, używane będzie pojęcie „przeglądarki", zwłaszcza, że w niektórych kontekstach w dalszej części pracy pojęcie „agent użytkownika" ma odmienne znaczenie.

W definicji 2.1.1 i 2.1.2 uwidoczniony jest podział dokumentu na dane („strumień danych", „źródło dokumentu") i wygląd („inne strumienie", „dane wygenerowane, stworzone przez ustawienia lub korekty"). Jest to bardzo istotny element definicji dokumentu. Podział ten ma szczególne znaczenie przy przetwarzaniu dokumentów przez maszyny, które w celu wydobycia wiedzy powinny mieć dostęp do danych niezależnie od formy ich prezentacji. Dlatego standardy W3 przewidują rozbicie dokumentów WWW na dokument z danymi i dokument z metadanymi dotyczącymi formy prezentacji tych danych. Najpopularniejszą parą takich dokumentów (czy raczej, w myśl definicji, „strumieni") są XHTML (treść) i CSS (format). Mniej popularną, ale również często stosowaną, parą strumieni są XML i XSL (Clark 1999).

Istotną różnicą jaką daje się zauważyć między definicjami 2.1.1 i 2.1.2 jest fakt, że definicja 2.1.1 za „dokument" uznaje bardziej aspekt treści niż wyglądu. Z kolei definicja 2.1.2 mówi, że obiektem dokumentu, czyli dokumentem w podejściu obiektowym (DOM) jest „reprezentacja danych", czyli wygląd.

Interpretacja definicji 2.1.1 jest zrozumiała w kontekście specyfikacji XHTML, która przewiduje

możliwość łączenia strumienia danych z różnymi strumieniami dotyczącymi prezentacji. To znaczy jeden dokument XHTML może mieć kilka alternatywnych arkuszy stylów CSS, które mogą odpowiadać kilku różnym wyglądom dokumentu lub za prezentację na różnych urządzeniach. Czyli niezależnie, czy oglądamy dokument na ekranie komputera, na wyświetlaczu telefonu czy na kartce po wydrukowaniu – pomimo, że formatowanie może się różnić – nadal jest to ten sam dokument. Z kolei w definicji 2.1.2, która formułowana jest w kontekście specyfikacji dostępności oczywiste jest, że prezentacja dokumentu wyświetlanego na ekranie będzie miała inne wymagania niż wydruk z drukarki. W związku z tym według definicji 2.1.2 dokument na ekranie i dokument na kartce to 2 różne dokumenty. Obie definicje wskazują na fakt zachowania przez dokument struktury (zgodnej z DTD).

Struktura dokumentu jest też głównym klasyfikatorem w kolejnych dwóch definicjach: 2.1.3 i 2.1.4. Definicja 2.1.3 mówi o hierarchicznej (drzewiastej) strukturze dokumentu. Jest to zrozumiałe ze względu na kontekst wyszukiwania i dostępu do poszczególnych elementów (węzłów) dokumentu. Z kolei definicja 2.1.4 jest bardziej ogólna i stawia jednie warunek, że dokument musi być co najmniej zbiorem uporządkowanym.

Wspólną cechą definicji 2.1.1-2.1.4 jest założenie struktury dokumentu, z którego wynika fakt, że w praktyce jest to zasób tekstowy. Może on co prawda zawierać dane multimedialne, jak obrazy, dźwięki, wideo czy animacje, ale same elementy i ich struktura opisane są tekstowo, w języku znaczników. Mniej restrykcyjne w tym względzie są definicje 2.1.5-2.1.8. W przypadku tych definicji dokumentem WWW może być np. zdjęcie opublikowane w WWW.

Definicje 2.1.5 i 2.1.6 to jedne z najstarszych definicji W3C. W definicjach tych przedstawione jest pochodzenie terminu „dokument” jako najbliższego terminowi „węzeł”, który używany jest, w tym samym kontekście, w innych, niż WWW, środowiskach hipertekstowych. Definicja 2.1.7 przyjmując, że każdy element zwrócony w odpowiedzi na zapytanie do WWW może być dokumentem, wskazuje na rekurencję definicji pojęcia „dokument”, czyli dokument może się składać z mniejszych dokumentów. Definicja 2.1.8 jest najbardziej ogólną definicją pojęcia „dokument”, co wskazuje na potrzebę istnienia możliwości odniesienia się do wielu rodzajów zasobów za pomocą jednego terminu. Tym terminem najczęściej jest właśnie „dokument”. W świetle definicji 2.1.8 pojęcie „dokument WWW” jest równoważne pojęciu „zasób WWW”.

Konkretne definicje potrzebne w celu rozwiązania problemu pracy zostały wybrane w rozdziale 5 Metody analizy spójności i zgodności s. 70.

2.1.1 Strona WWW

Będąc przy pojęciu „dokument WWW” warto zwrócić uwagę na pojęcie często z nim utożsamiane. Mowa o „stronie WWW”.

Definicja 2.1.9. Strona WWW (ang. *webpage*)

„Kolekcja informacji, składająca się z jednego lub więcej osadzonych zasobów WWW, z zamierzeniem, aby wyświetlać je jednocześnie i identyfikować przez pojedyncze URI. Bardziej szczegółowo, strona WWW składa się z zasobu WWW z zero, jednym lub więcej osadzonymi zasobami WWW, z zamierzeniem, aby wyświetlać je jako jedność i odnosić się do niego przez URI tego jednego zasobu, który nie jest osadzony.

Uwaga: Komponenty strony WWW mogą znajdować się w różnych lokalizacjach sieciowych. Lokalizacja strony WWW, jednakże, determinowana jest przez URI identyfikujące stronę.

Uwaga: Zakres strony WWW jest ograniczony do kolekcji zasobów WWW, które wyświetlane jednocześnie poprzez zażądanie URI tej strony WWW. Komponenty strony WWW rzeczywiście wyświetlane w widoku strony są zależne od klienta.” - (Lavoie i Nielsen 1999)

Definicja 2.1.10. Strona WWW

„Dokument podłączony do WWW i oglądalny przez każdego podłączonego do internetu [sic], kto posiada przeglądarkę internetową.” - (George A. Miller 2006)

Interpretacja definicji 2.1.9 uwidacznia zbieżność z definicjami 2.1.4-2.1.7. Jeśli za „elementy” z definicji 2.1.4 przyjąć „osadzone zasoby WWW”, to w myśl tej definicji i definicji 2.1.9 „strona WWW” jest „dokumentem WWW”. Podobnie ma się sprawa w kontekście definicji 2.1.5, jeżeli przyjmiemy, że "strona WWW” jest jednostką informacji w WWW odpowiadającą „ramie” czy „karcie” w innych systemach. Definicje 2.1.6 i 2.1.7 również nie wykazują elementów dyskryminujących „stronę WWW” jako „dokumentu WWW”, a zagnieżdżona budowa dokumentu WWW wyeksponowana w tych definicjach współgra z osadzaniem zasobów WWW według definicji 2.1.9.

Inne źródła, jak systemy biblioteczne uczelni wyższych, regulaminy systemów oferujących usługi on-line czy WordNet (definicja 2.1.10) wprost zrównują „stronę WWW” z „dokumentem WWW”. Wynika to z faktu, że definicje w tych źródłach używane są w kontekście tekstów skierowanych do osób najczęściej nie posiadających szczegółowej wiedzy z zakresu WWW. Jedną z niewielu prób rozróżnienia „strony WWW” i „dokumentu WWW” podjęli autorzy Wikipedii.

Definicja 2.1.11. Strona WWW

„Strona WWW (ang. *Web page* lub *webpage*) jest zasobem informacji pasującym do Ogólnoświatowej Pajęczyny (WWW) dostępnym poprzez przeglądarkę internetową. Informacja ta jest zwykle w formacie HTML lub XHTML i może zawierać nawigację do innych stron WWW poprzez odsyłacze hipertekstowe (ang. *hypertext links*).

Strony WWW mogą być pobierane z lokalnego komputera lub ze zdalnego serwera WWW. Serwer WWW może ograniczyć dostęp tylko do sieci prywatnej, np. firmowego intranetu lub może publikować strony w WWW. Strony WWW są żądane i serwowane z serwera WWW używając protokołu HTTP.

Strony WWW mogą składać się ze statycznych tekstów przechowywanych w systemie plików serwera WWW (statyczne strony WWW) lub serwer WWW może konstruować (X)HTML dla każdej strony WWW, w momencie zażądania jej przez przeglądarkę (dynamiczne strony WWW). Lepszą interakcję użytkownika ze stronami WWW mogą zapewnić skrypty po stronie klienta (ang. *client-side scripting*), gdy strona znajduje się już w przeglądarce.

Strona WWW jest typem dokumentu WWW” - (Contributors 2008)

Definicja 2.1.12. Dokument WWW

„Dokument WWW jest pojęciem podobnym do strony WWW, z tym, że jest terminem szerszym z następującymi różnicami:

	Strona WWW	**Dokument WWW**
Protokół przesyłu	Protokoły HTTP(S)	Protokoły HTTP(S) lub dowolny inny protokół komunikacji internetowej.
Format dokumentu	(X)HTML	(X)HTML lub dowolny inny typ poprawnego MIME Content-Type, jak np. ISO OpenDocument.
Kontekst	Strona	Strona, załącznik e-mail'a lub wiele innych rodzajów aplikacji klienckich.
Podgląd	Przeglądarka	Przeglądarka lub aplikacja kompatybilna z MIME.”

Tabela 2.1. Porównanie cech strony WWW i dokumentu WWW (integralna część definicji 2.1.12)

źródło: (Contributors 2008)

Definicja 2.1.9 jest - relatywnie do rozwoju WWW - bardzo starą definicją. Definicja 2.1.12 jest próbą stworzenia współczesnej definicji pojęcia „dokument WWW". W dyskusji przy artykule „Web document" w Wikipedii jako motywacja podany jest brak istnienia standardowej lub „bardziej autorytatywnej" definicji pojęcia „dokument WWW" oraz chęć stworzenia tej definicji metodą konsensusu. Istnienie tego artykułu w Wikipedii jest również niezbędne jako odwołanie z wielu innych artykułów.

Konstrukcja definicji 2.1.12 opiera się na uogólnieniu definicji 2.1.11. Definiując, że strona WWW jest elementem WWW w formacie (X)HTML dostarczanym do przeglądarki za pomocą protokołu HTTP(S) - zaznaczono, że jest ona typem dokumentu WWW. Rozszerzając więc definicję 2.1.11 dokument WWW nie jest ograniczony do jednego formatu, protokołu i aplikacji klienckiej, ale może być dowolnym zasobem związanym z Internetem dostarczanym za pomocą protokołu i w formacie, które są uznanymi standardami do dowolnej aplikacji wykorzystującej łącze internetowe. Takie uogólniające pojęcie dokumentu WWW jest zgodne z definicją 2.1.8.

Z definicji 2.1.11 i 2.1.12 wynika więc, że pojęcie „strona WWW" jest hiponimem pojęcia „dokument WWW". Patrząc jednak z perspektywy blogów każdy post[13] spełnia wymagania „dokumentu WWW". Z kolei blog lub dowolny jego widok zawierający posty jest stroną WWW. Wynika stąd, że „dokument WWW" jest meronimem „strony WWW". Ten związek jest zgodny z rekurencyjną naturą dokumentów WWW opisanych w definicji 2.1.7.

2.1.2 Serwis WWW

Ze względu na intensywną ostatnio komercjalizację Internetu bardzo często słyszy się frazę: „[...] więcej informacji na stronie: www. ... [...]". Tak naprawdę nie chodzi tu o stronę WWW, lecz o serwis internetowy.

Definicja 2.1.13. Strona główna (ang. *Host Page)*

„Strona WWW identyfikowana przez URI zawierający komponent <authority>, ale taka, w której komponent <path> jest pusty lub po prostu składa się z pojedynczego „/".

Przykłady: Strony WWW identyfikowane przez http://www.w3.org i http://www.cern.ch są stronami głównymi." - (Lavoie i Nielsen 1999)

Definicja 2.1.14. Serwis internetowy, serwis WWW, witryna internetowa (ang. *Web site*)

„Kolekcja wzajemnie powiązanych (ang. *interlinked*) stron WWW, zawierająca stronę główną, umieszczoną w tej samej lokalizacji sieciowej. „Wzajemnie powiązanych" jest rozumiane w ten sposób, że do każdej strony WWW, będącej składową serwisu internetowego, można dotrzeć poprzez sekwencję odwołań począwszy od strony głównej serwisu; przechodząc zero, jedną lub więcej stron WWW umieszczonych w tym samym serwisie; i kończąc na rozważanej stronie.

Przykłady: Strona zawierająca artykuł "Thought Paper on Automatic Recharacterization" jest częścią serwisu internetowego W3C, ponieważ spełnia dwie własności wspomniane wyżej. Pierwsza, jest umieszczona w tej samej lokalizacji sieciowej , co strona główna W3C, http://www.w3.org. Druga, możemy rozpocząć na stronie głównej W3C (http://www.w3.org) i podążać sekwencją wewnętrznych powiązań (ang. *links*), kończąc na artykule, konkretnie:

1. z http://www.w3.org do http://www.w3.org/WCA/, i
2. z http://www.w3.org/WCA/ do http://www.w3.org/WCA/1998/12/aut_char.html

Uwagi: Nierzadko serwisy internetowe są powielane, czy odzwierciedlane, na wielu fizycznych maszynach (np. dla celów balansowania obciążenia). Zwykle jest to nieistotne dla klienta (czy użytkownika) która maszyna jest używana aby dotrzeć do serwisu internetowego. W tym przypadku, pomocne może być rozważenie tej kolekcji

13 Post – zwyczajowa nazwa wpisu w blogu

„fizycznych” serwisów WWW, umieszczonych na różnych maszynach, jako jeden „logiczny” serwis WWW. Jest to możliwe w przypadku, gdy jedna nazwa domenowa jest przypisana każdej z maszyn; logiczny serwis WWW może być wtedy identyfikowany za pomocą unikatowej nazwy domenowej. Jeśli nie ma unikatowej nazwy domenowej, którą można przydzielić kolekcji powielonych serwisów, uznajemy każdą fizyczną maszynę jako osobny serwis WWW.” - (Lavoie i Nielsen 1999)

Błędne nazywanie serwisu WWW stroną WWW wynika z problemu rozróżnienia tłumaczeń na język polski angielskich słów „page” i „site”. „Strona”, która dla marketingu nie ma większego znaczenia strategicznego, jest zręczniejszym pojęciem niż „serwis”. Poza tym, w biznesie, pojęcie „serwis” może oznaczać wiele innych usług nie związanych z WWW. Dlatego „strona” wprowadzana jest jako pojęcie potoczne oznaczające „serwis WWW”. Jest to wygodne do momentu, gdy trzeba rozróżnić „serwis WWW” od jego składowych „stron WWW”.

Definicja 2.1.15. Nadserwis (ang. *supersite*)

„Pojedynczy, logiczny serwis WWW, który rozmieszczony jest w wielu lokalizacjach sieciowych, ale z zamierzeniem, aby przeglądać go jako pojedynczy serwis WWW. Dla użytkownika przezroczysty jest fakt rozproszenia serwisu po wielu lokalizacjach. Jedna strona główna dotyczy całego nadserwisu.” - (Lavoie i Nielsen 1999)

Definicja 2.1.16. Podserwis (ang. *subsite*)

„Grupa (ang. *cluster*) stron WWW zawarta w serwisie WWW, który utrzymywany jest przez innego wydawcę niż wydawca nadrzędnego (ang. *parent*) serwisu WWW lub głównego (ang. *host*) serwisu. Wydawca podstrony sprawuje kontrolę redakcyjną nad stronami WWW tworzących podstronę, czasami ograniczoną ogólnymi wytycznymi nałożonymi przez wydawcę strony głównej.” - (Lavoie i Nielsen 1999)

Grupowanie stron w serwisy WWW jest najbardziej intuicyjnym i najpowszechniejszym sposobem tworzenia kolekcji WWW. Podobnie, jak w przypadku dokumentu WWW, natura serwisu WWW również może być rekurencyjna. Definicje 2.1.15 i 2.1.16 określają nie tyle nad- i podzbiory stron WWW, co zbiory zbiorów, czyli serwisy składające się z mniejszych serwisów WWW.

Uzasadnieniem przytoczonych kilkunastu definicji dla jedynie kilku pojęć niech będzie fakt, że pojęcia te nie tylko mają interpretację w różnych (choć jednorodnych, jak zbiór rekomendacji W3C) kontekstach, ale również mogą mieć zupełnie różną semantykę. Na przykład pojęcie „węzła” w definicji 2.1.3 użyte jest w zupełnie innym znaczeniu niż w definicji 2.1.5. Aby rozróżnić te znaczenia, trzeba je uściślić dodając do pojęcia przydawkę, tj.: „węzeł drzewa dokumentu”, „węzeł sieci dokumentów” i „węzeł sieci komputerowej (maszyn)”.

2.2 Link

W kilku definicjach z poprzedniego podrozdziału pojawia się odniesienie do hipertekstu i odsyłaczy hipertekstowych. Odsyłacze hipertekstowe zwane też hiperłączami, czy krótko: linkami, grają kluczową rolę w budowaniu struktury WWW oraz algorytmach wyszukiwania w WWW. Jednak jako pojęcie bardzo popularne - używane w różnych kontekstach: zarówno naukowych, technicznych, jak i w życiu codziennym - może być interpretowane rozbieżnie. Dodatkowym aspektem wpływającym na niejednoznaczność pojęcia „link” wpływa stały rozwój technologiczny oraz coraz nowsze zastosowania linków. Przyjrzyjmy się definicjom linków tworzonych w różnych okresach istnienia WWW i w różnych kontekstach ich zastosowania.

Definicja 2.2.1. Kotwica

„Obszar w treści węzła, który jest źródłem lub celem linka. Kotwica może być całą treścią

węzła. Zwykle, kliknięcie myszą na obszar kotwicy powoduje przejście przez link, powodujące pozostawienie kotwicy na przeciwnym końcu wyświetlonego linka. Kotwice mają tendencje do bycia podświetlonymi w specjalny sposób (zawsze, bądź gdy mysz znajduje się nad nimi) lub do bycia reprezentowanymi przez specjalny symbol. Kotwica może i często rzeczywiście odpowiada całemu węzłowi. (znana czasami jako "*span*", "*region*", "*button*", or "*extent*").” - (Girschweiler 1995)

Użyte tu pojęcie „węzła” odnosi się do definicji 2.1.5.

Definicja 2.2.2. Link

„Związek pomiędzy dwiema kotwicami, zawartymi w tej samej lub różnych bazach danych. [...]” - (Girschweiler 1995)

Użyte tu pojęcie „baza danych” odnosi się do definicji 2.3.9. Definicja 2.2.2 wprowadza jeszcze, przez referencje do innych definicji, podział linków na **wewnętrzne**, gdy obie kotwice linka są w jednej bazie danych i **zewnętrzne**, gdy link ma kotwice w różnych bazach. W odniesieniu do statycznych stron WWW, nie używających techniki AJAX[14], można by odnieść podział na linki wewnętrzne i zewnętrzne do faktu przeładowania strony w przeglądarce. Linki zewnętrze powodują załadowanie nowej strony WWW, natomiast linki wewnętrzne – ponieważ odnoszą się do tej samej strony – powodują co najwyżej przewinięcie strony do wskazywanego przez link fragmentu.

Definicja 2.2.3. Link

„Referencja z jednego dokumentu do drugiego (zewnętrzny link) lub z jednej lokalizacji w tym samym dokumencie do drugiej (zewnętrzny link), przez którą można przechodzić wydajnie używając komputera. Jednostka łącząca w hipertekście.” - (Tim Berners-Lee, Fischetti, i Dertouzos 1999)

Definicje 2.2.2 i 2.2.3 są relatywnie stare i dotyczą początkowego okresu istnienia WWW. Zresztą definicja 2.2.3 jest autorstwa samego twórcy idei WWW. Berners-Lee wzorował się na koncepcji hipertekstu stworzonej przez Teda Nelsona, na potrzeby - rozwijanego w latach sześćdziesiątych – projektu Xanadu, opisanej szczegółowo w Nelson 1982) [15].

W połowie lat dziewięćdziesiątych, wraz z powstaniem języka XML, znaczenie pojęcia link w odniesieniu do WWW znacznie się rozszerzyło. Link już nie koniecznie musiał oznaczać interaktywny odsyłacz hipertekstowy. Specyfikacja HTML 4.01 (Raggett, Le Hors, i I. Jacobs 1999) dzieli linki na te umieszczane w nagłówku dokumentu (między znacznikami `<head>`) i te umieszczane w ciele dokumentu (między znacznikami `<body>`). Te drugie są tradycyjnymi odsyłaczami hipertekstowymi, na które można kliknąć (lub aktywować w inny sposób – np.

14 AJAX (ang. *asynchronous JavaScript and XML*) to termin ukuty w Garrett 2005), będący krótką nazwą dla “Asynchronous JavaScript+CSS+DOM+XMLHttpRequest”. AJAX nie oznacza konkretnej technologii, ale podejście do architektury aplikacji WWW. Asynchroniczność w tej koncepcji oznacza możliwość rezygnacji ze stałego cyklu interakcji użytkownika z przeglądarką WWW: 1. załaduj stronę, 2. czekaj na interakcję użytkownika, 3. przeładuj/załaduj nową stronę. Dzięki użyciu obiektów *XMLHttpRequest* możliwe jest przeładowanie pojedynczych fragmentów strony, co wpływa znacząco na poprawę efektywności i funkcjonalności strony. Samo pojęcie AJAX zostało ukute w 2005 roku, gdy popularne zaczęły być aplikacje WWW w stylu *Web 2.0*. Jednak sama idea zwiększania interakcyjności strony przez potrzeby jej przeładowywania znana była dziesięć lat wcześniej. Być może dlatego autor definicji, która pochodzi z połowy lat dziewięćdziesiątych, nie rozróżnia linków wewnętrznych i zewnętrznych względem faktu przeładowania strony.

15 Chociaż koncepcji linka można doszukać się już w urządzeniu memex z 1945 roku. W Bush 1989) Vannevar Bush opisał maszynę, która była podłączona do biblioteki i potrafiła wyświetlać z niej filmy i książki. Sama maszyna potrafiła też tworzyć linki na mikrofilmie. Ponieważ linki nie były odwołaniami jedynie między tekstami, ale głównie między filmami - można by powiedzieć, że memex był nie tylko pierwszą, analogową maszyną hipertekstową, ale hipermedialną.

klawiaturą, głosem). Ich początkiem jest kotwica (znacznik <a>) ze zdefiniowanym atrybutem href, a końcem kotwica lub dowolny inny element, którego nazwa lub identyfikator odpowiadają wartości atrybutu href zdefiniowanego w znaczniku początkowym linka. Z kolei linki definiowane w nagłówku dokumentu nie są (nie muszą) być widoczne dla użytkownika. Jednym z powodów braku wizualizacji takiego linka, jest fakt, że początkiem linka jest cały dokument, w którego nagłówku umieścimy znacznik <link>. Końcem linka, jest natomiast – tak jak w przypadku znacznika <a> – zasób, na który wskazuje atrybut href. Innym powodem, przez który taki link nie jest prezentowany użytkownikowi bezpośrednio, jest to, że linki nagłówkowe nie mają treści. Definiując link za pomocą znacznika <a> jego treść umieszcza się pomiędzy znacznikiem otwierającym , a zamykającym . Treścią jest umieszczony między znacznikami tekst, grafika lub inny element, który użytkownik widzi jako początek hiperłącza. Natomiast w przypadku linka nagłówkowego taka sytuacja nie jest możliwa, ponieważ znacznik <link> nie posiada znacznika zamykającego. Jest znacznikiem jednoelementowym, więc brak elementów, między które można by wstawić treść. Nie znaczy to jednak, że linki nagłówkowe mogą reprezentować tylko jeden typ powiązań. Wręcz przeciwnie: linki zdefiniowane znacznikiem <link> dostarczają o wiele więcej funkcjonalności, niż znaczniki <a>, które służą głównie jako odsyłacze hipertekstowe. Można by nawet linki tworzone znacznikami <a> - dla odróżnienia od linków definiowanych przez <link> - nazywać **hiperłączami** (próbuje to zrobić definicja 2.2.9), gdyby nie fakt, że obydwa znaczniki zawierają atrybut href. A jak napisał sam Berners-Lee w Tim Berners-Lee 1995): "href" oznacza "referencję hipertekstową".

Definicja 2.2.4. Link

„Link wyraża jednym lub więcej (jawnych lub ukrytych) związków pomiędzy dwoma lub więcej zasobami.

Uwaga: Typ związku może opisywać związki jak „autorstwa", „osadzony", itd. Typy mogą same być identyfikowane przez URI, jak na przykład w przypadku RDF." - (Lavoie i Nielsen 1999)

Drugą fundamentalną zmianą w uogólnieniu pojęcia „link" - poza brakiem konieczności jego wizualizacji - jest to, że link nie koniecznie musi łączyć dokładnie dwa elementy. Obie zmiany poszerzające dotychczasowe pojęcie linka wyszczególnione są w definicji 2.2.4. Dodatkowo pojęcie „referencji" zostało tu uogólnione do zbioru związków. Pojawia się też kwestia typów związków, które mogą być zdefiniowane przez URI. A URI to uogólnienie URL, którym definiuje się na przykład kotwicę końcową hiperłącza.

2.2.1 Standardy ISO

Link jako pojęcie najczęściej utożsamiane z obiektem opisanym w języku znaczników wywodzi się z koncepcji ID-IDREF wykorzystywanej np. w SGML[16]. Koncepcja ID-IDREF jest konstrukcją często wykorzystywaną do tworzenia grafowych struktur przy użyciu znaczników. Pojawiła się wraz z koniecznością opisania struktur bardziej skomplikowanych niż hierarchia. ID i IDREF odnoszą się do kotwic referencji, czyli obiektów na obu końcach linka. Rozwinięcie ID-IDREF, zwanej też referencją krzyżową (ang. *cross-reference*) to hiperłącze, opisane w standardzie ISO dotyczącym języka HyTime (Goldfarb et al. 1997).

Definicja 2.2.5. Hiperłącze

„Hiperłącze jest związkiem o zadanym typie pomiędzy dwoma lub więcej obiektami, z których każdy pełni jednoznaczną rolę w związku.

16 Standard Generalized Markup Language (ISO 8879:1986 SGML) jest standardem metajęzyka, którym zdefiniowane są popularne języki znaczników, jak HTML. Jest on również nadklasą dla popularnego języka XML.

Uwaga 209 Nie wszystkie związki są hiperłączami. Hierarchie (zawieranie), plany zdarzeń (związki koordynacyjne) i proste referencje krzyżowe (ID/IDREF) są innymi rodzajami związków, którym brakuje pewnych własności hiperłączy, a które same wprowadzają inne własności.

Uwaga 210 Termin „hiperłącze" jest używany zamiast nieprecyzyjnego terminu „link", aby uniknąć pomylenia go z cechą linków przetwarzających SGML. Jednakże, termin „link" może być używany z bardziej restrykcyjnie uszczegóławiającymi przymiotnikami, jak w „link hipertekstowy" lub bez uszczegóławiania, gdy kontekst jest jasny." - (Goldfarb et al. 1997)

W definicji 2.2.5, pomimo użycia terminu „hiperłącze", autorzy traktują pojęcie takiego linka dosyć szeroko, podobnie jak w definicji 2.2.4. Jednak uwaga 210 zaznacza, że tak zdefiniowane pojęcie hiperłącza nie obejmuje takich aspektów SGML'a, które np. w HTML są realizowane przez linki nagłówkowe (`<link>`). Uwaga 209 zauważa, że hiperłącze nie jest tak ogólnym pojęciem jak związkek. Myśl ta rozwinięta jest dalej, w uwadze 212 do specyfikacji HyTime'a. Mianowicie: o ile funkcjonalność hiperłączy można w pewnym stopniu osiągać używając atrybutów typu ID i IDREF, o tyle standaryzowany moduł hiperłączy ma dodatkowe zalety:

1. dostarcza zbiór formalizmów architektonicznych do opisu linków i zasad przechodzenia przez nie;
2. dzięki standaryzacji mechanizmów przechodzenia i rozwijania linków możliwe jest stworzenie silnika hiperłączy, z możliwością wykorzystania go w różnych aplikacjach;
3. do czego nawiązuje definicja 2.2.9: nie wszystkie IDREF są przeznaczone do interakcji z użytkownikiem lub innego warunkowego dostępu; niektóre jedynie łączą informacje potrzebne do interpretacji lub przetwarzania dokumentu. Mając dostępne formalizmy architektoniczne hiperłączy możliwe będzie rozróżnienie prawdziwych hiperłączy od innych zastosowań IDREF oraz sprecyzowanie reguł przechodzenia przez linki podczas interakcji.

Dalej standard ISO precyzuje, że hiperłącze, jako obiekt abstrakcyjny posiada 3 główne własności:

1. typ linka,
2. obiekty, które są kotwicami,
3. role kotwic, czyli semantyka ról, jakie odgrywają kotwice w związku reprezentowanym przez hiperłącze.

Uwaga 213 podaje przykład, który wyjaśnia w jaki sposób hiperłącze może być związkiem dotyczącym więcej niż 2 obiektów. Przykładowa związek typu „zatrudnienie" może definiować dwie role kotwic: „ „pracownik" i pracodawca". Instancja linka zatrudnienia posiadałaby dwie kotwice składające się z pojedynczych obiektów: osoby - pełniącej rolę pracownika i firmy (lub osoby), która pełni rolę pracodawcy. Związek pomiędzy pracownikiem i wieloma pracodawcami można reprezentować pozwalając, aby kotwica pracodawcy była listą. Podobnie związek wielu pracowników do jednego pracodawcy może być zrealizowana, gdy kotwica pracownika będzie listą. Gdy obie kotwice będą listami hiperłącze będzie reprezentowało związek wielu pracowników do wielu pracodawców. Jak zostaje to później zauważone w uwadze 217: o ile liczba kotwic hiperłącza jest ustalona, o tyle liczba obiektów, które są linkowane jest dowolna. Co ciekawe zawartość hiperłącza może być kotwicą, ale nie musi – w zależności od tego, czy hiperłącze definiuje dla niej rolę semantyczną czy nie. Role semantyczne mają przypisane zarówno kotwice, jak i typy linka. Typy linka mogą być również podtypami, co umożliwia stworzenie hierarchicznej struktury typów.

W celu umożliwienia dostępu do kotwic, ich adresy są rozwijane na żądanie aplikacji. W niektórych zastosowaniach, zwłaszcza interaktywnych, hiperłącza są używane do reprezentowania punktów decyzyjnych, które wpływają na dostęp do części hiperdokumentu, reprezentowanych przez kotwice hiperłącza. Takie zastosowania wymagają selektywnego

rozwinięcia adresów kotwic, aby umożliwić do nich dostęp. Selektywne rozwinięcie wprowadzone jest ze względu na aspekt praktyczny. Ponieważ adresem kotwicy może być nie tylko ID czy lokalizacja w drzewie, ale również np. zapytanie, dlatego rozwijanie wszystkich adresów kotwic (w tym odpowiadanie na zapytania) przed prezentacją kotwic może być nieefektywne. Selektywny dostęp do kotwicy hiperłącza nazwany jest „**przechodzeniem**" (ang. *traversal*). W zastosowaniu można określić dozwolone przejścia w odniesieniu do każdej kotwicy za pomocą reguł przechodzenia linka.

Przechodzenie odbywa się pomiędzy kotwicami lub pomiędzy sąsiednimi elementami kotwicy-listy, ale nigdy do lub z samego linka (chyba, że link jest sam dla siebie kotwicą). Kotwica, która jest dostępna z zewnątrz linka i z której możliwe jest przejście przez link nazywana jest „**kotwicą początkową**". Klasyczny link dwukierunkowy to taki, który posiada dwie kotwice, z których każda jest kotwicą początkową.

Gdy link nie jest jedną ze swoich kotwic lub kotwicą początkową, nazywany jest **niezależnym**, ponieważ jest niezależny kontekstowo od swoich kotwic. W szczególności link może znajdować się w innym dokumencie niż jego kotwice. Z punktu widzenia kotwic ich linki niezależne są „niewidoczne" poza samym oddziaływaniem takich linków na kotwice. Tzn. mając link wiemy do jakich kotwic można przez niego przejść, ale mając daną kotwicę nie jesteśmy w stanie uzyskać informacji o linkach niezależnych, które dotyczą tej kotwicy.

Problem wykrywania linków niezależnych. Zakładając zdecentralizowany system kotwic, takie podejście „z linka widać jego kotwice, ale z kotwic nie widać jego linków niezależnych" wydaje się rozsądne. Autorzy standardu, choć nie wymagają, to jednak proponują, aby własności hiperłączy, włączając w to zawartość hiperłączy niezależnych, były dostępne na żądanie. Takie rozwiązanie jest w praktyce niemożliwe, ponieważ wymagałoby równoległego aktualizowania dokumentów. Prześledźmy to na przykładzie WWW: jeśli tworząc dokument WWW, wstawiam do niego hiperłącze, którego kotwicą końcową jest dokument innego autorstwa, to najczęściej nie mam możliwości zaznaczenia w docelowym dokumencie istnienia takiego hiperłącza. Oznacza to, że moje hiperłącze jest – z poziomu dokumentu, na który się powołuję – niewidoczne. I pomimo że mój link nie jest niezależny, bo jest jednocześnie kotwicą początkową, to jednak postulat, aby był on globalnie dostępny jest ze względów wydajnościowych niemożliwy do spełnienia. Zdecentralizowany model WWW powoduje, że praktycznie niemożliwe jest zaindeksowanie całej WWW, a tylko wtedy możliwe byłoby udzielenie informacji o wszystkich linkach dotyczących danej kotwicy. Na przykład wyszukiwarka Google umożliwia wsteczne wyszukiwanie kotwic początkowych, z których można przejść do interesującej nas kotwicy poprzez użycie operatora `link:` w zapytaniu. Jednak ze względu na zakres i aktualność indeksu Google informacja ta nie może być pełna.

Jako częściowe rozwiązanie tego problemu można potraktować propozycję z uwagi 222 w standardzie HyTime. Mianowicie: aplikacje mogą umożliwiać dostęp do kotwic poprzez dostęp do samego linka, nawet, gdy sam link nie jest kotwicą. Czyli aplikacja może założyć, że link jest zawsze jednocześnie jedną ze swoich kotwic. Ale tylko po to, aby możliwe było przejście do samego linka (i stamtąd dalej), a nie po to, aby określić związek (bo to by oznaczało, że obiektem w związku jest związek). To rzeczywiście pozwoliłoby na lepsze zorientowanie się w linkach zawierających konkretną kotwicę, ale ponownie, ze względu na praktyczny zasięg indeksu aplikacji (zakres uwzględnianych dokumentów) informacja taka nie zawsze będzie pełna.

Co prawda istnieją techniki, takie jak *pingback* i *trackback*, które pozwalają serwerom je obsługującym wykrywać linki wsteczne. Niestety nie są one jeszcze na tyle powszechne, aby mówić o takiej funkcjonalności globalnie.

Chcąc rozróżnić hiperłącza analogiczne do tych definiowanych w HTML 4.01 przez znaczniki

<link> od tych, które w HTML 4.01 są definiowane w <body>, autorzy standardu HyTime wprowadzają pojęcie „**hiperłącze kontekstowe**”. Gdy hiperłącze jest jedną ze swoich kotwic i jest kotwicą początkową nazywa się je „kontekstowym”, ponieważ taki link najpewniej pojawi się w dostępnym dla czytelnika kontekście, gdzie zwykle używany jest jako kotwica początkowa. Czyli hiperłącze kontekstowe według standardu ISO, to ten element, który wcześniej - w odniesieniu do HTML 4.01 – nazwaliśmy tradycyjnym odsyłaczem hipertekstowym.

Wracając do przejść przez linki, należy zwrócić uwagę na szczególny przypadek przejść, gdy kotwicami są listy. Kotwice-listy mogą mieć zdefiniowane przechodzenie pomiędzy elementami listy. Najczęściej są to przejścia do elementu poprzedniego lub następnego - zależnie od kierunku przechodzenia, ewentualnie w obie strony, gdy lista jest dwukierunkowa. W przypadku, gdy lista jest cykliczna (ang. *circular*) możliwe jest przejście między elementem pierwszym i ostatnim, zgodnie z kierunkiem listy.

Jak zauważają autorzy w uwadze 223 w przypadku takiej listy aplikacja może zaprezentować przyciski „poprzedni” i „następny” dotyczące elementów danej listy, jak również umożliwić przejście z każdego elementu listy do innej kotwicy linka. W szczególności, gdy dwie kotwice są odpowiadającymi sobie listami przejście z elementu jednej kotwicy-listy do drugiej zawsze odbywa się do odpowiedniego elementu drugiej kotwicy-listy. Ilustruje to podany w uwadze 224 przykład: związkek „pokrycia napastników przez obrońców” między dwoma drużynami sportowymi można przedstawić za pomocą kotwic, będących listami odpowiadającymi (ang. *corresponding lists*).

Bardzo ważnym elementem standardowego modułu hiperłączy w HyTime są reguły przechodzenia. Atrybutem, za pomocą którego definiuje się reguły jest `linktrav`. Aby jasno określić dziedzinę tego atrybutu, czyli wszystkie możliwe reguły przejść autorzy definiują najpierw pojęcia przejścia, przybycia i opuszczenia.

Definicja 2.2.6. Przejście (ang. *traversal*)

> „Ruch od jednej kotwicy hiperłącza do drugiej kotwicy tego samego hiperłącza. Przejście można sobie wyobrazić jako przesunięcie „przez” link, aby dostać się z jednej z jego kotwic do drugiej. Przejście zawsze odnosi się do pojedynczego linka, niezależnie od tego czy więcej linków używa tego samego obiektu jako kotwicy.” - (Goldfarb et al. 1997)

Definicja 2.2.7. Przybycie (ang. *arrival*)

> „Przybycie do kotwicy hiperłącza innymi środkami niż za pomocą przejścia przez rozważany link. Na przykład, w typowej *online*'owej prezentacji, przewijanie dokumentu do punktu, w którym pojawia się kotwica stanowiłoby przybycie do kotwicy. Przybycie zawsze odbywa się poza linkiem. W szczególności, przejście do kotwicy, która jest również kotwicą innego linka stanowiłoby przybycie do kotwicy w odniesieniu do tego drugiego linka.” - (Goldfarb et al. 1997)

Definicja 2.2.8. Opuszczenie (ang. *departure*)

> „Ruch z kotwicy linka do miejsca, które nie jest kotwicą tego samego linka. Na przykład, po przejściu do kotwicy, przewinięcie dokumentu gdzieś indziej stanowiłoby opuszczenie. Opuszczenie zawsze odnosi się do linka, użytego do przejścia do kotwicy. W szczególności, jeśli dwa linki dzielą jeden obiekt jako kotwicę, po przejściu do tej kotwicy jednym linkiem, przejście przez drugi link stanowiłoby opuszczenie w odniesieniu do pierwszego linka.” - (Goldfarb et al. 1997)

Wydaje się, że w ostatnim zdaniu definicji 2.2.8 brakuje „... o ile przejście drugim linkiem odbyłoby się do kotwicy nie będącej również kotwicą, do której możliwe jest przejście za pomocą pierwszego linka.” Analogicznie w ostatnim zdaniu definicji 2.2.7 można by oczekiwać, aby

kotwica, z której odbywa się przejście, nie była jednocześnie kotwicą drugiego linka. W praktyce brak tych uszczegółowień oznaczałby, że możliwe jest przybycie i opuszczenie, które są w rzeczywistości przejściami między dwiema kotwicami jednego linka. To by przeczyło twierdzeniu: „przybycie zawsze odbywa się poza linkiem". Jednak logicznie rzecz biorąc: nawet jeśli obie kotwice należą do obu linków i oboma linkami możliwe jest przejście z jednej do drugiej, to nazwanie przemieszczenia się między kotwicami albo przejściem, albo przybyciem i opuszczeniem determinuje semantykę przemieszczenia się. Choć w praktyce samo przemieszczenie się między dwiema kotwicami wspólnymi dla dwóch linków sprowadza się do tego samego, to jeśli rozpatrujemy ten sam związek, co przy poprzednim przemieszczeniu, bieżące przemieszczenie będzie oznaczało przejście, a jeśli związek drugiego linka – opuszczenie kotwicy pierwszego i przybycie do kotwicy drugiego.

Mając pojęcia przejścia, przybycia i opuszczenia standard wprowadza 6 podstawowych opcji przechodzenia:

1. Przejście po przybyciu („E"). Przybycie odbywa się do kotwicy początkowej. Opcja „E" nie zakłada możliwości opuszczenia przy powrocie lub przejściu.
2. Powrót („R"). Możliwość przejścia jedynie do kotwicy, której odbyło się przejście. W uwadze 227 jako przykład powrotu prezentacji powrotu podano wyskakujące okienko modalne, które wymaga zamknięcia przez jakimkolwiek innym przejściem.
3. Wewnętrzne („I"). Po przybyciu takiej kotwicy możliwe jest przejście każdej innej kotwicy linka.
4. Opuszczenie („D"). Po przybyciu takiej kotwicy możliwe jest opuszczenie kotwicy i linka.
5. Brak dalszego przechodzenia („N"). Po przejściu do takiej kotwicy dalsze przechodzenie nie jest możliwe.
6. Przechodzenie zabronione („P"). Przejście do kotwicy jest zabronione z każdej innej kotwicy linka.

Kombinacje powyższych opcji stanowią dziedzinę atrybutu `linktrav`. Słowa kluczowe możliwe do użycia w atrybucie `linktrav` to:

- I – przejście po przejściu z kotwicy tego samego linka
- R – powrót po przejściu
- D – opuszczenie po przejściu
- A – dowolne przejście lub opuszczenie (ekwiwalent EID)
- N – brak przejścia po przejściu z kotwicy tego samego linka
- P – przejście do tej kotwicy zabronione
- ID – przejście lub opuszczenie
- RD – powrót lub opuszczenie
- EI – przejście po przybyciu, przejście po przejściu z kotwicy tego samego linka
- ER – przejście po przybyciu, powrót po przejściu z kotwicy tego samego linka
- ED – przejście po przybyciu, opuszczenie po przejściu z kotwicy tego samego linka
- EN– przejście po przybyciu, brak przejść po przejściu z kotwicy tego samego linka
- EP – przejście po przybyciu, przejście do kotwicy zabronione (czyli zakaz powrotu)
- ERD – przejście po przybyciu, powrót, opuszczenie (różni się od A tylko wtedy, gdy istnieją więcej niż dwie kotwice w hiperłączu

Podobne opcje zdefiniowane zostały dla atrybutu `listtrav`, który dotyczy przechodzenia list i list odpowiadających. Dodatkowo standard HyTime rozróżnia pięć form architektonicznych linków:

1. Hiperłącze (ang. *hyperlink*) jest najbardziej uniwersalne. Może zawierać dowolną liczbę kotwic i może być od nich całkowicie niezależny.
2. Link kontekstowy (ang. *contextual link*) reprezentuje prostą referencję krzyżową.
3. Link grupujący (ang. *aggregation link*) reprezentuje związki grupowe.
4. Link zmienny (ang. *variable link*) reprezentuje hiperłącza, w których role kotwic mogą

różnić się między instancjami linków tego samego typu.

5. Link niezależny (ang. *independent link*) zapewnia alternatywną składnię dla hiperłączy, w których adresy kotwic są podane przy użyciu prostego atrybutu IDREF, zamiast osobnego atrybutu dla każdej roli kotwicy.

Szczegółowe rozpatrywanie różnic pomiędzy tymi formami jednak wydaje się, w kontekście tej pracy, nieuzasadnione, ponieważ są to różnice głównie syntaktyczne. Sama idea hiperłączy jest podobna do tej ze standardów W3C. Czymś wychodzącym poza pojęcia znane z HTML'a mogą być kotwice-listy i dotyczące ich linki grupujące. Tego typu związki standaryzowane są przez W3C w rekomendacji XLink z 2001 roku, która powołuje się na HyTime jako standard, który obok HTML miał duży wpływ na projektowanie XLink.

2.2.2 XLink

XLink (Orchard, Maler, i DeRose 2001) to język tworzenia linków dla dokumentów XML (ang. *XML Linking Language*). Ta rekomendacja W3C standaryzuje umieszczanie w dokumentach XML'owych elementów, w celu stworzenia i opisania powiązań (ang. *links*) pomiędzy zasobami. Pomimo że standard dotyczy dokumentów XML, jego stosowanie może być powszechne w WWW, ponieważ obecnie głównym i zalecanym językiem dokumentów WWW jest XHTML[17], który spełnia wymogi języka XML. Co prawda w połowie 2008 roku W3C opublikowało szkic roboczy HTML 5[18], jednak jego specyfikacja zachęca do zachowania zgodności zarówno z formatami inspirowanymi przez SGML, jak i tymi opartymi na składni XML.

Definicja 2.2.9. Hiperłącze

„Hiperłącze to link przeznaczony przede wszystkim do prezentacji dla człowieka.” - (Orchard, Maler, i DeRose 2001)

Definicja 2.2.10. Link

„Link w języku XLink to jawny związek pomiędzy zasobami lub częściami zasobów.” - (Orchard, Maler, i DeRose 2001)

Definicja 2.2.11. Bazy linków (ang. *linkbases*)

„Dokumenty zawierające kolekcje linków przychodzących i zewnętrznych (ang. *third party*) nazywane są bazami danych linków lub bazami linków.” - (Orchard, Maler, i DeRose 2001)

W odróżnieniu od HyTime, gdzie termin hiperłącze był rozumiany bardzo szeroko, w specyfikacja XLink wprowadza definicję 2.2.9. Samo pojęcie linka (definicja 2.2.10) jest jednak tak ogólne jak hiperłącza w HyTime (definicja 2.2.5). Możliwość istnienia linków niezależnie od ich kotwic, czyli wprowadzonych przez HyTime linków niezależnych wykorzystywana jest do

17 XHTML (ang. *Extensible Hypertext Markup Language*) to język znaczników posiadający ten sam poziom ekspresji co HTML, ale jednocześnie zgodny ze składnią XML. XHTML 1.0 jest „ponownym sformułowaniem trzech typów dokumentów HTML 4 jako aplikacji XML 1.0” - (Steven Pemberton 2002). Trzy typy dokumentów XHTML o których mowa w cytacie to ścisły (ang. *strict*), przejściowy (ang *transitional*), ramkowy (ang. *frameset*). Wersja 1.0 XHTML została rekomendacją W3C w 2000 roku. Wersja 1.1 – rok później.

18 HTML 5 (ang. *Hypertext Markup Language*) to język znaczników, będący kontynuacją HTML 4.0.1. Był odpowiedzią na specyfikację XHTML 2.0 (Axelsson et al. 2006). Jak podaje specyfikacja HTML 5: „XHTML2 definiuje nowe słownictwo HTML z lepszym wsparciem dla hiperłączy, zawartości multimedialnej, przypisów edycyjnych, bogatych metadanych, deklaratywnych interaktywnych formularzy i opis semantyki utworów literackich, jak wiersz czy publikacja naukowa. Jednak brakuje mu elementów do wyrażania semantyki wielu treści WWW niebędących typem dokumentu. Na przykład: fora, aukcje, wyszukiwarki, sklepy internetowe itp. nie wpasowują się w metaforę dokumentu i nie są ujęte w HTML2.” - (Hyatt i Hickson 2008)

tworzenia baz linków (definicja 2.2.11).

Aby ułatwić używanie linków i zrozumienie dosyć już złożonej ich semantyki XLink dzieli je na 2 klasy: Linki proste (definicja 2.2.12) i rozszerzone (definicja 2.2.14).

Definicja 2.2.12. Linki proste (ang. *simple links*)

„Linki proste oferują skrótową składnię dla powszechnego rodzaju linka, linka wychodzącego z dokładnie dwoma uczestniczącymi zasobami (kategoria, do której wpadają HTMLowe znaczniki A i IMG). Ponieważ linki proste oferują mniejszą funkcjonalność niż linki rozszerzone, nie mają specjalnej wewnętrznej struktury. Pomimo że linki proste są koncepcyjnie podzbiorem linków rozszerzonych, różnią się od nich składnią. Na przykład, aby zamienić link prosty w link rozszerzony potrzeba kilka zmian strukturalnych." - (Orchard, Maler, i DeRose 2001)

Definicja 2.2.13. Łuk

"Informację o tym jak przechodzić parę zasobów, wliczając w to informację o kierunku przejścia i możliwym zachowaniu się aplikacji, nazywa się łukiem." - (Orchard, Maler, i DeRose 2001)

Definicja 2.2.14. Linki rozszerzone (ang. *extended links*)

„Linki rozszerzone oferują pełną funkcjonalność XLink, taką jak łuki przychodzące i zewnętrzne (ang. *third party*), jak również linki mające dowolną liczbę uczestniczących zasobów. W rezultacie, ich składania może być bardzo złożona, wliczając elementy wskazujące na zdalne zasoby, elementy zawierające lokalne zasoby, elementy specyfikujące reguły przejść i elementy specyfikujące czytelne dla ludzi tytuły zasobów i łuków. XLink definiuje definiuje sposób, w który można nadać linkowi rozszerzonemu specjalną semantykę pozwalającą odnajdywać bazy linków; używany w ten sposób, link rozszerzony pomaga aplikacji przetwarzać inne linki." - (Orchard, Maler, i DeRose 2001)

Linki proste trzymają się koncepcji prostego ID/IDREF, czyli przyporządkowaniu linkowi dokładnie dwóch kotwic, jak w linku kontekstowym z HyTime. Aby nie zniechęcać projektantów do używania standardu nawet w prostych przypadkach linki proste mają uproszczoną składnię pozwalającą pominąć atrybuty, które w wielu prostych przypadkach nie są potrzebne. Pomimo różnej składni definicja 2.2.12 wskazuje, że koncepcyjnie linki proste są podzbiorem linków rozszerzonych. W zasadzie można powiedzieć, że link prosty został zaprojektowany, aby odzwierciedlić prosty związek referencji krzyżowej, czyli łuku zdefiniowanego w definicji 2.2.13. Mając tylko linki proste możemy przedstawić opisywane przez nie związki za pomocą multigrafu skierowanego (definicja 2.4.3). Linki rozszerzone, dopuszczając więcej niż dwie kotwice, muszą umożliwić zdefiniowanie więcej niż jednego przejścia, z których każde przejście jest łukiem. Dlatego związków opisywanych przez linki rozszerzone nie da się przedstawić za pomocą grafu czy multigrafu, ale można to zrobić używając ogólniejszej konstrukcji, jaką jest hipergraf (definicja 2.4.4). Modelowanie związków linków zostanie jeszcze poruszone w kolejnych rozdziałach.

XLink ma szansę stać się powszechnie stosowanych standardem, ze względu na fakt, że oferuje bardzo duże możliwości opisania semantyki związku, co jest bardzo istotne, zwłaszcza w kontekście sieci semantycznej. Chociaż, jak podkreślają autorzy (Ian Jacobs i Walsh 2004): „XLink nie jest ani jedynym projektem linkowania zaproponowanym dla XML'a, ani nie jest promowany jako projekt dobry dla każdego zastosowania." Na przykład jeśli chcemy użyć linka w aplikacji sterownej głosem, chcąc trzymać się standardów należy to zrobić przy użyciu VoiceXML[19]. Co prawda idea takiego linka jest nieco odmienna od linków w XLink.

19 VoiceXML to oparty na składni XML standard W3C służący do definiowania interaktywnych dialogów głosowych

Definicja 2.2.15. Link

> "Zbiór gramatyk, które w momencie gdy pasują do tego co użytkownik mówi lub naciska, albo przenoszą do nowego dialogu lub dokumentu, albo rzucają zdarzenie w elemencie bieżącego formularza." - (McGlashan et al. 2004)

Definicja 2.2.15 różni się koncepcyjnie od definicji 2.2.12 i 2.2.14, ponieważ dotyczy dokumentów dla aplikacji głosowych. W związku z tym trudniej mówić tu o idei hipertekstu, skoro użytkownik takiego linka najczęściej nie widzi tekstu. Jednak nadal mamy do czynienia z hipermediami, które również w kontekście XLinka są możliwe do zastosowania.

Działając w duchu modularyzacji języka XHTML w wersji 1.1 (Shane McCarron i Altheim 2001) oraz chcąc „rozszerzyć możliwości zastosowania XLink do szerszej klasy języków niż tych ograniczonych przez styl składni dopuszczany przez XLink" (Masayasu Ishikawa i Steven Pemberton 2002) W3C opublikowało szkic roboczy HLink. HLink najpewniej został porzucony, ponieważ od 2002 roku nie pojawiła się żadna nowa wersja. Natomiast druga edycja XHTML 1.1 (na razie w wersji szkicu roboczego z lutego 2007 - (Shane McCarron i Masayasu Ishikawa 2007)) całkowicie opiera się na szkicu roboczym modularyzacji dla dokumentów XML'owych z roku 2006, w której można znaleźć zarówno moduł linków, jak i moduł hipertekstu. I to wydaje się być właściwym kierunkiem rozwoju, ponieważ w październiku 2008 roku ta modularyzacja (Masayasu Ishikawa et al. 2008) została rekomendacją W3C, a oparta na niej druga edycja HTML 1.1 spodziewana była też jeszcze w 2008 roku.

2.2.3 Typy linków

Poza tym, że w XHTML 1.1, jak w każdym języku używającym modułu link według standardu (Masayasu Ishikawa et al. 2008) możliwe jest definiowanie własnych typów linków, standard modułu podaje również listę znanych (ang. *recognized*) typów linków. Lista ta nie różni się niczym od typów linków zdefiniowanych w HTML 4.0 (Raggett, Le Hors, i I. Jacobs 1999), przy czym należy pamiętać, że HTML 4 nie dopuszczał jeszcze możliwości definiowania własnych typów linków. Znane typy linków to:

- `Alternate` – oznacza wersję zastępczą dokumentu, w którym link występuje. Użyty razem z atrybutem `hreflang` oznacza przetłumaczoną wersję dokumentu. Użyty razem z atrybutem `media` oznacza wersję na inne media (np. drukarkę).
- `Stylesheet` – odnosi się do arkusza stylów używanego w module stylów. Używa się go razem z typem `alternative` do wybieralnych przez użytkownika alternatywnych arkuszy stylów.
- `Start` – odnosi się do pierwszego dokumentu w kolekcji dokumentów. Ten typ linka przekazuje wyszukiwarce który dokument jest uważany przez autora za punkt startowy w kolekcji.
- `Next` – odnosi się to następnego dokumentu w liniowej sekwencji dokumentów. Agenty użytkownika mogą buforować następny dokument, aby zmniejszyć postrzegany czas ładowania.
- `Prev` – odnosi się do poprzedniego dokumentu w uporządkowanej serii dokumentów. Niektóre agenty użytkownika obsługują również synonim „`previous`".
- `Contents` – odnosi się do dokumentu dostarczającego spis treści. Niektóre agenty

między człowiekiem i komputerem. VoiceXML pozwala na tworzenie aplikacji, w sposób analogiczny do języka HTML pozwalającego tworzyć aplikacje wizualne. Tak jak dokumenty HTML są używane w przeglądarkach wizualnych, tak dokumenty napisane w VoiceXML są używane w przeglądarkach głosowych. Przeglądarki głosowe najczęściej zintegrowane są z siecią telefoniczną, dzięki czemu użytkownik może nawigować po aplikacji VoiceXML przez telefon, za pomocą komend głosowych.

użytkownika obsługują również synonim `ToC` (od "*Table of Contents*").

- `Index` – odnosi się do dokumentu dostarczającego indeks dla bieżącego dokumentu.
- `Glossary` – odnosi się do dokumentu dostarczającego słowniczek pojęć, które dotyczą bieżącego dokumentu.
- `Copyright` – odnosi się do oświadczenia o prawach autorskich dotyczących bieżącego dokumentu.
- `Chapter` – odnosi się do dokumentu służącego jako rozdział w kolekcji dokumentów.
- `Section` – odnosi się do dokumentu służącego jako sekcja w kolekcji dokumentów.
- `Subsection` – odnosi się do dokumentu służącego jako podsekcja w kolekcji dokumentów.
- `Appendix` – odnosi się do dokumentu służącego jako załącznik w kolekcji dokumentów.
- `Help` – odnosi się do dokumentu oferującego pomoc (więcej informacji, linki do innych źródeł informacji, itd.).
- `Bookmark` – odnosi się do zakładki. Zakładka to link do kluczowej pozycji w rozszerzonym dokumencie. Atrybut `title` może być używany, na przykład, jako etykieta zakładki. Zauważ, że wiele zakładek może być zdefiniowanych w każdym dokumencie.

Typy linków pozwalają definiować role linków. W znacznikach a i `link` role opisuje się za pomocą atrybutów `rel` i `rev`. Atrybuty te pełnią role komplementarne. Na przykład: jeśli w `rozdzial3.html` wstawimy znacznik `<link rel="glossary" href="slowniczek.html" />`, to żeby mieć dostęp z dokumentu `slowniczek.html` z powrotem do `rozdzial3.html` musimy w `slowniczek.html` wstawić link odwrotny, ale opisujący ten sam związek, t.j.: `<link rev="glossary" href="rozdzial3.html" />`.

W XLink do przypisania semantyki linkowi służy atrybut `xlink:role`, natomiast semantykę każdego z przejść w linku można określić atrybutem `xlink:arcrole`. Tu standard nie zakłada żadnych predefiniowanych ról, a semantykę tych atrybutów można odczytywać następująco: `xlink:role` opisuje związek dotyczący wszystkich kotwic xlinka, natomiast `xlink:arcrole` mówi o konkretnym związku (uwzględniając jego kierunek) pomiędzy parą kotwic.

2.2.4 Trackback i Pingback

Wraz z wprowadzeniem serwisów blogowych WWW nabrała prędkości w rozrastaniu się, gdyż każdy mógł opublikować swoje treści w sieci i nie wymagało to znajomości HTML i CSS oraz działania serwera WWW. Każdy mógł w przyjazny sposób, nie wychodząc z przeglądarki nie tylko przeglądać WWW ale i ją współtworzyć. Wraz z tą rewolucją blogosfery[20] pojawiła się możliwość jednolitego dostępu do tych treści zarówno dla wyszukiwarek, jak i dla użytkowników. Poza formatem RSS[21], którą każdy portal blogów musiał obsługiwać, wyszukiwarki mogły skorzystać przy indeksowaniu z jednolitego (przynajmniej w ramach jednego serwisu) formatu metadanych. To znacznie ułatwiło indeksowanie i poprawiło efektywność wyszukiwania.

20 Blogosfera (ang. *blogosphere*) - termin określający wszystkie blogi jako społeczność lub sieć społeczną. Wiele blogów jest połączonych między sobą. Ich autorzy czytają inne blogi i zostawiają na nich komentarze, umieszczają do nich linki, odnoszą się też w treści wpisów. W ten sposób tak połączone blogi wytworzyły własną kulturę. Podczas gdy blogi pozostają wciąż formą publikacji, blogosfera jest zjawiskiem społecznym i jak inne podlegającym badaniom socjologicznym. Terminu "blogosfera" po raz pierwszy użył (w formie żartu) Brad L. Graham, 10 września 1999. Określenie to jest wiązane ze słowem "logosphere", który od gr. "logos" (słowo) oznacza "świat słów", "świat dyskusji". Jego powszechne użycie datuje się od 2002 - (Contributors 2008)

21 RSS (ang. *Really Simple Syndication*) - jest to rodzina formatów sieciowych, opartych na języku XML służących do publikacji często zmieniających się treści, takich jak wpisy blogów, wiadomości. Dokument RSS, często zwany "kanałem", zazwyczaj zawiera streszczoną formę wiadomości ze skojarzonej strony WWW lub jej pełny tekst. RSS umożliwia użytkownikom automatyczne bycie na bieżąco z treścią ulubionych serwisów sieciowych - (Contributors 2008)

	Refback	**Trackback**	**Pingback**
Mechanizm wyzwalania	Odwiedzający serwis linkujący klika na link i jego przeglądarka przenosi go do serwisu linkowanego	Kod na serwerze linkującym bada dodane lub uaktualnione dokumenty, ekstrahuje linki i wysyła powiadomienie do linkowanego serwera o każdym znalezionym linku	Kod na serwerze linkującym bada dodane lub uaktualnione dokumenty, ekstrahuje linki i wysyła powiadomienie do linkowanego serwera o każdym znalezionym linku
Medium powiadamiania	Wartość *HTTP referer*	*HTTP POST*	Wywołanie *XML-RPC*
Mechanizm przechwytywania	Badanie przychodzących wartości *HTTP referer*	Skrypt przechwytywania trackback	Funkcja *XML-RPC*
Informacja wysyłana przez serwer linkujący	Żadna	• Nazwa linkującego serwisu • Tytuł linkującego posta • Fragment linkującego posta • URL linkującego posta	• URL linkowanego posta • URL linkującego posta
Dodatkowa informacja dla serwera linkowanego	*HTTP referer* wysłany przez przeglądarkę odwiedzającego w momencie kliknięcia na link	Adres IP serwera linkującego	Adres IP serwera linkującego
Mechanizm auto-wykrywania	Żaden	Specjalnie sformatowana informacja w ciele linkowanej strony	Specjalny nagłówek HTTP lub znacznik LINK na linkowanej stronie
Wymagana akcja, gdy otrzymywane jest powiadomienie	• Ekstrakcja wartości *referer* z przychodzących nagłówków *HTTP* • Pobieranie odwołującej się strony • Parsowanie pobranej strony w poszukiwaniu żądanej informacji	• Pobieranie informacji z: • podanych parametrów • lub pobieranie i parsowanie podanego URL'a	• Pobieranie strony spod „URL'a linkującego posta” • Parsowanie pobranej strony w poszukiwaniu żądanej informacji
Zalety	Nie wymaga specjalnego kodu na serwerach linkujących (sam link staje się powiadomieniem, gdy ktoś na niego kliknie)	Cała informacja potrzebna linkowanemu serwerowi (nazwa linkującego serwisu, tytuł i fragment posta) jest zawarta w samym powiadomieniu	• Mechanizm powiadamiania ma kompletną specyfikację techniczną • Mniej podatny na spam
Wady	• Brak powiadomienia, dopóki ktoś nie kliknie na link • Poleganie na przeglądarce odwiedzającego, że przekazuje właściwą informację w *HTTP referer* • Linkowany serwis musi pobrać i sparsować stronę linkującego serwisu, aby ekstrahować potrzebną informację	• Powiadamianie wymaga pozytywnej akcji serwera linkującego • Mechanizm powiadamiania ma tylko częściową specyfikację techniczną • Auto-wykrywanie informacji może uniemożliwić walidację XHTML	• Powiadamianie wymaga pozytywnej akcji serwera linkującego • Linkowany serwis musi pobrać i sparsować stronę linkującego serwisu, aby ekstrahować potrzebną informację

Tabela 2.2. Porównanie cech technik linkback: refback, trackback i pingback źródło: (Contributors 2008)

Powstały nawet osobne wyszukiwarki przeznaczone tylko dla blogów pozwalające określać nowe parametry wyszukiwania, jak data publikacji i autor, co w przypadku tradycyjnych stron WWW nie było możliwe. Z kolei użytkownicy portali blogowych mogli znajdywać nowe treści nie korzystając z wyszukiwania, ale grupowania i przeglądania np. według tagów, popularności bloga czy przechodząc do kolejnych losowo wybranych blogów w ramach jednego portalu.

Scentralizowane zarządzanie blogami na poziomie serwisu umożliwiło również kontrolowanie wzajemnego linkowania blogów i postów. Poza tworzeniem lokalnych rankingów możliwe stało się automatyczne tworzenie linków zwrotnych, które informowały autora i czytelników konkretnego posta o innych postach, które się na tego posta powołują (linkując do niego). Była to nowa funkcjonalność, ponieważ w stosunku do jedynej alternatywy, w postaci słowa kluczowego `link:` w wyszukiwarce Google, dawała informację pełną (w obrębie serwisu) i zawsze aktualną. Jednak, aby rozszerzyć tą funkcjonalność na całą blogosferę potrzebny był standard wzajemnego powiadamiania się serwerów o linkowaniu postów między serwisami. W odpowiedzi na taką potrzebę w 2002 roku pojawiły się dwie konkurencyjne specyfikacje: *Trackback* (Six Apart 2002) i *Pingback* (Langridge i Hickson 2002). Pomimo że żadna z nich nie została zatwierdzona jako standard przez jakąkolwiek organizację standaryzującą, obie techniki są wykorzystywane przez duże serwisy blogowe. Podstawową różnicą między *Trackback* i *Pingback* jest interakcja użytkownika. *Pingback* jest przezroczyste dla usera (pingowane są URL'e z treści posta - w WordPress musimy mieć zahaczone "*Attempt to notify any Weblogs linked to from the article (slows down posting.)*", a autor docelowego posta musi mieć "*Allow pings*"), a *trackback*'a trzeba zrobić świadomie i podać *excerpt*.

Chcąc szczegółowo porównać obie techniki można sięgnąć do zestawienia stworzonego przez Wikipedystów. Jak się okazuje w 2006 roku dołączyła do nich jeszcze jedna technika nazwana *Refback*. Technika ta ma nie wymagać rekonfiguracji serwerów obsługujących powiadamianie o linkowaniu, jednak nie stwierdzono, aby istniał działający serwis wykorzystujący *Refback*. Pomimo to, ze względu na podobną funkcjonalność, *Refback* wraz z *Trackback* i *Pingback* znalazł się w Wikipediowym zestawieniu metod powiadamiania autorów o dokumentach WWW linkujących do ich dokumentów nazwanych ogólnie *Linkback* (tabela 2.2.).

W zestawieniu widać, że jedyną zaletą *Refback* jest brak potrzeby konfiguracji serwerów linkujących. Stąd popularność pozostałych dwóch technik. Jeśli chodzi o wymagania, to *Pingback* wymaga serwera WWW obsługującego XML-RPC, za to uwalnia użytkownika od podawania informacji do powiadomienia i działa automatycznie. Z kolei *Trackback* używa tradycyjnej metody HTTP POST, ale wymaga ingerencji użytkownika w sam proces powiadamiania. Dodatkowo *Trackback* okazał się bardziej podatny na spam w postaci komentarzy, przez co część serwisów WWW zrezygnowała z tej techniki.

2.2.5 bLink

Realizacją koncepcji hipertekstu Nelsona w wersji analogowej, a właściwie łączącej świat analogowy i cyfrowy, może być bLink. BLink to skrót od linka książkowego (ang. *book link*) pozwalający tworzyć interaktywne referencje w tradycyjnych książkach drukowanych. Ta technika analogowego hiperłącza została zaprezentowana w 2007 roku na konferencji *O'Reilly Media Tools of Change* (Kelaidis 2007). BLink to maska nadrukowana na zwykły druk tuszem przewodzącym. Tusze przewodzące zawierają sproszkowane srebro lub węgiel, które nadają im właściwości przewodnika. Zastosowania takich tuszy znane są od lat. Głównie stosuje się je jako tani sposób drukowania obwodów na papierze. Link książkowy jest właśnie takim nadrukowanym obwodem, z tym, że jest to obwód otwarty. Ponieważ ludzkie ciało też jest przewodnikiem, dotknięcia takiego nadruku palcem powoduje zamknięcie obwodu i wyzwolenie akcji. Akcja może być wykonana bezpośrednio na książce, bądź przeniesiona na dowolne urządzenie, jak PDA, telefon czy odtwarzacz MP3. Autor wolałby jednak, aby książka pozostała autonomiczna. To znaczy, jeżeli link ma odtwarzać muzykę, to nie powinien być do tego potrzebny żaden inny sprzęt. Co najwyżej

słuchawki, które można wpiąć w grzbiet książki. Generalnie zastosowania bLinka nie powinny niszczyć pierwotnego i naturalnego sposobu używania książki, jak dzieje się to w przypadku książek elektronicznych (ang. *e-book*). BLink powinien tylko rozszerzyć możliwości użycia tradycyjnej książki, a nie zmienić ją w urządzenie elektroniczne.

2.3 Kolekcja

Ogromna liczba dokumentów występujących w WWW oraz różny poziom ich dostępności nie pozwala na objęcie wszystkich tych dokumentów w jednej analizie. Zakładając, że analiza nie jest przeprowadzana na zupełnie ogólnym poziomie istnieje potrzeba ograniczenia liczby dokumentów poddawanych analizie. W niniejszej pracy, analizy na najniższym poziomie szczegółowości, czyli o największej ziarnistości, dotyczą kolekcji dokumentów.

W większości prac naukowych z zakresu systemów wyszukiwania pojęcie "kolekcji dokumentów" przyjmowane jest jako pojęcie pierwotne. Intuicyjnie jest ono rozumiane jako „zbiór dokumentów”, najczęściej dokumentów hipertekstowych, będących pewnym podzbiorem WWW, np. serwisem. Oznacza to, że najczęściej dokumenty WWW w jednej kolekcji powiązane są odsyłaczami hipertekstowymi, choć nie musi to być regułą. Intuicyjność pojęcia „kolekcji dokumentów” może wynikać ze słownikowego znaczenia „kolekcji”.

Definicja 2.3.1. Kolekcja

„[...] publikacja zawierająca różne prace (ang. *variety of works*) [...]” - (George A. Miller 2006)

Z definicji 2.3.1 wynika, że, w kontekście WWW, kolekcję można kojarzyć z dokumentami publikowanymi w sieci. Polskie źródła nie podają definicji kolekcji wprost, a jedynie związek między pojęciem „kolekcji” i „zbioru”. Polski WordNet (Piasecki, Szpakowicz, i Broda 2009) podaje, że „kolekcja” jest hiponimem „zbioru”. Natomiast Słownik Języka Polskiego (PWN 2008) uznaje „kolekcję” za synonim „zbioru”.

W samej informatyce pojęcie „kolekcji” najczęściej używane jest jako zbiorcze określenie struktur danych, takich jak: zbiory, listy, wielozbiory, drzewa czy grafy. O ile same te pojęcia są pojęciami abstrakcyjnymi zaczerpniętymi z matematyki, o tyle używanie ich w kontekście konkretnych implementacji nadaje im konkretnych i szczegółowych znaczeń. Przykładem może być „Java Collections Framework” (Sun Microsystems, Inc. 2003), w którym każdy rodzaj kolekcji jest konkretnym interfejsem (klasą obiektową), posiadającym konkretne metody.

Podobnie ma się sprawa w bibliotekach. Tu pojęcie „kolekcji dokumentów” definiowane jest szczegółowo na poziomie metadanych, najczęściej w odniesieniu do konkretnego systemu, przy użyciu pojęć również specyficznych dla danego systemu. Na przykład biblioteka Uniwersytetu Michigan w polityce publikowania w Internecie przyjmuje definicję 2.3.2.

Definicja 2.3.2. Kolekcja

"Kolekcja dokumentów to zbiór dokumentów, które są logicznie powiązane, zwykle poprzez ich zawartość, docelowych odbiorców czy pochodzenie (np. kolekcja prac stworzonych przez program, projekt lub organizację)." - (U-M Gateway 2008)

Można zauważyć, że w definicji 2.3.2 nacisk położony jest na podobieństwo zawartości, tematyki dokumentów w kolekcji. Z kolei dokumentacja Wielkopolskiej Biblioteki Cyfrowej, opartej na platformie dLibra (Mazurek, Parkoła, i Werla 2006), wprowadza definicję 2.3.3.

Definicja 2.3.3. Kolekcja

"Kolekcja to zbiór publikacji o określonym charakterze, które można opisać pewnym wspólnym zbiorem atrybutów, według których można te publikacje przeszukiwać.

Niektóre atrybuty występują we [sic] wielu kolekcjach, inne natomiast są specyficzne dla określonych kolekcji. Np. atrybut *tytuł* występuje we wszystkich kolekcjach, gdyż każdą publikację można jakoś zatytułować. Atrybut *drukarz* natomiast jest elementem opisu publikacji tylko w kolekcji Dziedzictwo kulturowe. [...]" - (PSNC 2006)

W definicji 2.3.3 wyczuwa się, że pojęcie kolekcji bardziej związane jest z metadanymi - atrybutami opisującymi cechy dokumentu, niż z samą treścią, jak ma to miejsce w definicji 2.3.2. Oczywiście metadane powinny odzwierciedlać zawartość dokumentów, i kolekcje wyznaczone według obu definicji najczęściej będą się pokrywać. Jednak konfrontacja tych definicji wskazuje, że pojęcie „kolekcji" używane jest różnie, w różnych kontekstach i podejściach do metod przetwarzania dokumentów. W odniesieniu do samego WWW najbardziej autorytatywną definicją „kolekcji" powinny być dokumenty W3C.

Definicja 2.3.4. Kolekcja WWW

„Porcja lub sekcja serwisu WWW, składająca się z dwóch lub więcej stron WWW, która reprezentuje nietrywialny, niezależny (ang. *self-contained*) zasób, jednakże utrzymywany przez jednego wydawcę całej witryny." - (Lavoie i Nielsen 1999)

Według definicji 2.3.4 kolekcją WWW jest np. podserwis (definicja 2.1.16), ale nadserwis (definicja 2.1.15) już nie, ponieważ wymogiem tej definicji jest jeden wydawca, co w przypadku nadserwisu najczęściej nie jest spełnione. Jednak w nadserwisie wszystkie dokumenty odnoszą się do jednej i tej samej strony głównej, co jest ich cechą wspólną, która, w myśl definicji 2.3.2 i 2.3.3, łączy je w kolekcję. W rekomendacjach dotyczących standardów dokumentów XML pojęcie „kolekcji" rozumiane jest jako zbiór węzłów. Na to, że kolekcja jest zbiorem wskazuje wymóg jej uporządkowania w definicji 2.3.7.

Definicja 2.3.5. Węzeł

„Węzeł to instancja jednego z rodzajów węzłów zdefiniowanych Modelu Danych XQuery/XPath (*XDM*)." - (Chamberlin et al. 2007) (Kay et al. 2007)

Definicja 2.3.6. Węzły

„Istnieje siedem rodzajów węzłów w tym modelu danych: dokument, element, atrybut, tekst, przestrzeń nazw, instrukcja przetwarzania i komentarz." - (Malhotra et al. 2007)

Definicja 2.3.7. Sekwencja (ang. *sequence*)

„Sekwencja to uporządkowana kolekcja zero lub więcej elementów." - (Chamberlin et al. 2007) (Kay et al. 2007) (Malhotra et al. 2007)

Definicja 2.3.8. Kolekcja domyślna

„Kolekcja domyślna to sekwencja węzłów, które są wynikiem wywołania funkcji *fn:collection* bez argumentów." - (Chamberlin et al. 2007) (Kay et al. 2007)

Definicja 2.3.8, może być równoważna definicji 2.3.4 przy założeniu, że wszystkie węzły są rodzaju dokument. Jeśli węzły są innego rodzaju, to cała kolekcja może się mieścić w jednych dokumencie WWW, czyli drzewie, którego korzeniem jest węzeł rodzaju dokument. Jest to zgodne z rekurencyjnością kolekcji wynikającej z definicji 2.1.15 i 2.1.16. Ciekawym faktem, który uwypukla elastyczność pojęcia kolekcja, jest to, że o ile kolekcja jest zbiorem, więc jej węzły nie są uporządkowane, o tyle - w myśl definicji 2.3.8 - kolekcja domyślna to sekwencja, więc zawiera węzły uporządkowane. Aby ostatecznie podkreślić różnorodność interpretacji pojęcia kolekcji

można przytoczyć definicję 2.3.9, według której w ogólności „kolekcja” jest „bazą danych”.

Definicja 2.3.9. Baza danych

„Ten termin [„baza danych” - przyp. tł.] został użyty ogólnikowo w znaczeniu kolekcji węzłów. [...]” - (Girschweiler 1995)

W podobnym duchu termin „kolekcja” został użyty w definicji 2.2.11. W tej definicji też chodziło o bazę danych, ale nie dokumentów, tylko linków.

2.4 Modelowanie kolekcji dokumentów WWW

Chcąc modelować kolekcję dokumentów WWW, czyli elementy wzajemnie powiązane jednokierunkowymi odsyłaczami, najoczywistszym wydaje się być użycie struktury grafu skierowanego (w zasadzie „multigrafu zorientowanego” (Kulikowski 1986, 40)). W grafie tym, węzłami są dokumenty, a krawędziami odsyłacze hipertekstowe. Graf jest najpopularniejszym modelem kolekcji czy nawet całej WWW (Brin i Page 1998). Model grafu jest wykorzystywany w najbardziej znanych algorytmach wyszukiwania w WWW, jak HITS (Kleinberg 1998) czy – używany przez wyszukiwarkę Google – PageRank (Page et al. 1998). Banał modelowania WWW grafem jest tak powszechny, że nawet Tim Berners Lee – autor idei WWW – stwierdził, że współczesna wersja WWW mogłaby zawierać słowo „graf” w nazwie (Tim Berners-Lee 2007b).

Innym aspektem potwierdzającym fakt, że myśląc o sieci WWW jej twórcy wyobrażają sobie graf jest adaptowanie przez nich słownictwa z teorii grafów, jak ma to miejsce na przykład w definicjach 2.2.13 czy 2.3.5.

2.4.1 Elementy teorii grafów

Aby móc mówić o adaptowaniu pojęć i modelowaniu za pomocą grafu, najpierw należy przyjrzeć się definicjom z zakresu teorii grafów przyjmowanych w literaturze. Przytoczone definicje pochodzą z różnych źródeł, jednak dotyczą spójnej koncepcji. Ewentualne różnice w nazewnictwie i interpretacji omówiono poniżej.

Definicja 2.4.1. Graf (nieskierowany)

„*Graf liniowy* (lub po prostu *graf*) $G=(V, E)$ składa się ze zbioru obiektów $V = \{v_1, v_2, ...\}$ zwanych wierzchołkami oraz zbioru $E = \{e_1, e_2, ... \}$, którego elementy nazywa się krawędziami. Krawędź e_k utożsamia się z nieuporządkowaną parą wierzchołków (v_i, v_j). Wierzchołki v_i, v_j związane z krawędzią e_k nazywa się *wierzchołkami końcowymi* krawędzi e_k. Najpopularniejszym sposobem przedstawienia grafu jest rysunek, na którym wierzchołki są reprezentowane przez punkty, a każda krawędź – przez odcinek łączący jej wierzchołki końcowe. Często sam rysunek określa się jako graf, [...]” - (Deo 1980, 15)

Definicja 2.4.2. Graf skierowany

„*Graf skierowany* składa się ze zbioru wierzchołków $V = \{v_1, v_2, ...\}$, zbioru krawędzi $E = \{e_1, e_2, ... \}$ i odwzorowania Ψ, które przekształca każdą krawędź na pewną uporządkowaną parę wierzchołków (v_i, v_j). Tak jak w przypadku grafów nieskierowanych wierzchołki przedstawia się za pomocą punktu, a krawędź – fragment linii między v_i a v_j ze strzałką skierowaną od v_i do v_j.[...]. O grafie skierowanym mówi się także jako o *grafie zorientowanym*.” - (Deo 1980, 254)

Formalne definicje 2.4.1 i 2.4.2 definiują graf skierowany i nieskierowany jako dwie, różne klasy grafów. Graf skierowany nie jest nazwany szczególnym przypadkiem grafu nieskierowanego, choć

można odnieść wrażenie, że tak właśnie jest. Deo wprowadza jednak pojęcie grafu skierowanego dopiero w połowie swojej książki wskazując: „większość pojęć i terminologia grafów nieskierowanych mają także zastosowanie do grafów skierowanych" (Deo 1980, 254). Wprowadzając nowe pojęcia dotyczące grafów skierowanych, autor najczęściej robi to przez uściślenie analogicznych pojęć dotyczących grafów nieskierowanych. Fakt, że druga połowa książki dotyczy zagadnień związanych z grafami skierowanymi, podczas gdy zagadnienia z pierwszej połowy odnoszą się do obu klas grafów, wskazuje, że grafy skierowane są trudniejsze w analizie, ale i mają szersze pole zastosowań niż grafy nieskierowane. Uwidocznia się to np. w odniesieniu autora do rodzajów grafów: „Podobnie jak grafy nieskierowane, grafy skierowane są różnego rodzaju. W rzeczywistości, z uwagi na wybór przypisania skierowania każdej krawędzi, grafy skierowane mają więcej rodzajów niż grafy nieskierowane." (Deo 1980, 258). Autor zwraca jeszcze uwagę, że w niektórych źródłach pojęcia „graf zorientowany" i „graf skierowany" są używane zamiennie - za czym sam obstaje, a w innych pojęcia te są rozróżniane.

Nieco inne podejście do różnicy między grafami, w których krawędzie mają, bądź nie mają zwrotów można zaleźć w Kulikowski 1986). Tutaj najpierw wprowadzone jest pojęcie „grafu zorientowanego" (autor w ogóle nie używa terminów „skierowany" i „nieskierowany"), a następnie, jako jego uogólnienie wprowadzone jest pojęcie „grafu niezorientowanego" (Kulikowski 1986, 41). Siedem rozdziałów dalej Kulikowski przyznaje: „Z określenia grafów zorientowanych (orgrafów), które zamieściliśmy w rozdz. 2, wynika, że można je uznać za szczególny rodzaj grafów niezorientowanych, których krawędziom nadano dodatkowo zwrot przekształcając je w łuki." Jak widać autor rozróżnia tu terminy dla analogicznych pojęć w obu rodzajach grafów: „Odpowiednikami takich pojęć odnoszących się do grafów niezorientowanych, jak: krawędź, łańcuch i cykl, są – jak pamiętamy – pojęcia: łuk, droga i obwód w orgrafie, w którym przypisuje się im określone zwroty." (Kulikowski 1986, 302)

Kulikowski wprowadza również od razu definicję multigrafu (Deo wspomina tylko, że taki termin jest używany przez niektórych autorów w odniesieniu do grafów „z pętlami własnymi i/lub krawędziami równoległymi" (Deo 1980, 16)).

Definicja 2.4.3. Multigraf zorientowany

> „Graf zorientowany nazywamy *multigrafem zorientowanym*, jeśli na odwzorowanie φ nie jest nałożony warunek jednoznaczności, to znaczy jeśli każdej nieuporządkowanej parze węzłów $[a_i, a_j]$ może odpowiadać więcej niż jeden łuk l_{ij}." - (Kulikowski 1986, 40)

Z definicji 2.4.3 wynika, że „multigraf" jest uogólnieniem „grafu „ze względu na liczbę krawędzi incydentnych z parą lub jednym węzłem. Uogólnieniem na innym poziomie, a konkretnie: na poziomie możliwej liczby węzłów incydentnych z jedną krawędzią, jest hipergraf.

Definicja 2.4.4. Hipergraf

> „Hipergraf jest parą (V, E) rozłącznych zbiorów, w których elementy E są niepustymi podzbiorami (dowolnej liczności) V. Więc, grafy są szczególnymi przypadkami hipergrafów." - (Diestel 2006, 28)

Według definicji 2.4.4 w hipergrafie krawędzie (czy elementy zbioru E) mogą łączyć nie co najwyżej dwa – jak w przypadku grafów – lecz dowolną liczbę wierzchołków. Definicja ta jest podana używając nieco innych formalizmów niż poprzednie trzy definicje, ponieważ wprowadzona została przez jeszcze innego autora – Diestela. Co ciekawe, o ile Deo i Kulikowski używali różnych terminów: „skierowany" i „zorientowany" dla tego samego pojęcia, o tyle Diestel jest tym autorem, który rozróżnia pojęcia „graf skierowany" i „graf zorientowany". Poza formalnymi definicjami autor ujmuje tę różnicę następująco: „grafy zorientowane to grafy skierowane bez pętli i krawędzi

wielokrotnych (równoległych – *przyp. tłum.*)" (Diestel 2006, 28). To rozróżnienie jest tożsame z rozróżnieniem przez Kulikowskiego „multigrafu" i „grafu" (definicja 2.4.3).

Często zdarza się, że modelując pewną rzeczywistość potrzebne jest zróżnicowanie wartości węzłów lub krawędzi. Można sobie wyobrazić np. model sieci energetycznej, wodociągowej czy gazowej, w której węzły występują w różnych odległościach od siebie. Chcąc modelować te odległości trzeba nadać krawędziom odpowiadające tym odległościom wagi.

Definicja 2.4.5. Graf ważony

> „Graf G, którego węzłom lub krawędziom przyporządkowano jednoznacznie współczynniki wagowe, to jest określono jedną z funkcji: u: $A \rightarrow R$, v: $C \rightarrow R$, gdzie A i C są odpowiednio zbiorem węzłów i krawędzi grafu, a R jest przestrzenią liczb rzeczywistych, nazywamy *grafem ważonym*." - (Kulikowski 1986, 142)

Definicja 2.4.5 daje pewną dowolność interpretacji. Mianowicie, autor nie uściśla czy określona ma być „przynajmniej" jedna z funkcji u lub *v*, czy „co najwyżej" jedna. Czyli nie wiadomo czy autor dopuszcza w definicji przypadek, gdy i węzły, i krawędzie są obciążone wagami. Nie jest to jednak bardzo istotne, ponieważ w praktyce stosuje się najczęściej obciążanie samych węzłów albo – częściej – samych krawędzi. Definicję grafu ważonego, który jest zwykłym grafem z wagami (jedynie) na krawędziach można znaleźć w Hartmann i Weigt 2006). Obciążanie krawędzi jest zgodne z praktycznym modelowaniem różnego rodzaju sieci. Dlatego często „sieć" jest synonimem „grafu ważonego". W Deo 1980) brak formalnej definicji, a samo pojęcie „grafu ważonego" wprowadzone jest przy okazji problemu komiwojażera. Natomiast w Diestel 2006) termin graf ważony w ogóle nie jest używane. Jest natomiast formalna definicja sieci, która – w kontekście przepływu w sieci – jest uszczegółowieniem grafu skierowanego poprzez dodanie funkcji pojemności.

Problem używania – przez różnych autorów – różnych terminów dla tych samych koncepcji jest nagminnym problemem, zwłaszcza w teorii grafów. Na przykład Kulikowski dla pojęcia „grafu zorientowanego" używa skrótowo terminu „orgraf", natomiast Diestel – „digraf". Rzadko spotyka się też, w dwóch różnych źródłach, spójne definicje takich pojęć jak: „krawędź", „droga", „łańcuch", „ścieżka" czy „cykl". Dlatego każda praca naukowa wykorzystująca pojęcia z teorii grafów powinna na wstępie je zdefiniować. Jest to niestety uciążliwe, a - ze względu na ograniczone rozmiary publikacji - najczęściej pomijane. Próbę ustandaryzowania terminologii z zakresu teorii grafów podjęli anglojęzyczni wikipedyści (Contributors 2008). Niestety dodatkowym problemem jest tutaj kwestia tłumaczeń terminów. Na przykład termin „*path*" jest tłumaczony na polski jako „ścieżka" albo „droga". Co gorsza niektórzy polscy autorzy wprowadzają „ścieżkę" i „drogę" jako niezależne pojęcia. Dopóki, więc, nie zostanie podjęta próba ujednolicenia terminologii we wszystkich językach, w których istnieją publikacje z teorii grafów, dopóty nie należy przyjmować wspomnianych pojęć jako pierwotne. Kwestia ta dotyczy również każdej innej teorii, w której badania prowadzone są w sposób zdecentralizowany, czyli w zasadzie każdej popularnej teorii.

2.4.2 Indeks Cytowań Naukowych (SCI) i współczynnik istotności (IF)

W połowie lat pięćdziesiątych zeszłego wieku Eugene Garfield dostrzegł potencjał informacyjny zawarty w cytowaniach publikowanych artykułów. Początkowo stworzył Indeks Cytowań Genetyki (ang. *Genetics Citation Index*), a następnie rozszerzył go do bardziej ogólnego Indeksu Cytowań Naukowych (ang. *Science Citation Index (SCI)), a* cała idea została opublikowana w Garfield 1964). Jak wspomina autor w Garfield 2006):

> „[...] Irving H. Sher i ja stworzyliśmy współczynnik wpływu (ang. *impact factor*) dla

czasopism, aby pomógł w wyborze dodatkowych czasopism źródłowych. Aby to zrobić, po prostu przesortowaliśmy indeks cytowań autorów w indeks cytowań czasopism. Z tego zadania dowiedzieliśmy się, że początkowo nowy SCI powinien pokryć ścisłą grupę szeroko i często cytowanych czasopism. Zauważmy, że w 2004 roku *Journal of Biological Chemistry* opublikował 6500 artykułów, podczas gdy artykuły z *Proceedings of the National Academy of Sciences* były w tamtym roku cytowane ponad 300 000 razy. Mniejsze czasopisma mogłyby nie zostać wybrane, jeśli polegalibyśmy wyłącznie na liczbie publikacji, dlatego stworzyliśmy współczynnik wpływu dla czasopism (ang. *journal impact factor (JIF)*)."

Można zauważyć, że współczynnik wpływu mierzony jest dzięki skierowanemu związkowi cytowania. Traktując artykuły jako węzły, a związki cytowania jako krawędzie można przedstawić bazę obliczania współczynnika jako graf skierowany według definicji 2.4.2. Na taką interpretację powołuje się Kleinberg, tworząc analogię takiego grafu do grafu modelującego dokumenty i związki między nimi w WWW, wykorzystywanego w jego algorytmie HITS. Algorytm Kleinberga był z kolei inspiracją dla powstania algorytmu PageRank wykorzystywanego w najpopularniejszej obecnie wyszukiwarce Google.

2.4.3 Algorytm HITS

W (Kleinberg 1998) został przedstawiony algorytm HITS. HITS to algorytm wyszukiwania tematycznego na podstawie analizy hiperłączy (ang. *Hyperlink-Induced Topic Search*). Algorytm skupia się na analizie kolekcji stron WWW relewantnych dla wyszukiwań szeroko-tematycznych (ogólnych) w celu znalezienia źródeł (stron WWW) autorytatywnych dla tych tematów. Autor rozróżnia trzy typy zapytań:

- zapytania szczegółowe, np.: „Czy Netscape obsługuje API podpisywania kodu w JDK 1.1?"
- zapytania szeroko-tematyczne, np.: „Znajdź informacje o języku programowania Java."
- zapytania o strony podobne, np.: „Znajdź strony podobne do java.sun.com."

Stosując jedynie wyszukiwanie tekstowe, z każdym typem zapytania wiążą się problemy. Przy zapytaniach szczegółowych pojawia się problem małej liczności (ang. *scarcity*). Polega on na tym, że w odpowiedzi znajduje się niewiele stron zawierających potrzebną informację i często trudno jest te strony odnaleźć w dużym zbiorze stron z odpowiedzi. Z kolei przy zapytaniach ogólnych występuje problem obfitości (ang. *abundance*). Liczba stron w odpowiedzi pytanie szeroko-tematyczne jest o wiele za wysoka, żeby człowiek mógł je ogarnąć. Aby wyszukiwanie było efektywne, należy z ogromnej kolekcji stron relewantnych wyfiltrować te, które są autorytatywne lub definiujące.

Samo znalezienie stron autorytatywnych utrudniają fakty:

1. Na przykład: *www.harvard.edu* nie będzie wysoko w rankingu wyników na zapytanie „Harvard", ponieważ strona ta nie używa słowa kluczowego „harvard" w treści najczęściej, ani w inny sposób, w który funkcja wyszukiwania pełnotekstowego mogła tą stronę uznać za najważniejszą;
2. Nie można spodziewać się, że naturalne źródła autorytatywne będą używały na swoich stronach odpowiednich słów kluczowych, np. Honda, czy Toyota nie zamieszczają na stronach domowych terminu „producent pojazdów".

Jako rozwiązanie tych problemów autor proponuje analizę struktury hiperłączy. Przede wszystkim, sam fakt linkowania oznacza, że autor strony linkującej sądzi, że strona do której linkuje jest autorytatywna w kontekście tego hiperłącza. Kleinberg twierdzi, że tego typu osąd jest dokładnie tym, czego potrzeba do sformułowania pojęcia autorytetu, jako źródła autorytatywnego. Poza tym, dzięki linkom, autorytety, które same siebie nie opisują – są opisane przez linkujące do nich strony.

Należy zauważyć, że algorytm HITS został wymyślony pod koniec lat 90-tych, w momencie, gdy

języki skryptowe typu: PHP, JSP nie były jeszcze powszechnie używane, więc strony były głównie statyczne. Oznaczało to, że hiperłącza były tworzone ręcznie przez autorów stron WWW lub webmasterów. Obecnie, dzięki dynamicznemu generowaniu stron, większość hiperłączy jest w serwisach WWW powtarzana na każdej podstronie. Najczęściej nie są to linki autorytatywne. Kleinberg wspomina o negatywnym wpływie na algorytm linków nawigacyjnych. Dla nich proponuje heurystykę dzieląc linki na poprzeczne (ang. *transverse*) i wewnętrzne (ang. *intrinsic*). Linki wewnętrzne to takie, które łączą 2 dokumenty w jednej domenie. Domeny rozróżniane są na podstawie adresów URL. Ponieważ linki wewnętrzne, które najczęściej są linkami nawigacyjnymi, wprowadzają patologie do analizy struktury hiperłączy, dlatego autor przed analizą usuwa ze struktury wszystkie linki wewnętrzne.

Jednak nie tylko linki wewnętrzne mogą być niezwiązane z informacją autorytatywności. Kleinberg zwraca uwagę na linki sponsorowane (reklamy) oraz na linki typu „designed by", gdy witryna WWW została wykonana na zamówienie przez firmę trzecią. W dzisiejszym WWW niemal obowiązkowe są w stopce każdej strony linki typu „powered by", „valid" i „license". Te pierwsze wskazują na używane przez serwis oprogramowanie serwerowe (WWW, baza danych, silnik skryptowy, itp.), bądź częściej, na system CMS, używany do zarządzania treściami witryny. Drugie to linki do narzędzi sprawdzających poprawność kodu XHTML, CSS według standardów W3C. Z kolei „license" to linki do standardowych treści licencji typu *Creative Commons*, które świadczą o zgodzie autora na dowolne wykorzystywanie treści lub wyglądu witryny.

Dziś jeszcze innym elementem, wprowadzonym przez dynamiczne generowanie stron, mającym wpływ na zniekształcenie analizy hiperłączy jako struktury tworzonej świadomie przez autorów treści, jest automatyczne linkowanie słów kluczowych. Niektóre serwisy, wyświetlając treści, automatycznie zamieniają określone słowa kluczowe w hiperłącza. Najczęściej są to odwołania do słownika z wyjaśnieniem danego pojęcia. Czasem mogą to być linki sponsorowane.

Aby uporać się z tego typu problemami Kleinberg zaproponował heurystykę polegającą na zmniejszeniu znaczenia tego typu linków poprzecznych. Dla zadanego parametru *m* o wartości np. z zakresu 4-8 należy w analizie wziąć pod uwagę nie więcej niż *m* linków z jednej domeny wskazujących tą samą stronę. Lecz ta heurystyka nie została przez autora sprawdzona empirycznie.

Jednak dziś pozbycie się wspomnianych problemów zarówno z linkami wewnętrznymi, jak i poprzecznymi daje nam postulat oddzielenia formy od treści. Dzięki, realizowanej coraz szerzej, idei syndykowania informacji RSS, pozwalającej nadać jej dowolny wygląd, mamy dostęp do czystej treści - bez linków sponsorowanych, związanych z nawigacją, czy wyglądem serwisu. Niestety nie każdy serwis obsługuje jeszcze jakikolwiek format wymiany danych, dlatego dzisiejsze wyszukiwarki WWW ciągle jeszcze analizują również strony WWW w całości - z nierozdzieloną treścią i formą.

Aspektem wpływającym na wykrywanie autorytetów, odnoszącym się nie tylko do WWW, jest problem znalezienia balansu między kryteriami relewancji i popularności. Autor wskazuje na przykład poważnych problemów wynikających z użycia prostej heurystyki: Ze wszystkich stron zawierających szukane słowo wyświetl te z największą liczbą linków przychodzących. Ale jak wskazano wcześniej wiele zapytań (np. „producent pojazdów") może nie zawierać w odpowiedzi autorytetów. Z drugiej strony, heurystyka będzie uważać popularne strony jak *www.yahoo.com* czy *www.netscape.com* za wysoce autorytatywne ze względu na każde słowo kluczowe, które będą zawierać.

Graf ukierunkowany. Na potrzeby algorytmu Kleinberg przyjmuje jako model WWW graf skierowany G = (V, E). Zbiór V zawiera węzły odpowiadające stronom, a skierowane krawędzie (p, q) ϵ E oznaczają istnienie hiperłącza ze strony p do strony q w sposób analogiczny do definicji 2.4.2. Dodatkowo wprowadza dwa pojęcia dla węzłów: stopień wyjścia i stopień wejścia oraz pojęcie podgrafu.

Definicja 2.4.6. Stopień wyjścia (ang. *out-degree*)

„[...] stopień wyjścia węzła *p* to liczba węzłów, do których *p* posiada hiperłącza, [...] ”

Definicja 2.4.7. Stopień wejścia (ang. *in-degree*)

„[...] stopień wejścia węzła *p* to liczba węzłów, które posiadają hiperłącza do *p*. ”

Definicja 2.4.8. Podgraf

„Z grafu *G* można wyizolować małe regiony, czy podgrafy, w następujący sposób. Jeśli $W \subseteq V$ jest podzbiorem stron, używamy *G*[*W*], aby oznaczyć graf powstały (ang. *induced*) na *W*: jego węzłami są strony w *W*, a jego krawędzie odpowiadają wszystkim hiperłączom pomiędzy stronami w *W*.”

Ze względu na złożoność obliczeniową analizy struktury hiperłączy wyłowienie autorytatywnych stron zbioru odpowiedzi na szeroko-tematyczne zapytanie zachodzi konieczność ograniczenia tego zbioru stron. Aby skupić moc obliczeniową na stronach relewantnych należy znaleźć odpowiedni podgraf grafu WWW. Nie jest to zadanie trywialne, ponieważ gdyby wziąć na przykład pod uwagę tylko podzbiór zawierający wszystkie słowa kluczowe zapytania, może się okazać, że zbiór ten jest nadal zbyt duży. Poza tym jak wcześniej zauważono niektóre najlepsze autorytety mogłyby się w tym zbiorze nie zmieścić. Najlepiej, gdyby analizowany zbiór był kolekcją S_σ, która posiada następujące cechy:

(1) S_σ jest relatywnie mała,
(2) S_σ jest bogata w strony relewantne,
(3) S_σ zawiera większość najsilniejszych autorytetów.

Utrzymując niewielką liczność S_σ mamy wystarczającą moc obliczeniową, żeby zastosować nietrywialny algorytm. Zapewniając, że S_σ jest bogate w autorytety prościej będzie znaleźć dobre autorytety, ponieważ będą one mocno linkowane wewnątrz S_σ.

Aby otrzymać kolekcję, która zaspokoi pierwsze dwie własności S_σ można wziąć pierwsze *t* stron z odpowiedzi wyszukiwarki WWW. Taki zbiór, nazwany R_σ, daleki jest jednak od spełnienia postulatu (3). Dodatkowo R_σ, będący podzbiorem wszystkich możliwych stron zawierających słowa zapytania σ, jest najczęściej słabo ustrutkturyzowany, ze względu na niewielką liczbę hiperłączy pomiędzy stronami z R_σ.

Możemy jednak posłużyć się R_σ, aby zbudować S_σ, które spełni trzy wymienione postulaty. Chociaż R_σ może nie zawierać silnych autorytetów dla zapytania σ, to jest wysoce prawdopodobne, że do takich autorytetów będą posiadały hiperłącza strony z R_σ. Dlatego należy rozszerzyć zbiór R_σ dodając do niego wszystkie strony, które linkują lub są linkowane przez dokumenty z tego zbioru. Aby utrzymać postulat (1) należy jednak wprowadzić ograniczenie, że pojedynczy dokument z R_σ nie może wnieść do tego zbioru więcej niż *d* nowych dokumentów (do których ma hiperłącze, lub które mają hiperłącze do niego).

W ten sposób otrzymujemy kolekcję S_σ, którą nazywamy zbiorem bazowym, a graf tworzony przez dokumenty tego zbioru **grafem ukierunkowanym** (ang. *focused graph*). Empiryczne testy potwierdziły, że dla *t*=200 i *d*=50 używając silnika AltaVista kolekcja S_σ zazwyczaj spełnia trzy postulaty, a jej wielkość waha się w granicach 1000-5000.

Autorytety i huby. Otrzymując graf ukierunkowany najprostszym rozwiązaniem w celu wyłowienia stron autorytatywnych wydaje się posortowanie stron z S_σ według ich stopnia wejścia. Jednak takie rozwiązanie poza rozpoznaniem dobrych autorytetów dla zapytania znajdzie również strony, które są generalnie popularne, niezależnie od zapytania. Aby pozbyć się ogólnie popularnych stron należy sprawdzić nie tylko czy mają wysoki stopień wejścia, ale również czy

zbiory stron linkujących do nich znacząco się pokrywają. Poza stronami, które są autorytetami należy również wyróżnić huby, czyli strony, które mają hiperłącza do wielu autorytetów. Strony hub łączą więc autorytety o wspólnej tematyce, a odrzucają nierelewantne strony o dużym stopniu wejścia.

Autorytety i huby tworzą wzajemnie wzmacniany związek (ang. *mutually reinforcing relationship*): dobry hub to strona, która wskazuje na wiele dobrych autorytetów; dobry autorytet to strona wskazywana przez wiele dobrych hubów. Aby zidentyfikować autorytety i huby w podgrafie dokumentów z odpowiedzi należy rozbić tę cykliczną zależność za pomocą algorytmu iteracyjnego.

Algorytm iteracyjny. Algorytm zakłada, że każda strona p ma przypisane dwie niezerowe wartości: wagę autorytetu $x^{(p)}$ i wagę huba $y^{(p)}$. Ograniczenie nałożone na te wagi stanowi, że wagi każdego typu są znormalizowane w ten sposób, że suma kwadratów wag autorytetu wszystkich stron z S_σ równa się 1 i suma kwadratów wag huba wszystkich stron z S_σ równa się 1. Im większą wagę autorytetu/huba ma strona, tym jest lepszym autorytetem/hubem.

Aby wyrazić liczbowo wzajemnie wzmacniany związek autor wprowadza 2 operacje:

- I, która ustawia wagę autorytetu jako sumę wszystkich wag hubów na niego wskazujących,
- O, która ustawia wagę huba jako sumę wszystkich wag autorytetów wskazywanych przez niego.

Jak zostało pokazane algorytm aktualizacji wag można uruchomić dla grafu w różnej kolejności. Jednak zawsze można dość do momentu, w którym operacje I i O nie zmieniają wcześniejszych wartości wag, na potrzeby iteracji algorytmu pogrupowanych w wektory. Korzystając z własności algebry liniowej oraz testów empirycznych wykazano, że po najwyżej 20 iteracjach wektory stają się stabilne, czyli wagi osiągają granicę, do których są zbieżne.

Podsumowując należy zauważyć, że kluczowym elementem algorytmu jest modelowanie WWW jako grafu i kolekcji dokumentów WWW, jako jego podgrafu. Algorytm wykorzystuje naturalną równowagę pomiędzy autorytetami i hubami wynikającą ze struktury hiperłączy. Pomimo, że główna cześć algorytmu posługuje się nowatorskim podejściem do wyszukiwania w WWW poprzez analizę struktury hiperłączy, to jednak pośrednio korzysta z metody analizy treści dokumentów. Konstruowanie grafu ukierunkowanego, będącego daną wejściową dla przedstawionego algorytmu HITS, oparte jest na wynikach wyszukiwarki WWW analizującej tekst.

Główną zaletą algorytmu jest poprawa wyników wyszukiwania zwróconych przez wyszukiwarkę pełnotekstową, poprzez wyłowienie i posortowanie stron autorytatywnych dla zapytania. Jednak można zauważyć, że chociaż HITS nie jest typowym algorytmem grupującym, to jednak jego rozwinięcie może zostać wykorzystane do rozróżniania różnych znaczeń terminów użytych w zapytaniach. Przykładem tej koncepcji może być (Pańczyk 2008). Praca ta opiera się o analizę wyników współczesnych wyszukiwarek korzystających z koncepcji analizy struktury hiperłączy. Przedstawiony w pracy algorytm mierzy podobieństwo znaczeniowe między dwoma terminami w oparciu o liczbę wyników zwróconych przez wyszukiwarkę WWW na zapytanie złożone z badanych terminów. Podając jako wejście więcej niż 2 terminy algorytm automatycznie mierzy podobieństwa między każdą parą terminów i zwraca termin najbardziej niepodobny do innych jako najpewniej niepasujący do tematyki, z której pochodzą pozostałe terminy.

2.4.4 Algorytm PageRank

Innym algorytmem znajdowania stron autorytatywnych poprzez analizę struktury hiperłączy jest powstały w tym samym okresie PageRank stworzony na potrzeby wyszukiwarki Google przedstawionej w, a opisany w Page et al. 1998). Ze względu na sukces wyszukiwarki Google, PageRank jest obecnie głównym odniesieniem dla innych algorytmów wyszukiwania i oceniania dokumentów WWW.

Ponieważ PageRank od początku był algorytmem praktycznym, dlatego w porównaniu z HITS

większy nacisk został położony na wydajność i możliwość rzeczywistego zastosowania go w wyszukiwarce. W związku z tym, pomimo podobnej idei przyświecającej HITS i PageRank, oba algorytmu różnią się w kilku kwestiach. Po pierwsze: HITS działa na relatywnie niewielkiej liczbie dokumentów uzyskanych z wyszukiwarek pełnotekstowych. Z kolei PageRank operuje na całej bazie dokumentów WWW przygotowanej specjalnie dla niego. Po drugie: HITS jest uruchamiany w momencie zadawania zapytania, podczas gdy PageRank, aby pokryć obliczeniami wszystkie dokumenty z kopii WWW bez opóźniania zwrócenia wyników dla użytkownika, musi być uruchamiany niezależnie i zakończyć działanie przed pierwszym zapytaniem. Te różnice wynikają z założenia, że HITS ma być algorytmem poprawiającym wyniki innych wyszukiwarek, natomiast PageRank ma być częścią autonomicznej wyszukiwarki Google, z własnym indeksem stron WWW. Z tego powodu aktualność obliczonych wartości PageRank zależy od procesu aktualizacji indeksu Google. Aby nowe strony WWW zostały włączone do indeksu Google i obliczony został dla nich PageRank, wymagana jest cała iteracja aktualizowania bazy dokumentów i przeliczania rankingu, co w początkowej fazie istnienia Google mogło trwać nawet kilka miesięcy.

Inną podstawową różnicą jest to, że PageRank wyrażony wzorem oblicza dla każdej strony WWW pojedynczą wartość, w odróżnieniu od wag autorytetu i huba z HITS. Wartość tą wyraża wzór (1):

$$PR(u)=\sum_{v\in B_u}\frac{PR(v)}{L(v)} \tag{1}$$

gdzie:

u – strona, dla której liczony jest PageRank,
v – strona linkująca do strony u,
B_u – zbiór wszystkich stron linkujących do strony u,
L(v) – liczna linków wychodzących ze strony v.

Dzieje się tak dlatego, że przy tworzeniu PageRank przyjęto nieco inną intuicję dla analizy struktury hiperłączy. Wartość PageRank dla każdej strony można interpretować jako prawdopodobieństwo dotarcia do danej strony poprzez losowe klikanie w hiperłącza na kolejnych stronach WWW. Z tego względu suma wartości PageRank dla wszystkich stron równa się jeden. Aby zapobiec problemowi zapętlenia się użytkownika poprzez dłuższe klikanie kilku linków tworzących cykl, dodatkowo z odpowiednim prawdopodobieństwem losowe klikanie może ponownie rozpocząć się od nowej losowej strony. Takie rozwiązanie problemu cyklu ma symulować znudzenie użytkownika, który po pewnym czasie klikania wpisuje nowy URL lub korzysta z zakładek.

Aby obliczyć wartości PageRank, podobnie jak w przypadku HITS użyty został algorytm iteracyjny. Autorzy empirycznie oszacowali, że aby osiągnąć zbieżność bazy ze 161 milionami linków wystarczy około 45 iteracji, a dla bazy o o połowę większej liczbie linków (322 milionów) jej zbieżność można osiągnąć po około 52 iteracjach. Te wyniki mają świadczyć na korzyść skalowalności algorytmu. Same operacje wykonywane podczas iteracji to uaktualnianie wartości PageRank dla każdej strony na podstawie PageRanków stron do niej linkujących. PageRank dla strony uaktualniany jest jako suma PageRanków przekazanych przez strony mające hiperłącza do tej strony, przy czym dana strona może przekazać tylko część swojej wartości PageRank dystrybuując ją równomiernie na każdy link z niej wychodzący.

Na przykład, jeśli strona A ma hiperłącze do strony B, ale linkuje również do trzech innych stron, to obliczając PageRank dla B bierzemy pod uwagę tylko ¼ wartości PageRank dla A. Obliczona w ten sposób wartość PageRank jest dodatkowo pomniejszana przez współczynnik wygaszania *d* (ang. *damping factor*), zgodnie ze wzorem wzór (2):

$$PR(p_i)=\frac{1-d}{N}+d\sum_{p_j\in M(p_i)}\frac{PR(p_j)}{L(p_j)} \tag{2}$$

gdzie:

p_i *– strona, dla której liczony jest PageRank,*
d – współczynnik wygaszania,
N – całkowita liczba stron,
p_j *– strona linkująca do strony* p_i*,*
$M(p_i)$ *– zbiór wszystkich stron linkujących do strony* p_i*,*
$L(p_j)$ *– liczna linków wychodzących ze strony* p_j *.*

Empirycznie wyznaczona wartość $d = 0{,}85$ ma oznaczać wspomniane prawdopodobieństwo z jakim użytkownik na każdej stronie będzie kontynuował losowe klikanie zamiast zacząć od nowej, losowej strony.

2.4.5 Metody linkowania serwisów WWW i SEO

Głównym elementem, któremu WWW zawdzięcza swoją popularność jest możliwość linkowania między serwisami. Początkowo, gdy wyszukiwarki WWW nie były jeszcze zbyt wydajne, stosowano katalogi, które zbierały i grupowały serwisy tematycznie. W momencie pojawienia się wyszukiwarek opierających się na analizie struktury linków okazało się, że ich efektywność jest na tyle wysoka, że nie opłaca się utrzymywać ręcznie aktualizowanych katalogów WWW. Ale wraz z efektywnymi wyszukiwarkami WWW pojawił się problem rankingu serwisów. Najpopularniejsza metoda rankowania - PageRank - stała się obiektem ataków ludzi chcących wypromować swoje strony w rankingu Google'a. Powstała nawet osobna dziedzina wiedzy, zajmująca się tzw. pozycjonowaniem stron w rankingach różnych wyszukiwarek. Dziedzina ta nazywana jest SEO (ang. *Search Engine Optimization*) (Google 2008). Metody SEO dotyczą podejmowania przez Webmasterów akcji mających na celu poprawienie oceny w rankingu wyszukiwarek. Akcje te można podzielić na te działające w zgodzie z założeniami algorytmu rankującego, czyli tzw. *white hat SEO* oraz na akcje próbujące sztucznie zawyżyć ocenę strony, czyli tzw. *black hat SEO*. Te pierwsze dotyczą m.in. poprawnego formatowania i linkowania dokumentów, właściwego używania słów kluczowych i trzymania się standardów W3C.

Ponieważ kluczowym elementem oceny PageRank dla danej strony jest liczba przychodzących linków, dlatego chcąc znaleźć się na pierwszych pozycjach wyników z wyszukiwarki Google strona musi posiadać linki odsyłające do niej, umieszczone na innych stronach, albo nawet lepiej, w innych serwisach (domenach). Internauci chcąc promować swoje serwisy próbowali wpłynąć algorytm PageRank linkując wzajemnie swoje strony WWW. Takie linkowanie nazywane jest linkowaniem wzajemnym (ang. *reciprocal link*). W WWW zaczęły pojawiać się serwisy oferujące wzajemne linkowanie i zjawisko wymiany linków zaczęło nabierać rozpędu. Obserwując zjawisko coraz powszechniejszego linkowania wzajemnego w celu manipulowania PageRank'iem Google wprowadził kary za linki wzajemne powodujące obniżenie wartości PageRank dla wzajemnie linkujących stron. Wtedy pojawił się kolejny pomysł na oszukanie algorytmu. Linkowanie pośrednie czy przechodnie (ang. *3-way linking*) to sposób na utrudnienie wykrycia linków wzajemnych. Polega ono na wykorzystaniu pośrednictwa trzeciej strony. Typowy link wzajemny między stronami A i B można zobrazować tak: A → B → A. Pośrednie linkowanie to A → B → C → A. W ten sposób strony korzystają z linków przychodzących, a algorytmom jak PageRank trudniej jest je wykryć. Większym uogólnieniem tego pomysłu są *webring*'i, czyli pierścienie WWW. Można zauważyć, że pośrednie, 3-elementowe linkowanie tworzy trójkąt. *Webring* tworzy wielokąt o jeszcze większej liczbie boków/linków. Od określonej liczby boków, wykrycie takich linków wzajemnych przestaje być praktycznie możliwe.

2.4.6 Giant Global Graph

Wracając do modelowania za pomocą grafu, należy zauważyć, że część źródeł w definicji pojęcia „graf" zamiast terminu „zbiór wierzchołków" używa terminu „kolekcja wierzchołków". Fakt ten pokazuje jak bardzo zbiór wierzchołków grafu jest utożsamiany z kolekcją dokumentów

hipertekstowych. Najnowsze badania skupiają się jednak na odwzorowaniu WWW, gdzie podstawową jednostką nie jest dokument, a użytkownik – najczęściej autor dokumentu. W świetle koncepcji nazwanej przez Tima O'Reilly (O'Reilly 2005) terminem „Web 2.0" to nie związki między dokumentami są ciekawe, tylko związki między ludźmi, które dzięki WWW możemy analizować. Z perspektywy antropologicznej obrazuje to film etnologa i ekologa mediów Michaela Wesch'a (Wesch 2007), którego konkluzja: „*We are the Web* (My jesteśmy WWW)" jest nowym wyznacznikiem kierunku badań naukowych i marketingowych.

Pomimo zmiany sposobu postrzegania WWW i jej elementów model grafu pozostaje najwygodniejszy i najpopularniejszy. Pod koniec 2007 Berners-Lee – twórca teorii WWW i Semantic Web - napisał o społecznym aspekcie WWW: "Ja nazwałem ten graf *Semantic Web*, ale może powinienem go nazwać *Giant Global Graph*!" (Tim Berners-Lee 2007b). Cały post ukazywał kierunek zmian i rozwoju WWW. Na początku powstała Sieć (*the Net*), która pozwoliła komputerom widzieć się, bez dostrzegania kabli. Wkrótce okazało się, że to nie komputery są interesujące, ale dokumenty, które się na nich znajdują. Na bazie Sieci powstała więc Pajęczyna (*the Web*). Dzięki Pajęczynie możliwe jest nawigowanie po morzu dokumentów nie martwiąc się o to, na którym komputerze dokumenty się znajdują. Sieć i Pajęczyna mogą być stworzone na kształt czegoś, co matematycy nazywają grafem, ale są one na innym poziomie. Sieć łączy komputery, Pajęczyna łączy dokumenty. Kolejnym krokiem było dostrzeżenie, że to nie dokumenty są interesujące, ale tematyki, które są w nich poruszane. Jednak jednakowe tematyki poruszane w różnych serwisach nie były przez system postrzegane jako powiązane. Podobnie profil osoby w jednym serwisie społecznościowym nie jest przez system wiązany z profilem tej osoby w innym serwisie. Stąd okazało się, że to nie serwisy społecznościowe, w których tematyki są poruszane są interesujące, ale sama sieć społeczna – Społeczny Graf (*the Social Graph*). Sposób w jaki użytkownik jest połączony, a nie jego strony. W tym miejscu Berners-Lee proponuje użycie pojęcia „Graf", żeby odróżnić je od Pajęczyny, czyli WWW.

3 Sieć semantyczna

W 2001 Berners-Lee opublikował (Tim Berners-Lee, Hendler, i Lassila 2001), w którym opisał swoją wizję kolejnej generacji WWW. Wizję tę nazwał „Semantyczna WWW" (ang. *Semantic Web*). Główną ideą, która przyświeca wizji Semantycznej WWW jest możliwość przetwarzania WWW przez maszyny. Aby zrealizować ten cel należało przyjąć pewną ścisłą składnię, dzięki której komputery mogłyby przetwarzać dane według niej sformatowane. Głównym problemem okazała się luźna adaptacja standardu HTML przez przeglądarki, powodującą, że umieszczane w sieci dokumenty pomimo poprawnej prezentacji rzadko trzymały się ścisłej składni HTML. Często pojawiał się problem nie domykania znaczników, krzyżowania znaczników używania niedozwolonych lub działających tylko w danej przeglądarce atrybutów, itp. Główną winę za tego typu praktyki ponosi monopol Internet Explorera w latach 90-tych, który pozwalał twórcom dokumentów na taką nieelegancję kodu poprzez „domyślanie się" poprawnej składni, której autor powinien był użyć. Dodatkowo przeglądarka ta umożliwiała na stosowanie niestandardowych znaczników działających tylko w tej przeglądarce. Ponieważ rynek był zdominowany przez jedną przeglądarkę webmasterzy często nie przejmowali się poprawnością swojego kodu. Jedyną weryfikacją kodu, zamiast zgodności ze standardami, było poprawne renderowanie się dokumentu w najpopularniejszej wówczas przeglądarce.

Taka praktyka powodowała, że tworzone treści mogły być wykorzystywane jedynie przez ludzi. Trudno bowiem oczekiwać, aby każdy program mający analizować dokumenty w WWW musiał posiadać silnik renderujący jak przeglądarka. Zwłaszcza, że strona renderowana w jednej przeglądarce mogła wyglądać zupełnie inaczej w innej przeglądarce (patrz Testy Acid w podrozdziale 1.2.5, s. 13). Aby uściślić składnię przyjęto, że nowa wersja HTML będzie aplikacją XML'a przeznaczonego do wymiany danych, a więc łatwo przetwarzalnego przez maszyny. Mając taką podstawę popularne zaczęło być używanie semantyki zarówno dostępnej w standardzie jak i rozszerzanej przez społeczność WWW. W szkicu roboczym XHTML 2.0 (Axelsson et al. 2006) nadal znajduje się odwołanie do semantyki atrybutu `rel` wprowadzonego w HTML 4.0 (patrz 2.2.3 Typy linków s. 36). Ten atrybut stał się podstawą rozwijania „ministandardów" semantycznych.

Mikroformaty. Przykładem standaryzowania semantyki wyrażanej za pomocą XHTML są mikroformaty (ang. *microformats*) opisane w Allsopp 2007). Obecnie istnieje kilkanaście mikroformatów, a kolejne są na etapie opracowywania. Pozwalają one wzbogacić strony WWW o informację semantyczną w celu umożliwienie automatycznego przetwarzania publikowanych informacji. Używając głównie XHML'owych atrybutów `class i rel` pozwalają oznaczyć semantycznie np.: zdarzenia (*hCalendar*), kontakty (*hCard*), koordynaty geograficzne (*geo*), życiorysy (*hResume*), pojęcia folksonomii[22] (*rel-tag*) czy więzi społeczne (*XFN*).

3.1 Sieć społeczna

Dzięki fali popularności serwisów społecznościowych powstała nowa możliwość badania sieci społecznej: przez jej model w serwisach WWW. Serwisy społecznościowe to najczęściej zamknięte, komercyjne systemy, dostęp do ich danych jest utrudniony. Poza tym dochodzi problem poufności danych osobowych oraz rozrzucenia sieci społecznej po różnych serwisach, których mapowanie w celu uzyskania całościowego widoku sieci dodatkowo komplikuje analizę. Dlatego praca badawcze dotyczące sieci społecznych korzystają z jawnie i publicznie udostępnionych związków zadeklarowanych przez użytkowników.

FOAF. Najpopularniejszym formatem zapisu związków między osobami jest standard FOAF

22 Folksonomia (wspólne tagowanie, społeczna klasyfikacja, społeczne indeksowanie, społeczne tagowanie) – neologizm oznaczający praktykę niehierarchicznej kategoryzacji treści z wykorzystaniem dowolnie dobranych słów kluczowych - (Voß 2007)

(Brickley i Libby Miller 2007). Nazwa pochodzi od skrótu Przyjaciel Przyjaciela (ang. *Friend of a Friend*). Ponieważ FOAF jest ontologią, zapisywany jest on w standardzie RDF. Stosując FOAF można opisać osobę poprzez np. nazwisko (`foaf:name`), adres e-mail (`foaf:mbox`), stronę domową (`foaf:homepage`), zdjęcie (`foaf:depiction`). Za pomocą atrybutu `foaf:knows` możliwe jest opisanie innych osób, które osoba tworząca swój profil zna. Zakładając, że osoby tworzą szczegółowo swoje profile, wystarczy, że w `foaf:knows` zdefiniowany będzie atrybut identyfikujący drugą osobę (najczęściej adres e-mail). Resztę danych o drugiej osobie można znaleźć w pliku FOAF jej autorstwa. Z założenia, że każdy użytkownik opisuje siebie i swoje znajomości i umieszcza ten opis w sieci, wynika zdecentralizowana forma zarządzania związkami i możliwość stworzenia globalnego grafu tych związków.

XFN. Mniej znanym, ale zdobywający popularność formatem zapisu w WWW związków między osobami jest chronologicznie pierwszy mikroformat XFN (GMPG 2009). Jak wskazuje nazwa XHTML'owa Sieć Znajomości (ang. *XHTML Friends Network*) zapis semantyki związków umieszcza się w samym dokumencie XHTML, a nie - jak w przypadku FOAF - w podłączonym pliku RDF. Dzięki temu najpopularniejszym zastosowaniem XFN są *blogroll*'e, czyli listy blogów umieszczanie przez autorów na swoich blogach. Ponieważ *blogroll* jest listą blogów, które autor czyta, poleca lub jest związany z nimi w inny sposób, najczęściej w świecie rzeczywistym istnieje między nim i autorami tych blogów pewien związek. Ten fakt został wykorzystany przez twórców XFN, którzy opracowali standardowe typy związków oraz ich cechy: symetryczność, przechodniość i odwrotność. Korzystając z XFN można nie tylko - jak w przypadku FOAF – zaznaczyć swój związek z autorem linkowanego bloga, ale umieścić w atrybucie `rel` typ tego związku. Autorzy pogrupowali następujące wartości (lub ich kombinacje) dla atrybutu `rel`:

- *friendship* (najwyżej jedna wartość)
  - *contact* – ktoś, z kim wiesz jak się skontaktować; często symetryczna,
  - *acquaintance* – ktoś, z kim wymieniłeś pozdrowienia i ewentualnie odbyłeś ze dwie rozmowy; często symetryczna,
  - *friend* – ktoś, dla kogo jesteś przyjacielem; często symetryczna,
- *physical*
  - *met* – ktoś, kogo spotkałeś osobiście; symetryczna,
- *professional*
  - *co-worker* – ktoś, z kim pracujesz lub pracuje w tej samej organizacji; symetryczna, zwykle przechodnia,
  - *colleague* – ktoś w studiujący/pracujący w tej samej dziedzinie; symetryczna; często przechodnia,
- *geographical* (najwyżej jedna wartość)
  - *co-resident* – ktoś, z kim dzielisz ulicę; symetryczna i przechodnia,
  - *neighbor* – ktoś, kto mieszka niedaleko, np. pod przylegającym adresem lub w sąsiednim korytarzu; symetryczna; często przechodnia,
- *family* (najwyżej jedna wartość)
  - *child* – genetyczny potomek osoby lub ktoś zaadoptowany; odwrotna do *parent*
  - *parent* – odwrotna do *child*
  - *sibling* – ktoś, z kim dzielisz rodzica; symetryczna, zwykle przechodnia,
  - *spouse* – ktoś, kogo poślubiłeś; symetryczna, nie przechodnia,
  - *kin* – krewny, ktoś, kogo uważasz, za członka swojej rozszerzonej rodziny; symetryczna i przeważnie przechodnia,
- *romantic*
  - *muse* – ktoś, kto przynosi ci inspirację; brak odwrotnej,
  - *crush* – ktoś w kim się zadurzyłeś; brak odwrotnej,
  - *date* – ktoś, z kim chodzisz na randki; nie przechodnia,
  - *sweetheart* – ktoś z kim masz intymny związek i przynajmniej jakieś zobowiązanie,

zwykle wyłączne; symetryczna, nie przechodnia,

- *identity*
  - *me* – link do siebie pod innym adresem. Nie może występować z innymi wartościami XFN. Wymagana symetryczność. Istnieje niejawny związek *me* z zawartości katalogu do samego katalogu.

Na stały wzrost popularności XFN ma wpływ tworzenie i integrowanie narzędzi ułatwiających użytkownikom deklarowanie związków podczas dodawania linków np. do *blogroll*'i. Niektóre serwisy społecznościowe udostępniają wtyczki umożliwiające wprost opublikowanie sieci znajomych w formacie XFN. Popularność formatów zapisu związków to jedyny sposób na rozpowszechnienie informacji o związkach pozwalające analizować rzeczywiste sieci społeczne w WWW.

Na przykład w Ding, Zhou, et al. 2005) udało się pobrać z WWW 2 miliony plików RDF zawierających deklaracje FOAF. 1,5 miliona plików pochodziło z serwisów blogowych, a 5000 z innych źródeł. Analizie poddano jedynie te ostatnie 5000, ponieważ pliki z blogów tworzone były najczęściej przez automat bloga. Z kolei 5000 plików z innych źródeł zostało stworzone ręcznie, a różnorodność używanych w nich struktur i słownictwa dowodzi, że powstały świadomie. W momencie analizy (czerwiec 2004) autorzy sprawdzili, że FOAF był drugą najczęściej używaną ontologią. Na pierwszym miejscu znalazła się ontologia RDF, a popularny - wydawałoby się RSS – na miejscu 6. Dokumenty FOAF zostały podzielona na: ścisłe i ogólne. Pierwsze opisywały tylko jedną osobę (i jej znajomych), a drugie mogły szczegółowo opisywać wiele osób. Do modelowania sieci FOAF użyto grafu skierowanego, w którym węzłami są osoby, a krawędziami deklaracja znajomości. Łączenie informacji FOAF w celu zbudowania globalnego grafu odbywało się poprzez znajdowanie wspólnych węzłów w grafach z poszczególnych plików. Osoby - wspólne węzły identyfikowane były na podstawie pola *foaf:mbox*, czyli adresu e-mail. Oczywiście liczba dostępnych informacji nie pozwoliła na zbudowanie spójnego grafu, a spójne podgrafu najczęściej miały postać gwiazdy. Jak się okazało tylko 7% węzłów miało zarówno krawędzie wychodzące i przychodzące, a 97,7% węzłów posiadających tylko krawędzie przychodzące było incydentnych z tylko jedną krawędzią.

Mając za mało danych wejściowych w poprzedniej analizie autorzy (Ding, Finin, i Joshi 2005) postarali się wyznaczyć problemy do rozwiązania zanim będzie można analizować sieci społeczne za pośrednictwem sieci semantycznej. Podstawowe grupy problemów to:

- reprezentacja wiedzy – ze względu na fakt zdecentralizowania ontologii pojawia się tu szereg podproblemów
- zarządzanie wiedzą – podobnie jak powyżej, zdecentralizowanie rzutuje na to, że na poziomie grafów RDF połączenia są gęste, ale na poziomie dokumentów zawierających te grafy o wiele rzadsze
- ekstrakcja, integracja i analiza sieci społecznych – wiedza powstała jest zaszumiona, niepełna i nie zawsze wiarygodna
- wnioskowanie świadome pochodzenia i zaufania do danych – problem wiarygodności dodatkowo poszerza problem wnioskowania

Ponieważ jak zauważyli autorzy pole *foaf:Person* pojawia się z 17 ontologiach, więc oprócz FOAF można użyć innych dokumentów RDF w celu analizy sieci społecznej. W pracy wzięto pod uwagę 2 zbiory danych:

- DS-SWOOGLE zawierające 255 tysięcy dokumentów zawierających 37 milionów trójek RDF (po odfiltrowaniu co najwyżej 10 tysięcy trójek z jednego serwisu)
- DS-FOAF, który przed analizą został zmniejszony do DS-FOAF-VAR, poprzez odrzucenie dokumentów FOAF z witryn zawierających więcej niż tysiąc takich dokumentów (ze względów dotyczących automatu, wspomnianych wcześniej); ostatecznie zawierający 4 tysiące ścisłych dokumentów FOAF i 37 tysięcy instancji pola *foaf:Person*

Podsumowując autorzy zaznaczyli, że celowe było by połączenie, czy nałożenie na siebie różnych warstw sieci społecznych uwzględniając: sieć FOAF, systemy reputacji (np. *PageRank*), sieć zaufania i indeksy współautorstwa (np. *DBLP*).

3.1.1 Sieć zaufania

Wspomniana warstwa sieci zaufania wynikała z równolegle prowadzonych badań nad sieciami społecznymi w WWW. W analogiczny sposób do tego jak HITS czy PageRank wyznaczają strony autorytatywne na podstawie sieci dokumentów, zaczęto analizować sieci społeczne w celu znalezienia lokalnego i globalnego autorytetu w postaci osoby. Same metody wyznaczania takich osób były już w socjologii znane od dawna, jednak w kontekście Internetu nabrały nowego wymiaru.

W pracy (Pujol, Sangüesa, i Delgado 2002) autorzy przeprowadzili testy algorytmu oceniającego ludzi na podstawie ich reputacji. Sieć społeczna użyta do testów powstała poprzez losowy wybór 34 osób z Politechniki Katalońskiej. Sieć zamodelowano grafem nieskierowanym sumując wagi krawędzi w obie strony. Pierwotne wagi krawędzi to wspólne linki i zasoby na stronach domowych osób oraz wspólne e-maile. Testowany algorytm podobnie jak PageRank początkowo zakłada jednakowy stopień autorytetu węzła, a wnioskuje reputację każdego węzła na podstawie autorytety węzła i węzłów na niego wskazujących. Algorytm nie musi znać całego grafu, ponieważ działa lokalnie w sposób asynchroniczny. Kolejne iteracje wyznaczania reputacji są zbieżne i kończą się dla danego węzła, gdy jego funkcja zbieżności nie przekracza zadanego progu. Wzorem dla testowanego algorytmu jest ocena z systemu CiteSeer. Dla porównania testowane dla tego grafu są również algorytmy PageRank i HITS. Ze względu na teorię małych światów graf jest daleki od grafu pełnego, więc aby uniknąć problemu ślepych ścieżek stosowany jest losowy skok o prawdopodobieństwie 0,54. Testowany algorytm wypadł najlepiej, ale jak podkreślają autorzy, gdyby losowy skok w PageRank ustawić również większy od ½ (zwykle jest to: 1 - *współczynnik wygaszania* = 0,15), to wyniki byłyby bardzo zbliżone. Więc jedyną zaletą testowanego algorytmu jest fakt, że działa on lokalnie, w odróżnieniu o PageRanka działającego globalnie i wymagającego znajomości całej sieci.

Z kolei autorzy (Golbeck, Parsia, i Hendler 2003) stworzyli sieć zaufania dodając do ontologii FOAF 9-stopniową skalę zaufania w danym kontekście. Mając dane zaufanie bezpośrednie, odczytywane z deklaracji FOAF, zaufanie pośrednie wyznaczane jest poprzez iloczyn zaufań w ścieżce. Co więcej umożliwiono przeliczanie zaufania między dwoma dowolnymi osobami (identyfikowanymi przez adres e-mail) według własnego algorytmu. Możliwość ta została zaimplementowana jako Web service, a algorytm przeliczania należy podać jako funkcję korzystającą z Java'owego API dającego dostęp do grafu zaufania. Dodatkowo stworzono dwie implementacje wykorzystujące obliczane zaufanie. Pierwsza to agent sieci IRC, który można odpytywać o średnie, minimalne i maksymalne długości ścieżek między dwoma węzłami w grafie zaufania oraz o ich przepustowość. Druga implementacja to funkcjonalność klienta pocztowego, dzięki której przy temacie wiadomości pokazują się wyliczona wartości zaufania do ich nadawców.

3.1.2 Ranking i filtrowanie w sieci zaufania

Rozszerzeniem pomysłu zastosowania sieci zaufania do automatycznego oceniania nowych wiadomości jest spersonalizowana sieć zaufania, która umożliwia automatyczne ocenianie dokumentów WWW. Taki pomysł został opisany w Kopel i Kazienko 2007). Sieć zaufania osadzona jest w środowisku wielo-agentowym MAS (ang. *Multi-Agent System*). Każdy agent ma za zadanie utrzymanie **osobistej sieci zaufania** swojego użytkownika. Osobiste sieci zaufania, z kolei, składają się na globalną sieć zaufania WoT (ang. *Web of Trust*)

Metoda zaproponowana w pracy opiera się na idei FOAF, ale nie bezpośrednio na danych w tym formacie. Podobnie jak w przypadku wcześniejszych prac w tej tematyce, informacje na temat zadeklarowanych związków między użytkownikami służą wnioskowaniu i wyprowadzaniu nowych

związków i w ten sposób budowania globalnej WoT. Dodatkowo pod uwagę brane są oceny użytkownika na temat dokumentów oraz powiązania tych dokumentów z ich autorami. Jak wspomniano wcześniej sieć zaufania ma służyć proponowaniu ocen dla dokumentów WWW. Propozycje ocen można uznać za ranking, przy czym nie musi to być ranking wyników wyszukiwania, ale ranking nowych dokumentów pojawiających się w subskrybowanych przez użytkownika źródłach. Takie zastosowanie jest odpowiedzią na zalewanie użytkownika nowymi dokumentami, który na przykład śledzi wiele blogów za pomocą RSS. Ponieważ czytnik RSS z nowymi dokumentami bardzo przypomina klienta poczty, więc sama idea ponownie wydaje się w kontekście tego przykładu być powieleniem pomysłu z (Golbeck, Parsia, i Hendler 2003). Jednak ze względu na uniwersalność technik RSS oraz ogromną liczbę stale aktualizowanych źródeł jej zakres zastosowania jest o wiele szerszy. Poza tym sama metoda, w odróżnieniu od porównywanej, dzięki wykorzystaniu MAS, działa w sposób rozproszony.

Sam pomysł zastosowania WoT do rankowania dokumentów jest bezpośrednio przejęty z metodologii **filtrowania kolaboratywnego** CF (ang. *collaborative filtering*). Główne założenie CF mówi, że osoby, które zgodziły się w przeszłości najpewniej zgodzą się ponownie w przyszłości. Czyli jeśli zaufanie oznacza zgodność w ocenie dokumentów z inną osobą, to oceny tej osoby nowych dokumentów mogą być dla nas jak najbardziej odpowiednie. Dodatkowo korzystając z idei FOAF i wyciągając pośrednie zaufania do osób trzecich jesteśmy w stanie stworzyć spersonalizowany ranking prawie wszystkich nowych dokumentów.

Aby zrealizować ideę CF w połączeniu z WoT w (Kopel i Kazienko 2007) wzięto pod uwagę zarówno oceny dokumentów, jak i reputację użytkowników tworzących lub oceniających te dokumenty. Dlatego w metodzie użyto dwa rodzaje miar: zaufanie jednego użytkownika do drugiego oraz ocenę konkretnego dokumentu przez konkretnego użytkownika. Autorstwo danego dokumentu również jest traktowane jako ocena tego dokumentu zakładając, że autor wystawiłby swojemu dokumentowi najwyższą ocenę.

Zaufanie do użytkownika to wartość ustawiana ręcznie i świadomie przez użytkownika na podstawie wcześniejszych doświadczeń, interakcji z użytkownikiem i jego dokumentami lub znajomości ze świata rzeczywistego. Zaufanie może odzwierciedlać więzy rodzinne, podobne zainteresowania lub traktowanie danej osoby jako autorytetu. Ponieważ każdy użytkownik sam ustala wartość zaufania do drugiego, więc związek zaufania jest związkiem asymetrycznym. Ustawiając ręcznie zaufanie do innych użytkownik buduje swoją osobistą sieć zaufania. Czyli osobista WoT składa się z osób, do których użytkownik zadeklarował zaufanie. Ustawiając zaufanie dla kolejnych osób, użytkownik dodaje je do swojej osobistej WoT.

Metoda ma pozwolić automatycznie tworzyć spersonalizowany ranking najczęściej nowych dokumentów stworzonych przez użytkowników spoza osobistej WoT. Aby móc wyciągnąć z globalnej WoT oceny tych nowych dokumentów należy oszacować zaufanie do autorów tych dokumentów lub użytkowników, którzy już je ocenili. A więc, poza ręcznie ustawianym zaufaniem użytkownika (T^{usr}), agent przechowuje również zaufanie wywnioskowane na podstawie komunikacji z agentami innych użytkowników. Takie zaufanie agenta a_i do użytkownika a_j jest oznaczane $T^{agn}(a_i \rightarrow a_j)$ i przeliczane według wzoru (3):

$$T^{agn}(a_i \rightarrow a_j) = \frac{\sum_{a_k \in PWoT(a_i)} T^{usr}(a_i \rightarrow a_k) \cdot T^{rsp}(a_k \rightarrow a_j)}{\sum_{a_k \in PWoT(a_i)} T^{usr}(a_i \rightarrow a_k)} \tag{3}$$

Ponieważ agent działa w imieniu użytkownika, a sam użytkownik jest jednocześnie autorem dokumentów, dlatego pojęcia „zaufanie do użytkownika", „zaufanie do agenta", „zaufanie do autora" są w tym wypadku równoważne i używane zamiennie w różnych kontekstach. Różne znaczenia z kolei mają pojęcia „zaufanie użytkownika" i „zaufanie agenta" oznaczające dwie różne wartości: ustawianą ręcznie i wyliczaną przez agenta.

Aby wyliczyć $T^{agn}(a_i \rightarrow a_j)$ agent a_i odpytuje wszystkie agenty a_k ze swojej sieci zaufania. Te, jeśli mają zaufanie użytkownika do autora a_j, to od razu odpowiadają. Jeśli nie, to zapytanie jest przekazywane rekurencyjnie od wszystkich agentów w sieciach zaufania agentów a_k, zgodnie ze wzorem (4):

$$T^{rsp}(a_k \rightarrow a_j) = \begin{cases} T^{usr}(a_k \rightarrow a_j), & \text{jeśli } T^{usr}(a_k \rightarrow a_j) \text{ jest znane} \\ \lambda_k \cdot T^{agn}(a_k \rightarrow a_j), & \text{jeśli } T^{usr}(a_k \rightarrow a_j) \text{ jest nieznane} \\ & \text{i } T^{agn}(a_k \rightarrow a_j) \text{ jest znane} \\ 0, & \text{w przeciwnym przypadku} \end{cases} \quad (4)$$

Pierwszym problemem wynikającym z braku natychmiastowej odpowiedzi jest wartość odpowiedzi od agentów trzecich. Aby rozróżnić zaufanie użytkownika T^{usr} od zapytania agenta T^{agn} wprowadzono współczynnik $\lambda \in [0,1]$. Współczynnik, ustawiany dla każdego użytkownika niezależnie, świadczy na ile ważniejsze dla użytkownika jest ustawione przez niego zaufanie od zaufania proponowanego przez jego agenta. Na przykład $\lambda_k = ½$ oznacza, że zaufania użytkownika a_k mają dwa razy większą wartość niż zaufania szacowane przez agenta a_k i pośrednio przez agentów przez niego odpytywanych.

Drugim problemem pojawiającym się przy rekurencyjnym przesyłaniu zapytań w globalnej WoT jest problem znany z sieci komputerowych: takie zapytania mogą krążyć w nieskończoność. Dlatego zastosowane rozwiązanie zainspirowane jest rozwiązaniem tego problemu z pakietami w sieciach TCP/IP. Każde zapytanie ma ustawioną wartość TTL (ang. *Time To Live*), która określa czas życia zapytania. Wartość TTL każdego zapytania jest zmieszana o 1 przy każdym przekazaniu dalej. W momencie, gdy TTL zapytania osiąga 0 zapytanie nie jest dalej przekazywane. Zależnie od sieci domyślną wartość TTL można wyznaczyć empirycznie, jednak zgodnie z hipoteza 6 stopni oddalenia[23] wydaje się, że wartość 6 powinna być wystarczająca.

Drugim rodzajem miar, obok zaufania do użytkowników, są oceny dokumentów. Podobnie, jak w przypadku zaufania rozróżniono wartości ustawiane ręcznie przez użytkownika i wyliczane automatycznie przez agentów, tak i w przypadku ocen mamy ocenę użytkownika (R^{usr}) i ocenę agenta (R^{agn}). Ocena użytkownika to wartość nadana dokumentowi przez użytkownika ręcznie lub jawnie zaakceptowana wartość, proponowana jako ocena agenta. Same oceny agenta wyliczane są w sposób analogiczny do zaufania. Ocena dokumentu d_j przez agenta a_i wyliczana według wzoru (5):

$$R^{agn}(a_i \rightarrow d_j) = \frac{\sum_{a_k \in PWoT(a_i)} T^{usr}(a_i \rightarrow a_k) \cdot R^{rsp}(a_k \rightarrow d_j)}{\sum_{a_k \in PWoT(a_i)} T^{usr}(a_i \rightarrow a_k)} \quad (5)$$

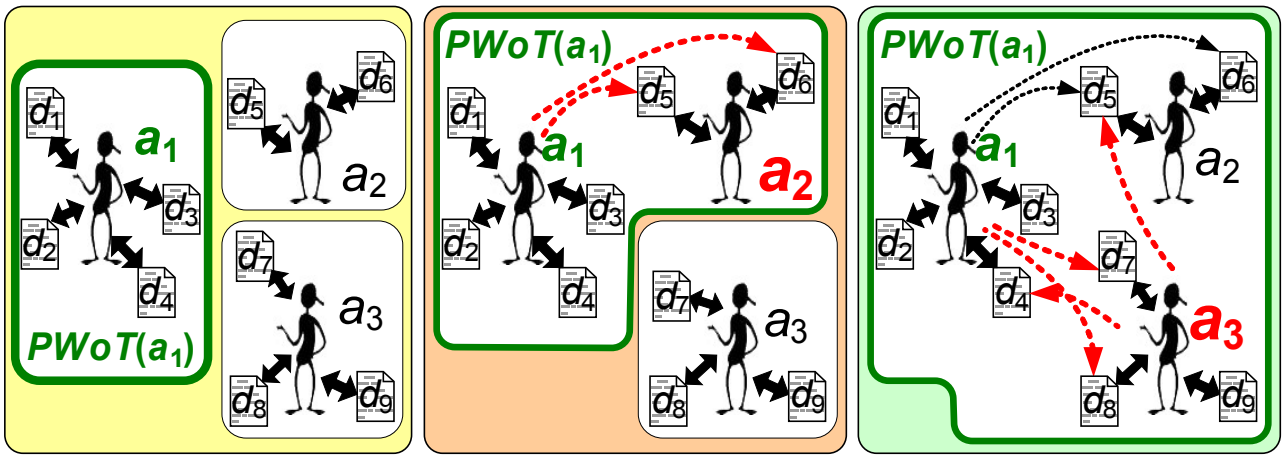
Odpowiedź agenta a_k na zapytanie o ocenę dokumentu d_j obliczana jest według wzoru (6):

23 Hipoteza 6 stopni oddalenia (ang. *six degrees of separation hypothesis*) to hipoteza potwierdzona m.in. w eksperymencie „małych światów" w Milgram 1967). Załóżmy, że jendym stopniem oddalenia pomiędzy osobami nazwiemy bezpośrednią znajomość, a dwoma stopniami oddalenia jednego bezpośrednio znajomego łączącego dwie osoby. Kolejne stopnie oddalenia to łańcuchy wzajemnie bezpośrednich znajomych łączących dwie osoby. Hipoteza 6 stopni oddalenia stwierdza, że każdy człowiek na świecie jest oddalony od każdego innego o nie więcej niż 6 stopni oddalenia.

$$R^{rsp}(a_k \to d_j) = \begin{cases} 1, & jeśli\ Ath(a_k \to d_j)\ jest\ prawdziwe \\ R^{usr}(a_k \to d_j), & jeśli\ R^{usr}(a_k \to d_j)\ jest\ znane \\ \tau_k \cdot T^{agn}(a_k \to a_j), & jeśli\ R^{usr}(a_k \to d_j)\ jest\ nieznane \\ & i\ R^{agn}(a_k \to d_j)\ jest\ znane \\ 0, & w\ przeciwnym\ przypadku \end{cases} \quad (6)$$

Współczynnik τ_k jest odpowiednikiem λ_k ze wzoru , oznaczający ważność ocen użytkownika a_k w stosunku do ocen innych użytkowników z jego osobistej WoT.

Przykład. Przyjrzyjmy się sytuacji zobrazowanej na rysunku 3.1.. Użytkownik a_1 jest autorem dokumentów d_1-d_4. Początkowo jego osobista sieć zaufania $PWoT(a_1)$ nie zawiera żadnych osób zaufanych (żółty prostokąt). Po przeczytaniu wpisu d_5 w blogu użytkownika a_2 i obejrzeniu nakręconego przez a_2 filmiku d_6 użytkownik a_1 ocenia te dokumenty. Po a_1 chwili ustawia zaufanie do użytkownika a_2 i przez to włącza go do swojej WoT (różowy prostokąt). Po pewnym czasie a_1 ocenia inne dokumenty (d_7, d_8) kolejnego użytkownika. Użytkownik ten okazuje się być wspólnym znajomym a_1 i a_2, który zdążył już obejrzeć i ocenić dokumenty znajomych (d_4, d_6). Użytkownik a_1, ustawiając zaufanie do a_3 i jego przyszłych ocen, dodaje go do swojej WoT (zielony prostokąt).



Rysunek 3.1. Przykład rozszerzania osobistej sieci zaufania użytkownika a_1, oznaczonej $PWoT(a_1)$.

System rankingu nowych dokumentów oparty na tak działającej sieci zaufania nie rozważa semantyki ocen, a jedynie ich wartość. Autorzy dając użytkownikom wolną rękę nie precyzują co ma oznaczać ocena dokumentu. Pozostaje więc pytanie: czy wysoka ocena dokumentu przez użytkownika świadczy, że jest on dobrze napisany i wiarygodny, czy po prostu interesujący dla użytkownika. Co za tym idzie wysoka reputacja użytkownika może świadczyć, że wysoko ocenia dokumenty, że ma podobne zainteresowania lub że jest autorytetem w danej dziedzinie. Bez względu na źródło reputacji, taki sposób analizy związków między obiektami typów dokument i autor stał się inspiracją dla metody analizy spójności kolekcji dokumentów WWW.

3.1.3 Wyszukiwanie społeczne

Połączenie popularności sieci społecznych w Internecie z wyszukiwaniem informacji dało początek nowej dyscyplinie zwanej **wyszukiwaniem społecznym** (ang. *social search*). Wyszukiwanie społeczne, które ma być bardziej synchroniczną formą **filtrowania kolaboratycznego**, jest powrotem do idei ręcznego oceniania treści WWW. Od ręcznego tworzenia

katalogów (jak zaczynało Yahoo!) poprzez algorytmy HITS i PageRank, gdzie dokumenty oceniane są pośrednio przez linkowanie do nich, do dzisiejszych narzędzi pozwalających oceniać ręcznie wspólnie znajdowane dokumenty wszystko można by nazwać wyszukiwaniem społecznym. Jednak termin ten nabrał nośności wraz serwisami typu Web 2.0, gdy użytkownicy zaczęli nie anonimowo włączać się we współtworzenie WWW. Tendencję wyszukiwania tymi metodami społecznymi widać np. w posunięciach Google, które udostępniło w swojej wyszukiwarce zalogowanym użytkownikom funkcjonalność promowania lub usuwania dokumentów z wyników wyszukiwania. Innych „wielki gracz" - Microsoft – udostępnił (na razie w wersji beta) narzędzie *SearchTogether*, które jest połączeniem komunikatora i przeglądarki WWW, które pozwala grupie kontaktów na żywo oceniać i komentować wspólne wyniki wyszukiwania. Istnieje wiele podobnych projektów i ciągle powstają nowe.

3.2 Dane Linkowane i Sieć Danych

Dane Linkowane (ang. *„Linked Data"*) to termin ukuty w 2006 roku przez Tima Bernersa-Lee w notatce dotyczącej problemów projektowych Sieci Semantycznej. W Tim Berners-Lee 2007a) autor wprowadza 4 proste reguły dotyczące linkowania w „sieci danych" (ang. *web of data*). Ponieważ w *Semantic Web* nie chodzi tylko o umieszczenie w WWW danych, ale również o to, aby za pomocą linków dane mogły być eksplorowane przez ludzi lub maszyny. Dzięki „podlinkowanym danym" (ang. *linked data*), mając interesujące dane można łatwo znaleźć kolejne powiązane dane.

Podobnie do przypadku „sieci hipertekstu", czyli tradycyjnego WWW, sieć danych składa się z dokumentów w sieci. Jednak w sieci hipertekstu linki to kotwice związków w dokumentach hipertekstowych napisanych w HTMLu, natomiast w sieci danych linki łączą dowolne rzeczy opisane za pomocą RDF. URI mogą identyfikować różne rodzaje obiektów lub pojęć, ale aby zapewnić rozwój sieci poniższe oczekiwania mogą dotyczyć zarówno dokumentów HTML, jak i RDF.

1. Używaj URI jako nazw dla rzeczy.
2. Używaj HTTP URI, aby ludzie mogli sprawdzić te nazwy.
3. Gdy ktoś sprawdza URI, udzielaj użytecznych informacji.
4. Załączaj linki do innych URI, aby można było odkrywać nowe rzeczy.

Te cztery proste reguły to klucz do w pełni działającej sieci semantycznej. Jednak zadziwiająca ilość informacji nie jest współcześnie podlinkowana ze względu na nie dopełnienie jednej lub kilku z powyższych reguł.

Chociaż Berners-Lee nazywa powyższe cztery punkty regułami, to zauważa, że są to raczej wytyczne oczekiwanego zachowania. Nie zastosowanie się do nich nie psuje niczego, ale traci się możliwość rzeczywistego połączenia między danymi. To z kolei prowadzi do ograniczenia ponownego użycia w nieoczekiwany sposób. A właśnie niespodziewane ponowne użycie informacji to właśnie wartość dodana przez WWW – konkluduje autor.

Obecnie Linked Data to podstawa, na której budowane są usługi sieci semantycznej, takie jak DBpedia, Freebase czy silniki wyszukiwania za pomocą SPARQL[24]. Jak powtarzał wielokrotnie Tim Berners-Lee: „Linked Data to właściwy sposób tworzenia Sieci".

3.2.1 Grafy przeglądalne

Przy okazji omawiana czwartej reguły linkowania danych w Sieci Semantycznej Berners-Lee wprowadza pojęcie grafu „przeglądalnego" (ang. *browsable*). Istotnym wzorcem linkowania jest zbiór danych, który można eksplorować przechodząc kolejnymi linkami i pobierać dane. Sprawdzając URI węzła w grafie RDF, serwer zwraca informację o łukach wychodzących

24 SPARQL (ang. *Simple Protocol and RDF Query Language*) – język zapytań i protokół dla plików RDF. Pozwala wyszukiwać dane zawężone według kryteriów określonych poprzez predykaty RDF. W styczniu 2008 SPARQL został uznany za standard przez W3C.

i przychodzących do węzła. Inaczej mówiąc, wraz z węzłem otrzymujemy wszystkie stwierdzenia, w których ten węzeł jest podmiotem lub obiektem.

Formalnie, graf *G* jest „przeglądalny”, gdy sprawdzając URI dowolnego węzła grafu *G* otrzymamy informację opisującą węzeł, a opisanie oznacza:

1. Zwrócenie wszystkich stwierdzeń, w których ten węzeł jest podmiotem lub obiektem.
2. Opisanie wszystkich pustych węzłów[25] dołączonych do tego węzła.

W praktyce oznacza to, że każde stwierdzenie dotyczące dwóch rzeczy w osobnych dokumentach musi być powtórzone w obu dokumentach. Na przykład, jeśli w pliku FOAF znajdujemy informację o przynależności osoby do danej grupy, to dzięki powiązaniom tej grupy do innych osób, opisanym w innym dokumencie, możemy znaleźć wszystkich członków tej grupy. Jest to sformalizowanie stosowanej w WWW praktyki linkowania linkami wzajemnymi.

Dzięki grafom „przeglądalnym” możliwe jest za pomocą SPARQL tzw. „eksplorowanie”, czyli otrzymanie informacji o rzeczy w postaci trójek RDF, w których rzecz jest podmiotem lub obiektem. Czyli otrzymane trójki to fakty o danej rzeczy lub fakty o innych rzeczach związanych w dowolny sposób z tą rzeczą.

```
SELECT ?wlasnosc ?maWartosc ?jestWartoscia
WHERE {
{ <http://dbpedia.org/resource/Wroclaw> ?wlasnosc ?maWartosc  }
UNION
{ ?jestWartoscia ?wlasnosc <http://dbpedia.org/resource/Wroclaw> }}
```

Rysunek 3.2: Eksplorowanie informacji o Wrocławiu za pomocą zapytania w języku SPARQL

W przykładowym zapytaniu SPARQL z rysunku 3.2, rzeczą na temat której informacje chcemy eksplorować jest wpis w DBpedii na temat Wrocławia (`http://dbpedia.org/resource/Wroclaw`). Jak widać w rezultacie mamy otrzymać złączenie dwóch typów trójek RDF: te, w których URI Wrocławia występuje jako pierwszy element trójki (podmiot) oraz te, w których występuje jako trzeci element (obiekt). Gdy URI jest podmiotem zapytanie zwraca wszystkie wartości (`?maWartosc`) wszystkich jego własności (`?wlasnosc`). W przypadku, gdy URI jest obiektem zapytanie odnajduje wszystkie rzeczy (`?jestWartoscia`), których własności mają wartość równą temu URI.

Skoro stwierdzenia, które dotyczą rzeczy w dwóch dokumentach powinny być powtórzone w obu, pojawia się problem ze spójnością. Aby graf był całkowicie „przeglądalny” dzięki linkom wzajemnym, musi być całkowicie spójny. To wymaga koordynacji, zwłaszcza, gdy dotyczy różnych autorów czy programów. Jednym z rozwiązań, jakie proponuje Berners-Lee w (Tim Berners-Lee 2007a) jest automatyczne generowanie grafu np. z relacyjnej bazy danych, której silnik sam dba o spójność. Jednak, gdy ma się do czynienia z wieloma źródłami, trzeba pójść na kompromis. Ten najczęściej jest zdroworozsądkową odpowiedzią na pytanie: „Jeśli ktoś ma URI tej rzeczy, to informacja o jakich związkach z innymi obiektami może być dla niego użyteczna?”.

Innym aspektem tego problemu jest przeniesienie – w przestrzeń danych linkowanych – danych ogólnodostępnych w WWW w innych formatach. W (Kopel i Zgrzywa 2009) omówiono problem transponowania danych bibliograficznych w standardzie OAI-PMH[26] do formatu pozwalającego odpytywanie bazy w języku SPARQL.

25 bNode (ang. *blank node*) – węzeł w grafie RDF, który nie może być w danej chwili nazwany. Dzięki temu możliwe jest przypisanie węzłowi niektórych własności, nie definiując jeszcze jego URI.

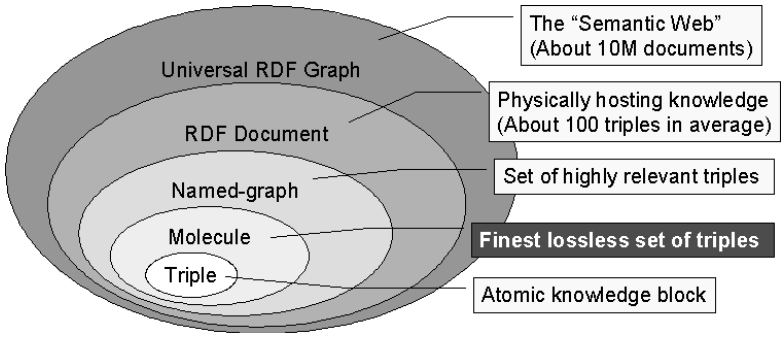
26 OAI-PMH (ang. *Open Archives Initiative Protocol for Metadata Harvesting*) – rozwijany przez OAI protokół do pobierania przez HTTP metadanych rekordów w archiwum. Celem stworzenia protokołu jest umożliwienie zbudowania usług, które będą operowały na metadanych z różnych źródeł.

3.2.2 „Danosieć”

Termin „sieć danych” zaczął być popularny w 2006 roku. Miała to być nowa nazwa dla sieci semantycznej, która w kontekście nowych aplikacji wiązała się z linkowaniem treści, czyli danych niezależnie od warstwy prezentacji. Termin powstał intuicyjnie nazywając to co powstało wskutek linkowania danych. Sama nazwa sieci danych pojawiła się wcześniej w Reed i Strongin 2004). Dla odróżnienia można by tą nazwę przetłumaczyć „Danosieć” od angielskiej nazwy *Dataweb*, w odróżnieniu od – używanej w kontekście *LinkedData – Web of Data*. Danosieć nie jest jedynie koncepcją, ale produktem działającym za pośrednictwem usługi XDI. XDI (ang. *XRI Data Interchange*) to usługa identyfikowania, wymiany, linkowania synchronizowania danych z dowolnego źródła w postaci dokumentów XML. Dodatkowym aspektem zarządzania danymi w postaci XML jest problem zarządzania dostępem do i używaniem tych danych. W tym celu XDI wprowadza nowy, kompatybilny z URI identyfikator: XRI. XRI (ang. *Extensible Resource Identifier*) to rozszerzony identyfikator zasobu WWW, który dodatkowo zawiera informację pozwalająca na zaufaną wymianę danych, analogicznie do sposobu, w jaki URI pozwalają na otwartą wymianą treści WWW.

3.2.3 Ziarnistość Semantic Web

Ze względu na rozproszenie źródeł grafów RDF problemem staje się śledzenie ich pochodzenia na odpowiednim poziomie ziarnistości. Poziom dokumentu RDF jest zbyt wysoki, ponieważ może on zawierać informacji z różnych domen, więc wzajemnie nierelewantne. Z kolei poziom trójki podmiot-predykat-obiekt jest zbyt niski. Dlatego w Ding, Finin, et al. 2005) autorzy wprowadzają pośredni poziom – poziom molekuł RDF. Uwzględniając poziom molekuł RDF, struktura sieci semantycznej wygląda jak na rys. 3.3.



Rysunek 3.3. Schemat ziarnistości Semantic Web

źródło: (Ding, Finin, et al. 2005)

Atomowym elementem wiedzy jest wspomniana trójka RDF (ang. *triple*). Odpowiedni zbiór trójek tworzy proponowaną przez autorów molekułę. Nadzbiorem molekuły jest graf nazwany zawierający ściśle związane ze sobą trójki. Grafy nazwane umieszczane są w dokumentach RDF – fizycznych kontenerach wiedzy zawierających przeciętnie ok. 100 trójek. Dokumenty RDF tworzą Sieć Semantyczną, zbiór o najmniejszym poziomie szczegółowości.

W związku z pojawieniem się pojęcia molekuła RDF, które nie ma fizycznego odzwierciedlenia w sposobie przechowywania grafów w dokumentach RDF, potrzeba metody dekompozycji globalnego grafu do molekuł. Aby to zrobić autorzy prowadzają pojęcia *lossless* i *finest*. Molekuła powinna być częścią grafu RDF spełniającą oba te pojęcia. Podgraf *lossless* oznacza, że można go użyć do odbudowy pierwotnego grafu RDF bez użycia dodatkowych trójek. Podgraf jest *finest* jeśli nie można go dalej dekomponować na podgrafy *lossless*.

4 Grupowanie wyników wyszukiwania w WWW

Tak samo, jak rozwiązywanie złożonych problemów najczęściej sprowadza się do rozbicia ich na mniejsze i prostsze podproblemy, tak i analizę złożonych danych najczęściej rozpoczyna się od podziału tych danych na mniejsze grupy (ang. *cluster*). Samo grupowanie, zwane też analizą skupień (ang. *cluster analysis*), szczególnym przypadkiem szerszej grupy problemów, zwanych klasyfikacją. Klasyfikacja, z kolei, to procedura umieszczania elementów na podstawie ich cech w grupach wcześniej zdefiniowanych przez zbiór uczący. Podobnie do teorii grafów sama taksonomia pojęć związanych z klasyfikacją i grupowaniem, jest w środowiskach naukowych dosyć płynna. Najczęściej grupowanie traktuje się jako szczególny przypadek klasyfikacji bezwzorcowej, ponieważ grupy, do których należy przypisać elementy nie są na wstępnie znane. W Kłopotek 2001) znajdujemy następujące rozróżnienie: klasyfikacja dotyczy uczenia z nadzorem (nauczyciel podaje kryterium przynależności do nazwanej klasy), a klastering (czyli grupowanie) – bez nadzoru (nauczyciel podaje funkcję przynależności do klasy).

Algorytmy grupowania dzieli się na hierarchiczne i partycjonujące. Algorytmy hierarchiczne mogą być aglomeracyjne ("z dołu do góry") lub dzielące, zwane deglomeracyjnymi, ("z góry do dołu"). Pierwsze zakładają, że każdy element jest grupą i następnie łączą je w większe grupy. Drugie dzielą początkowy zbiór na mniejsze podgrupy. Czasami obok hierarchicznych i partycjonujących wyróżnia się jeszcze grupowanie rozmyte, służące kategoryzacji, które pozwala na przypisanie elementu do kilku grup. Poza tym algorytmy grupowania można podzielić ze względu na symetryczność, bądź asymetryczność kryterium grupowania, czyli miary podobieństwa/odległości między elementami.

Grupowanie najczęściej stosowane jest w dziedzinach eksploracji danych i uczenia maszynowego. Jednak samo pojęcie grup nie musi być stosowane tylko do wyników algorytmów grupowania. Na przykład, w definicji 2.1.16 termin „grupa” dotyczy podserwisu WWW, który można traktować jako kolekcję dokumentów WWW. Czyli analizę kolekcji dokumentów, polegającą na rozpatrywaniu podkolekcji składowych czy łączeniu w nadkolekcje też można nazwać grupowaniem. Przykładem takiej analizy stosowanej nie tylko dla dokumentów WWW jest metoda LSA.

4.1 Analiza ukrytej semantyki - LSA

Opatentowana według (Deerwester et al. 1989) metoda analizy ukrytej semantyki LSA (ang. *Latent Semantic Analysis*) to technika wykorzystywana w przetwarzaniu języka naturalnego. Dzięki LSA mając jako dane wejściowe terminy indeksujące dokumenty możemy wprowadzić dodatkowy poziom powiązań pomiędzy terminami i dokumentami. Metoda LSA zamienia macierz niebinarną [terminy X dokumenty], którą można uzyskać np. za pomocą ważenia terminów tf-idf[27], na dwie macierze: [terminy X pojęcia] i [dokumenty X pojęcia]. Dzięki tej pośredniej zależności między terminami i dokumentami otrzymujemy dostęp do wyższego poziomu abstrakcji, jakim są pojęcia.

Na wstępie należy zmniejszyć i skondensować dane wejściowe. Pierwotna macierz [terminy X dokumenty] otrzymana w wyniku indeksowania jest najczęściej zbyt duża i zbyt rzadka, co niepotrzebnie wpływa na efektywność metody. Dodatkowo, zależnie od metody indeksowania, macierz może być w różnym stopniu zaszumiona. Dlatego stosuje się łączenie wymiarów. Polega ono na sumowaniu wektorów terminów o małych wagach.

Aby obliczyć korelację terminów należy przemnożyć skalarnie ich wektory. Więc aby móc operować na korelacjach terminów stosuje się iloczyn macierzowy dla skondensowanej macierzy samej ze sobą. W wyniku otrzymujemy macierz, wszystkich korelacji terminów (każdy z każdym).

27 tf-idf (ang. *term frequency–inverse document frequency*) - to metoda ważenia terminów indeksowych w oparciu o częstość ich występowania w dokumencie, w stosunku do liczby dokumentów, w których termin występuje

Dla takiej macierzy przeprowadza się dekompozycję SVD[28], w wyniku której otrzymujemy iloczyn trzech macierzy: UEV^T. U i V są macierzami ortonormalnymi, to znaczy, że wynikiem ich iloczynu jest macierz jednostkowa. Macierz *E* jest macierzą diagonalną – posiada niezerowe wartości tylko na przekątnej. Dzięki takiej dekompozycji możemy niezależnie operować na macierzy U, w której wierszach zawarte są związki termin - pojęcie oraz na macierzy V, której wektory odzwierciedlają związki dokument - pojęcie.

Zastosowania LSA obejmują najczęściej klasyfikację i grupowanie oraz wyszukiwanie semantyczne. Grupowanie dokumentów na poziomie pojęć osiąga się przez proste porównywanie wierszy macierzy *V*. Najczęściej do ich porównywania stosuje się podobieństwo cosinusowe. Analogicznie przez porównywanie dwóch wektorów macierzy U można grupować terminy. Wyszukiwane semantyczne polega na przeniesieniu zapytania w przestrzeń pojęć. Aby to zrobić, zgodnie z modelem wektorowym, z terminów zapytania tworzy się mini-dokument, którego wektor porównuje się z pozostałymi wektorami macierzy indeksowej.

4.1.1 Probabilistyczna LSA

Wspomniana złożoność obliczeniowa SVD wymagająca zmniejszania macierzy wejściowej była punktem wyjścia dla modyfikacji metody LSA. Wprowadzona w Hofmann 1999) metoda PLSA (ang. *Probabilistic Latent Semantic Analysis*) zakłada taki sam algorytm jak LSA, z tą różnicą, że dekompozycja macierzy nie odbywa się przez SVD. Zamiast tego używana jest dekompozycja na bazie modelu ukrytych klas, która ma solidne podstawy statystyczne i jest mniej złożona obliczeniowo. Niestety okazuje się, że metoda ta ma poważne problemy z nadmiernym dopasowaniem (ang. *overfitting*), ponieważ liczba jej parametrów rośnie liniowo ze wzrostem liczby dokumentów.

4.2 Grupowanie WWW w praktyce

Jak zaznaczono wcześniej grupowanie to najprostszy, intuicyjny sposób zarządzania dużymi zbiorami danych. Problem ten szczególnie uwidacznia się w WWW, gdzie od dłuższego czasu problemem nie jest dostęp do informacji, ale jej ogarnięcie i uporządkowanie. W kolejnych podrozdziałach opisano kilka nowych projektów, które używają grupowania WWW umożliwiającego dostęp interesujących faktów. Czasami pojęcie grupowanie nie jest używane wprost, ale samą idea funkcjonalności grupowania można odnaleźć w różnych aspektach tych projektów.

4.2.1 Aurora

Jak widać w wideo (Garrett 2008a) z bloga Adaptive Path (Garrett 2008b) pomysły na sposób prezentowania dokumentów w Aurorze – przeglądarce przyszłości - to wielokryteriowe grupowanie. Pomysł Garretta – twórcy pojęcia AJAX i głównego projektanta Aurory – z (Garrett 2008d) to grupowanie obiektów w widoku przestrzennym (ang. *spacial view*) według podobieństwa semantycznego lub wzorca użycia użytkownika. Dwa wymiary przestrzeni służą grupowaniu w ścisłym tego tego słowa znaczeniu – system prezentuje klastry (grupy) jako obiekty zbliżone do siebie w konkretnej płaszczyźnie. Klastry tworzone są automatycznie na podstawie ich semantycznej informacji. Użytkownik może również tworzyć ręcznie klastry zbliżając do siebie obiekty. Poza odległością o przynależności obiektu do wybranego klastra świadczy również jego przezroczystość: im bardziej wyświetlany obiekt jest przezroczysty, tym mniejsza jego przynależność do klastra (czy odległość od centroidu klastra).

Trzeci wymiar przestrzeni reprezentuje czas. Widok przestrzenny to wgląd wgłąb osi czasu. Czas obiektu to czas jego ostatniej interakcji z użytkownikiem. Na pierwszym planie prezentowane są

28 SVD (ang. *Singular Value Decomposition*) - rozkład według wartości osobliwych. Jest to metoda dekompozycji macierzy na iloczyn trzech specyficznych macierzy

obiekty aktualnie „otwarte”. Dodatkowym elementem usprawniającym prezentacje dokumentów w przestrzeni jest grawitacja. Obiekty podobne semantycznie przyciągają się wzajemnie tworząc klastry. Nowe obiekty umieszczane są w przestrzeni zgodnie z wpływem grawitacji. Jeśli, na przykład, nowy obiekt powiązany jest z dwoma klastrami – zostanie umieszczony pomiędzy nimi. Dodatkowo jeśli jeden z tych klastrów jest znacząco większy – nowy obiekt znajdzie się bliżej niego. Przesuwanie obiektów wzdłuż osi czasu zależne jest od czasu ostatniej interakcji użytkownika z nimi. Dokumenty zamknięte powoli oddalają się od widoku użytkownika.

Co ciekawe w Aurorze obiekty dzielą się na 3 klasy: ludzie, miejsca i rzeczy. Ludzie to głównie kontakty użytkownika wraz z semantyczną historią komunikacji i współpracy. Obiekty ludzi mogą też zawierać informacje udostępniane przez te osoby (uwzględniając ograniczenie i uprawnienia dostępu) lub dane dołączone przez użytkownika. Miejsca to zasoby sieciowe określone przez URI – nie koniecznie strony WWW, ale również zapisane stany aplikacji WWW. Obiekty trzeciej klasy - rzeczy – zawierają obiekty specjalnych klas, jak: obiekty danych, obiekty logiki, obiekty prezentacji czy tokeny. Obiekty danych to proste repozytoria ustrukturyzowanych danych. Na tych obiektach działają i manipulują nimi obiekty logiki. Interfejs oraz wizualizacje tych manipulacji dają użytkownikowi obiekty prezentacji. Z kolei dostęp do danych i usług zapewniają tokeny - obiekty przechowujące dane potrzebne do autentykacji i autoryzacji.

Rysunek 4.1. Interfejs użytkownika projektu Aurora

źródło: (Garrett 2008c)

Domyślny widok przestrzenny (rys. 4.1) dotyczy tylko obiektów, z którymi użytkownik miał już styczność. Jeśli wyszukiwane są nowe obiekty, są one umieszczane w przestrzeni razem z istniejącymi, przy czym obiekty nierelewantne stają się przezroczyste, niemal niewidoczne. W projekcie nie ma rankingu relewancji, chociaż nazwy klastrów znaczą więcej niż słowa w tekście. Aby pozbyć się w wynikach wyszukiwania efektu „pustynnego autobusu” oś czasu jest kompresowana, tak, aby zmieścić tylko wyszukane obiekty.

W projekcie Aurora uwidocznione są niektóre aktualne trendy w dziedzinie wyszukiwania informacji w WWW. Przede wszystkim sama treść dokumentów to za mało, żeby dobrze wyszukiwać. Wyszukiwarka musi uwzględniać informacje semantyczne, a najlepiej umieć na ich podstawie wnioskować, żeby zwrócić użytkownikowi naprawdę wartościowe wyniki. Prezentacja

wyników wyszukiwania dokumentów musi być intuicyjna, efektywna i ergonomiczna. Tekstowa lista kilku dziesięciu pierwszych dokumentów i link do kolejnych porcji listy nie spełnia tych wymogów. Prezentowany pomysł widoku przestrzennego z podglądem zawartości obiektów i prostą – niezależną od platformy – nawigacją między nimi to sposób wizualizacji odpowiadający aktualnym trendom. Bardzo istotną ideą przyświecającą rozwojowi WWW, a przez to mającą wpływ na wyszukiwanie w niejm jest oddzielenie treści od formy. Dzięki temu poza ułatwieniem agentom Semantic Web przetwarzania wiedzy, użytkownik dostaje możliwość prezentacji i manipulacji danymi w praktycznie każdej potrzebnej mu postaci. Z kolei sam banał istotności wyszukiwania w dzisiejszej interakcji z WWW czy komputerem w ogóle podkreślają pytania: „Czy Aurora jest przeglądarką, która przejmuje część funkcjonalności komputera (ang. *desktop*)? Czy może Aurora jest środowiskiem komputerowym (desktopowym), które integruje dostęp do zasobów WWW?". Autorzy projektu na oba pytania odpowiadają „tak".

4.2.2 Freebase Parallax

Koncept Adaptive Path skupia się głównie na interfejsie i sposobie wizualizacji grupowania dokumentów WWW. Nowe spojrzenie na manipulację danymi daje screencast możliwości systemu Parallax (Huynh 2008b). Parallax jest nowatorskim interfejsem przeglądania WWW zaprojektowanym dla Freebase. Z kolei Freebase jest opartą na grafie bazą danych zawartych w WWW. Freebase strukturyzuje dane pobierane m.in. z Wikipedii oraz dane wprowadzane przez użytkowników. Dzięki takiemu ustrukturyzowaniu WWW możliwe jest tworzenie zapytań w specjalnym języku MQL, w sposób analogiczny do zapytań SQL dla tradycyjnych baz danych. Jednak, ponieważ interfejs języka zapytań nie jest specjalnie intuicyjny i wygodny dla przeciętnego użytkownika powstają projekty takie jak Parallax, które mają umożliwić wygodny sposób przeglądania WWW, którą w tym momencie można już uznać za Semantic Web.

Główny pomysł prezentacji danych, czy raczej już wiedzy, w Parallax to użycie faset (ang. *facets*). Jak tłumaczy autor Parallax – David Huynh – w wywiadzie z Jon'em Udell'em (Huynh 2008a) używanie faset to sposób na pozwolenie użytkownikowi na ułożenie zapytania, bez potrzeby używania kodu SQL, czy innego języka. Dzięki temu użytkownik otrzymuje możliwości dostępu do informacji oferowanej przez języki zapytań, ale nie wymaga to od niego większego wysiłku niż tradycyjna nawigacja.

Nawigacja po wynikach to głównie filtrowanie danych na podstawie semantycznych atrybutów wspólnych dla obiektów pobranych z Freebase. Jednak poza filtrowaniem, możliwe jest przejście do innych wyników, powiązanych z bieżącymi. Na przykład, mając kolekcję prezydentów USA, możemy za pomocą atrybutu powiązania *children* przejść do kolekcji wszystkich dzieci tych prezydentów. W tradycyjnym podejściu, aby uzyskać taki wynik należałoby po kolei dla każdego prezydenta sprawdzać czy ma dzieci i dodawać je do nowej kolekcji. W Parallax stworzenie takiej kolekcji wiąże się tylko z jednym kliknięciem. Dzieje się tak dlatego, że system na bieżąco wylicza wszystkie możliwe ścieżki powiązań dla każdej pary obiektów. Dzięki temu uzyskujemy nowy rodzaj powiązania. Tradycyjnie linki reprezentowały powiązanie jeden do jednego, ewentualnie jeden do wielu. Tutaj mamy do czynienia z nawigacją wzdłuż powiązań wiele do wielu.

Używanie faset i powiązań wiele do wielu można uznać za jeden ze sposobów grupowania wyników. Zakładając, że wartości atrybutów wiążących zostały poprawnie wydobyte z Freebase takie grupowanie wydaje się o wiele bardziej przydatne niż grupowanie oparte na heurystykach. Poza tym, w zależności od semantyki atrybutów, Parallax umożliwia wizualizację kolekcji na mapie, osi czasu lub diagramach jedno- i dwuwymiarowych.

4.2.3 Powerset

Alternatywną dla DBpedii aplikacją dodającą możliwości Semantic Web do artykułów Wikipedii jest Powerset. Powerset przetwarzając artykuły wyławia z nich fakty, czyli zapisane w język naturalnym trójki RDF. Dzięki temu możliwe jest eksplorowanie faktów zawartych w artykułach

oraz zadawanie pytań w języku naturalnym. Elementy faktu (podmiot, orzeczenie, dopełnienie nie zawsze występują w zdaniu w odpowiedniej kolejności oraz mogą być rozdzielone dodatkowymi informacjami, dlatego Powerset umożliwia równolegle do eksplorowania faktów podświetlenie wyrazów trójki, oraz ich kontekstu w tekście źródłowym artykułu. Dostęp do tekstu odbywa się więc przez fakty. Z kolei same fakty mogą być grupowane nie tylko według artykułów i paragrafów, w których występują, ale również według obiektów, których dotyczą. Obiekty pomimo jednakowej nazwy mogą być rozróżnione np. Henryk VIII – osoba i Henryk VIII – opera. Dzięki temu otrzymujemy mechanizm łączący artykuły nie tylko za pomocą linków, ale również za pomocą semantyki faktów i obiektów. Barney Pell, CTO projektu Powerset, w wywiadzie dla The Semantic Web Gang (Pell 2008) wyjaśnia, że zapytania w języku naturalnym i podświetlanie słów kluczowych zapytania w tekście wynikowym to żadna nowość, ale Powerset idzie dalej. Dzięki analizie faktów w wynikach zapytania podświetlana jest również odpowiedź, czyli obiekt, którego dotyczyło zapytanie. Na przykład w odpowiedzi na pytanie „Kogo pokonał Hulk Hogan?" podświetlone zostaną nie tylko podmiot („Hulk Hogan") i orzeczenie („pokonał"), ale również każde dopełnienie tworzące z tymi elementami trójkę faktu, czyli dokładną odpowiedź na pytanie. Dodatkowo dzięki semantyce faktów w odpowiedzi nie otrzymamy informacji o tych, którzy pokonali Hulka (czego pytanie nie dotyczy), co jest normą w wyszukiwarkach opartych jedynie na indeksowaniu słów kluczowych.

Poza wyszukiwaniem w faktach wydobytych z artykułów Wikipedii, Powerset korzysta również z Freebase, aby odpowiedzieć na bardziej ogóle pytania typu „malarze impresjonizmu". Dzięki temu wynikiem zapytania mogą być nie teksty, w których się pojawia odpowiedź, ale syntetyczna lista, która jest odpowiedzią na pytanie (np. lista osób pokonanych przez Hulka). Z drugiej strony Powerset ułatwia również znalezienie fragmentu tekstu zawierającego jedną z odpowiedzi, co w przypadku długich tekstów artykułów zwykle bywa uciążliwe.

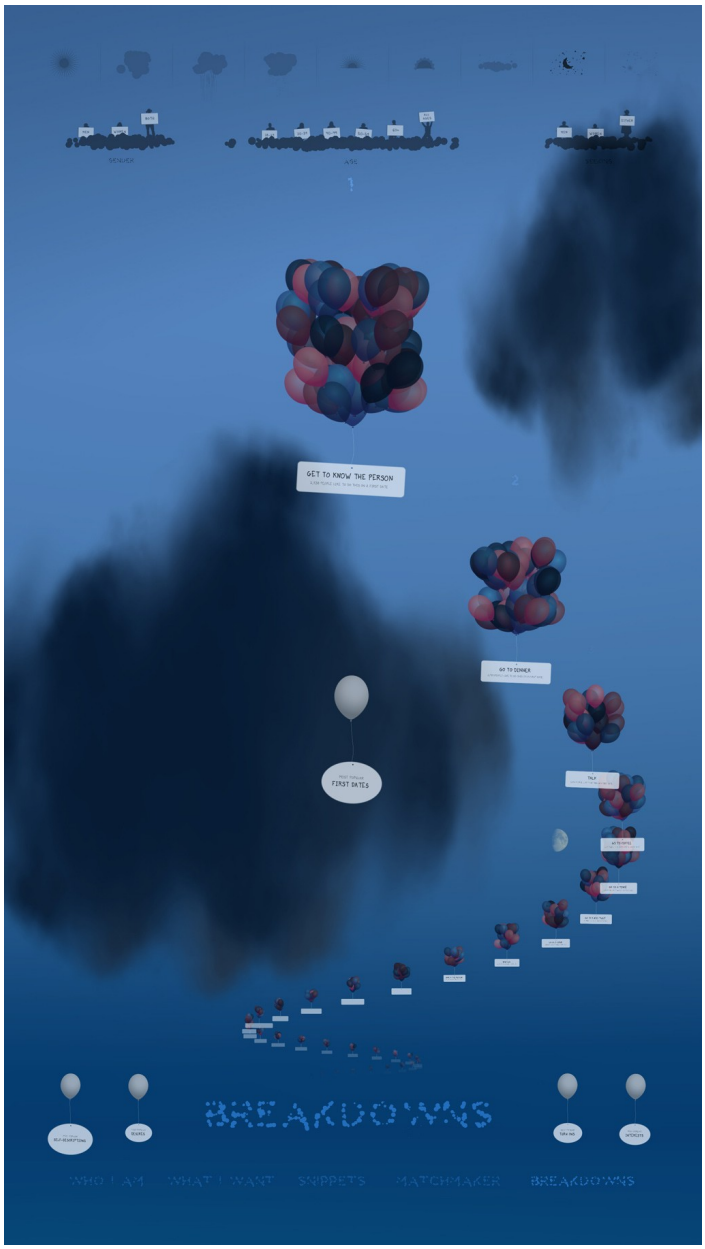
4.2.4 Grupowanie w sztuce

Na swojej wystawie w Muzeum Sztuki Współczesnej w Nowym Jorku, zatytułowanej „Design and the Elastic Mind" Jonathan Harris i Sep Kamvar przedstawili interaktywną instalację „I Want You To Want Me". Instalacja wystawiona w Walentynki 2008 ma na celu pokazanie poszukiwania miłości i siebie w świecie on-line'owych randek.

Baza danych eksponatu to aktualizowana co kilka godzin kolekcja profili osób pobieranych z on-line'owych serwisów randkowych. Każdy profil przedstawiony jest jako balonik. Niebieskie baloniki to mężczyźni, różowe – kobiety. Jaśniejsze baloniki oznaczają ludzi młodszych, ciemniejsze – starszych. W każdym baloniku jest cień sylwetki w różnych pozycjach. Profile grupowane są ze względu na różne parametry. Na przykład ze względu na zawartości zdań w profilach zaczynających się od „Jestem..." albo „Szukam...". Inne parametry grupowania to linie początkowe profilu (ang. *openers*), końcowe (ang. *closers*) i przewodnie (ang. *taglines*). Grupowanie wizualizowane jest na 56" wyświetlaczu dotykowym, służącym jako interfejs. Klikając na baloniki można podejrzeć pełną informację o profilach. Swatka (ang. *matchmaker*) to algorytm parujący profile na podstawie opisów ich właścicieli: „kim są" i „czego pragną".

W trybie analizy (ang. *breakdowns*) jak widać na rysunku 4.2 profile pogrupowane według wybranego kryterium. Rysunek 4.2. przedstawia grupowanie według kryterium „to co ludzi kręci" (ang. *turn ons*). Liczność grup baloników ułożonych na kształt ogona węża świadczy o ich popularności. Jak widać największa popularnością cieszy się inteligencja. Inne kryteria to, przedstawione jako białe baloniki na dole, pierwsze randki, samo-opisy, pragnienia i zainteresowania. Dodatkowo, jak przy każdej analizie klastrów, przydatną opcją jest filtrowanie. Tutaj filtry demograficzne zrealizowano w następujący sposób: Filtry zgrupowane są w chmury na górze ekranu. Trzy chmury oznaczają: płeć właścicieli profili, przedział wiekowy i płeć poszukiwanych partnerów. Na chmurach siedzą sylwetki trzymające tablice z konkretnymi opcjami filtra. Sylwetki stojące oznaczają filtr włączony. Na przykład podnosząc na kolejnych chmurach

osoby z tablicami: „kobieta”, „30-39”, „kobieta”, analiza dotyczyć będzie lesbijek w wieku 30-39 lat.



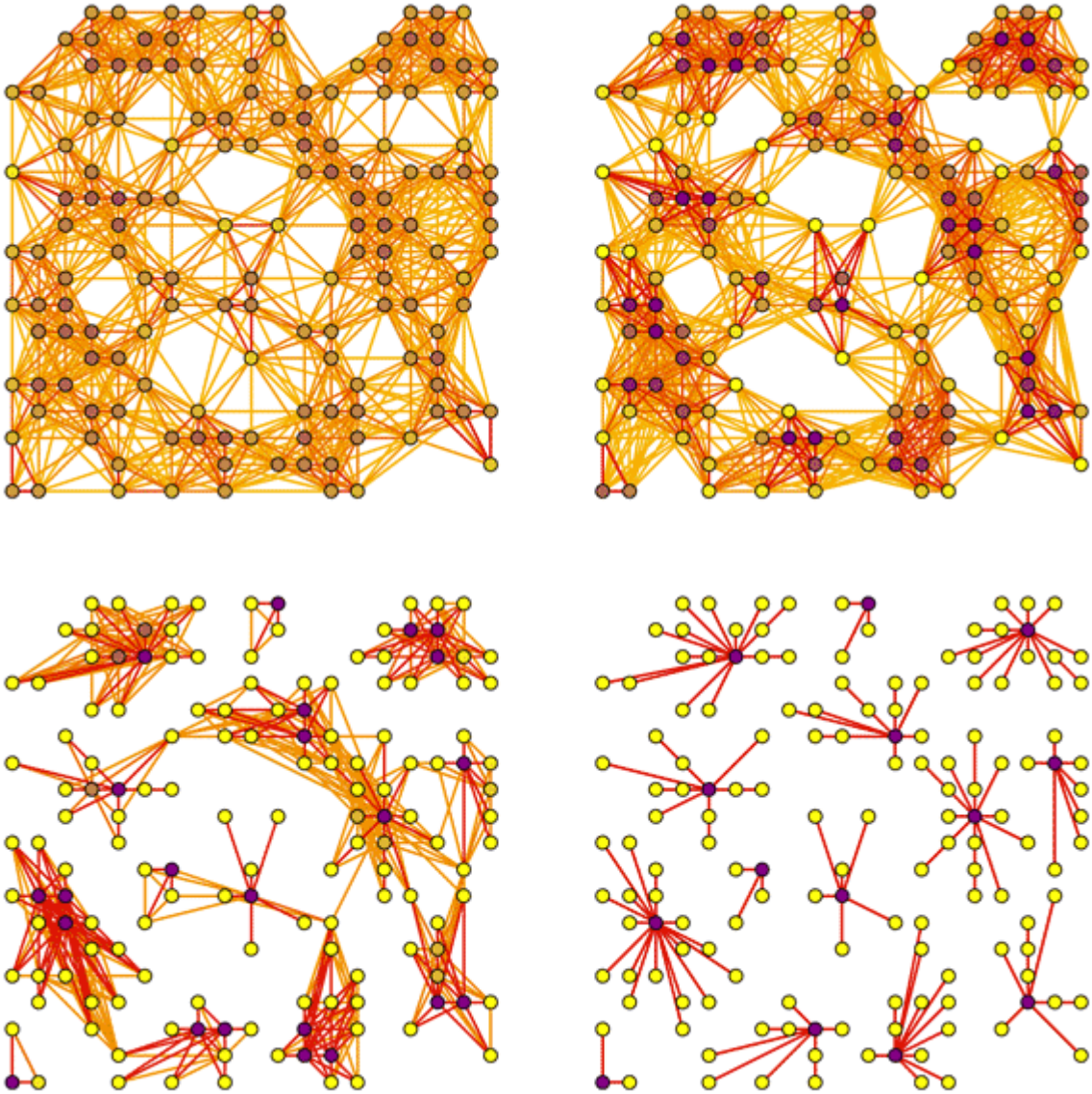
Rysunek 4.2. Opcja „Turn ons” w trybie „Breakdowns” w eksponacie "I Want You To Want Me" z Muzeum Sztuki Współczesnej

źródło: (Harris i Kamvar 2009)

4.3 Grupowanie grafu

Typowe grupowanie najczęściej odbywa się dla zbiorów. Zbiór w odróżnieniu od grafu nie posiada żadnej informacji strukturalnej czy nawet informacji na temat kolejności – wszystkie elementu zbioru są „równe”. Grupowanie odbywa się na podstawie pewnej funkcji odległości (bądź jej odwrotności, czyli podobieństwa), która jest zdefiniowana dla dowolnych dwóch elementów zbioru. Taką sytuację można odnieść do grafu pełnego, gdzie dla każdej pary węzłów określona jest waga krawędzi z nimi incydentnych. W zasadzie gdyby udało się określi jednolitą funkcję, która

opisywałaby wagi krawędzi dla każdej pary węzłów, wtedy graf można by sprowadzić do zbioru i grupować węzły typowymi algorytmami grupowania.



Rysunek 4.3. Ilustracja działania algorytmu MCL

źródło: (Dongen 2009)

Jednak graf ważony jest szczególnym przypadkiem zbioru obiektów (węzłów), w którym zamiast funkcji odległości mamy określoną strukturę i wagi krawędzi. Ta specyfika powoduje, że zastosowanie dla grafu ważonego typowych algorytmów grupowania jest co najmniej nieefektywne, jeśli w ogóle możliwe. Problem grupowania grafów był głównym problemem rozprawy doktorskiej (Dongen 2000). W wyniku pracy opracowany został algorytm grupowania grafu MCL (ang. *Markov Cluster*). Grupowanie grafu w MCL oparte jest na symulacji stochastycznego przepływu (łańcuch Markova) opisanego przez macierze Markova. Sam algorytm polega na przekształcaniu tych macierzy za pomocą dwóch operatorów *expand* i *inflate*.

Jak widać na rysunku 4.3 naprzemienne stosowanie tych operatorów, nazwane procesem MCL, na macierzy reprezentującej graf prowadzi do tego, że przepływ rozdzielany jest w osobne regiony, które interpretowane są jako grupy, czyli podgrafu pierwotnego grafu. Operacja *expand* to po prostu

podnoszenie do macierzy kwadratu. Operacja *inflate* ma na celu przeskalowanie macierzy stochastycznej, aby po potęgowaniu pozostała stochastyczną. Czyli macierz jest normalizowana i odrzucane są wartości poniżej założonego progu istotności. Proces MCL najczęściej kontynuowany jest, podobnie jak w innych algorytmach grupujących, do momentu osiągnięcia zbieżności lub spełnienia innego kryterium, np.: docelowa ziarnistość (liczba grup), liczba iteracji operatorów, itp.

W (Brandes, Gaertler, i Wagner 2003) został przedstawiony alternatywny dla MCL algorytm grupujący dla grafów: *Geometric MST Clustering* (GMC). Algorytm GMC łączy partycjonowanie spektralne z grupowaniem geometrycznym. W pracy porównywany jest on z MCL i z *Iterative Conductance Cutting* (ICC) wprowadzonym w Kannan, Vempala, i Vetta 2004). ICC to algorytm grupowania deglomeracyjnego używający cięć minimalnego przewodnictwa.

Grupowanie grafów polega na wyznaczeniu podgrafów. Najczęściej podgrafu interpretowane są jako grupy węzłów, ponieważ informacja strukturalna (krawędzie) najczęściej potrzebna jest tylko w celu zastosowania algorytmu grupującego. Można powiedzieć, że węzły w grupie łączy pewien związek – przynależność do danego podgrafu w wyniku grupowania. Przyjmując taką interpretację można wyniki grupowania przedstawić w postaci hipergrafu. Węzły w takim hipergrafie to węzły pierwotnego grafu, dla którego przeprowadzono grupowanie. Krawędzie tego hipergrafu to grupy wynikowe algorytmu grupującego. W związku z tym, robiąc formalną analogię można powiedzieć, że tak jak algorytm grupowania zbioru dotyczy przekształcenia jednorodnego zbioru w zbiór podzbiorów, tak algorytm grupowania grafu najczęściej ma na celu przekształcenie grafu w hipergraf.

5 Metody analizy spójności i zgodności

Spójność i zgodność są aspektami analizy kolekcji dokumentów WWW. Jednak, aby oszacować miary spójności i zgodności kolekcji, nie wystarczą same dokumenty. Niezbędna jest również informacja o związkach, jakie występują pomiędzy tymi dokumentami oraz między dokumentami i innymi obiektami infrastruktury WWW. W niniejszej pracy wśród pozostałych obiektów wchodzących w związki z dokumentami wybrano użytkowników i pojęcia. Użytkownicy to najczęściej autorzy dokumentów, ale nie tylko. Związek pomiędzy użytkownikiem i dokumentem może zostać stworzony na przykład: gdy dokument (w zasadzie autor) powoła się (zacytuje) na użytkownika lub z drugiej strony: gdy użytkownik oceni dokument czy doda go do ulubionych. Podobnie ma się sprawa ze związkami dokument – pojęcie. Dokument może mieć przypisany tag, czyli słowo kluczowe, według którego można grupować czy filtrować dokumenty. Z drugiej strony słowo kluczowe może zostać wydobyte z treści dokumentu poprzez analizę leksykalną czy ważenie terminów. Oczywiście równie istotne są związki między obiektami jednego typu, czyli np. autor – autor.

Aby modelować kolekcję dokumentów na potrzeby konkretnej metody należy w tym miejscu uściślić pojęcia „dokument" i „kolekcja". Z kilkunastu definicji dokumentu przytoczonych w podrozdziale 2.1 Dokument WWW s. 22 najbardziej odpowiednią dla opracowanej metody wydaje się definicja 2.1.7. Zakłada ona rekurencyjność zgodną z definicjami 2.1.11 i 2.1.12 oraz wskazuje źródło pochodzenia dokumentu jako odpowiedź na pytanie. Z kolei żadna z wcześniej przytoczonych definicji kolekcji nie pasuje w pełni do obiektu modelowanego w metodzie. Dlatego na potrzeby metody została wprowadzona autorska definicja 5.1

Definicja 5.1. Kolekcja dokumentów WWW

„Kolekcją dokumentów WWW nazwiemy zbiór (może być uporządkowany) dokumentów umieszczonych w WWW i występujący w pewnym związku. Związek ten może być tak jawny i bezpośredni jak hiperłącze z jednego dokumentu do drugiego. Może też być tak ukryty i pośredni jak wzajemny związek autorów obu dokumentów czy jednakowa klasyfikacja obu dokumentów przez wyszukiwarkę jako relewantnych dla określonego zapytania."

Najbliższe definicji 5.1 są definicje 2.3.2 i 2.3.3, jednak żadna z nich nie mówi wprost o związkack, które mają wpływ na istnienie kolekcji oraz nie kładzie nacisku na umiejscowienie dokumentów w WWW.

Jeśli chodzi o uściślenie terminu „link", to niestety jego popularność i elastyczność uniemożliwiają przyjęcie jednej definicji. Stąd podobnie jak w przypadku wcześniej przytoczonych definicji linka, konkretne znaczenie determinowane będzie przez bezpośredni kontekst użycia tego terminu.

Przyjmując definicję kolekcji i dokumentu WWW można zdefiniować czym jest spójność i zgodność.

Definicja 5.2. Spójność kolekcji dokumentów

„Spójnością kolekcją dokumentów WWW nazwiemy własność kolekcji wynikającą ze związków między dokumentami i ich autorami."

Definicja 5.2 wskazuje, że spójność zależy wprost od siły związku między obiektami WWW. Intuicyjnie: im silniejsze wzajemne związki zachodzą pomiędzy dokumentami kolekcji, pomiędzy autorami tych dokumentów oraz pomiędzy autorami i dokumentami, tym większa jest spójność kolekcji.

Definicja 5.3. Zgodność kolekcji dokumentów

„Zgodnością kolekcją dokumentów WWW nazwiemy własność kolekcji wynikającą ze związków między dokumentami i pojęciami użytymi do opisania tych dokumentów.”

Definicja 5.3, dotycząca zgodności, jest analogiczna do definicji spójności 5.2. Tak jak spójność dotyczy autorów dokumentów, tak zgodność skupia się na sile związku dokumentów z pojęciami, czyli słowami kluczowymi, tagami. Podobnie też zgodność będziemy nazywać tym większą, im silniejsze związki wystąpią pomiędzy obiektami dokumentów i pojęć.

Aby zmierzyć spójność i zgodność kolekcji należy przyjąć konkretne miary. Z kolei na miary będą miały wpływ przyjęte wagi reprezentujące siłę związków między obiektami.

5.1 Związki w WWW

Mówiąc o związkach w WWW na myśl przychodzą przede wszystkim powiązania dokumentów WWW hiperłączami, czyli hipertekstowe odwołania z jednego dokumentu do innego. Takie powiązanie jest **skierowane**, to znaczy: możemy rozróżnić dokument odwołujący się i ten, do którego odwołanie następuje. Pomimo że może występować link zwrotny, linkujący z powrotem do dokumentu odwołującego się z dokumentu, do którego skierowało nas pierwsze odwołanie, to sam fakt podlinkowania innego dokumentu w hipertekście jest operacją skierowaną. Podobnie ma się sprawa z użytkownikami WWW, którzy deklarują wzajemne związki. Jeśli, na przykład, jeden z użytkowników umieści w swoim blogroll'u link do bloga innego użytkownika z informacją XFN „*friend*”, to taka deklaracja przyjaźni jest związkiem skierowanym. Ponieważ autor wskazywanego bloga może umieścić zwrotny link z informacją XFN „*colleague*”, oznaczający słabszy związek niż przyjaźń, więc uogólnienie tych dwóch związków w jednen może być niejednoznaczne. I dalej przez analogię: mając dwa pojęcia możemy zbadać związek między nimi (zakładając, że to rzeczowniki) w ontologii typu „jest częścią”/”jest uogólnieniem”, dostępnej np. w WordNet'cie. Ponieważ taka ontologia jest najczęściej drzewem, kierunek związku jest istotny. Kierunek związku można jawnie odczytać w przypadku związku między dwoma obiektami tego samego typu: np. dokument – dokument. W przypadku badania związku między obiektami różnych typów kierunek związków może zależeć od interpretacji semantyki tego związku. Na przykład: kierunek związku autor – dokument można interpretować jako „dokument ma autora”, czyli kierunek związku: od dokumentu do autora. Można też odwrotnie przyjąć, że „dany obiekt jest autorem dokumentu”. Kierunek tak odczytanej semantyki związku będzie przeciwny.

Równie ważnym, a może nawet ważniejszym, atrybutem związku między obiektami WWW jest waga związku. Na wagę związku mają wpływ poziom i typ związku. Dzięki sieci semantycznej coraz więcej typów związków między obiektami dokument, autor i pojęcie można odczytać z WWW w sposób bezpośredni, gdzie do tej pory metodą analizy związków była ekstrakcja informacji czy nawet drążenie danych (ang. *data mining*). Na potrzeby niniejszej pracy wyróżniono następujące poziomy i typy związków między obiektami:

1. dokument – dokument:
   - fakt istnienia hiperłącza między dokumentami; typ związku to semantyka zawarta w atrybucie `rel` (patrz 2.2.3 Typy linków s. 36), np.:
     - *ogólny-szczegółowy*, *rozdział – spis treści*, *adnotacja, tekst źródłowy*
     - *errata, aktualizacja, szkic roboczy, kolejna wersja*, itp.
   - bezpośrednie cytowanie treści innego dokumentu
   - umieszczenie dokumentu w liście referencji/bibliografii
   - podobieństwo treści mierzone metodami ważenia terminów, typy to przyjęte miary np.:
     - miara cosinusowa, nakładania, Dice'a lub Jaccard'a dla wektorów terminów ważonych
     - liczba wspólnych terminów w tytułach/abstraktach 2 dokumentów
     - podobieństwo wielkości dokumentów (liczba słów, obrazków, tabel, itp.)

- podobieństwo pochodzenia dokumentów, np.
  - jeden/powiązany wydawca, blog, serwis (URL),
  - ta sama/powiązana kolekcja biblioteki cyfrowej, kanał RSS,
  - automatycznie generowane kolekcje wynikające np. z popularności, relewancji,
  - odległość bezwzględnych lub względnych (dzień tygodnia, pora dnia) czasów stworzenia/aktualizacji,
  - dostępność formatów, języków, licencji dla treści dokumentów, itp.

2. autor – autor:
   - deklarowanie faktu znajomości on-line, typy:
     - za pomocą własności *foaf:knows*,
     - za pomocą kombinacji typów *XFN*, np.: *friend met, co-worker acquaintance met*,
     - dodając do listy znajomych w serwisach społecznościowych,
   - współuczestnictwo w zdarzeniach; typami są role osób w zdarzeniach, np.:
     - organizator – uczestnik sesji konferencyjnej ,
     - manager - członek grupy projektowej,
     - słuchacze wykładu, itp.
   - wspólne cechy osób wynikające np. z:
     - podobieństwa preferencji (profili z serwisów społecznościowych), np. gusta muzyczne, zakładki WWW, listy życzeń, itp.
     - aspektów demograficznych, np.:lokalizacja, język, itp.
3. pojęcie – pojęcie:
   - podobieństwo leksykalne,
   - odległość w grafie ontologii stworzonej np. na podstawie WordNet'owego związku hiponim – hiperonim (czyli ogólny – szczegółowy),
4. dokument – autor:
   - (współ-)autorstwo dokumentu,
     - autorstwo bezpośrednie
     - redakcja pracy zbiorowej, gdzie rozdziały tworzą różni autorzy, np. wydawnictwa konferencyjne, blogi moderowane
   - powołanie się w dokumencie na nazwisko autora,
   - komentowanie/recenzowanie dokumentu (jeśli nie traktować komentarza/recenzji jako osobnego dokumentu),
5. dokument – pojęcie:
   - słowo kluczowe (tag) przypisane do dokumentu przez autora lub eksperta w dziedzinie,
   - termin ważony uzyskany np metodą tf-idf

W praktyce trudno znaleźć dostęp do danych, w których dostępne były by informacje o wszystkich poziomach związków. Dlatego praktyczny algorytm wykorzystujący związki powinien być ukierunkowany i dostosowany do dostępnych w badanym źródle danych informacji o związkach. Stąd, jak zaznaczono we wprowadzeniu, rozpatrzone zostaną dwa podejścia do rozwiązania: ogólne i ukierunkowane na dostępne dane.

5.2 Model świata WWW – graf DAC

Dotychczasowe prace dotyczące analizy WWW skupiały się głównie na dokumentach i związkach między nimi. Związki to przede wszystkim wzajemne cytowanie za pomocą hiperłączy. Najbardziej znane metody: HITS i PageRank opierają się na takiej właśnie analizie. W niniejszej pracy zaproponowany został model WWW rozszerzony o dodatkowe obiekty WWW, o których – dzięki idei Sieci Semantycznej – możemy uzyskać informacje. W celu analizy spójności i zgodności kolekcji dokumentów, oprócz modelowania samych dokumentów WWW, można zamodelować autorów i pojęcia. Taki wybór determinowany jest przez cel analizy. Wprowadzenie autorów do

modelu umożliwi pełniejszą analizę spójności kolekcji dokumentów. Natomiast wprowadzenie – jako osobnych obiektów – pojęć poszerzy możliwości analizy zgodności kolekcji.

Jak wspomniano, rozwój Sieci Semantycznej daje możliwości dostępu do metadanych, które umożliwiają pełniejszą analizę WWW. Do niedawna znalezienie informacji o autorze dokumentu/strony WWW było często bardzo trudne. Pomimo że standard HTML przewidywał odpowiednie atrybuty w znacznikach `meta` dokumentów, to jednak w procesie publikowania treści w WWW, dodawanie metadanych najczęściej było pomijane. Wynikało to najprawdopodobniej z faktu, że autor treści rzadko był osobą umieszczającą dokument HTML na serwerze. W czasach statycznych stron WWW, gdy webmasterzy ręcznie edytowali kod HTML tagi `meta` za sprawą używania szablonu pozostawały jednakowe dla całego serwisu. Niezależnie od tego kto był rzeczywistym autorem tekstu lub zdjęcia i jeśli nawet jego nazwisko pojawiło się w treści dokumentu czytelnej dla człowieka, to maszyna przetwarzająca taki dokument, znajdywała w metadanych informację, że autorem treści jest autor całego serwisu. Oczywiście można było próbować wydobyć informacje o autorze treści z samej treści, ale takie heurystyki NLP, czyli przetwarzania języka naturalnego (ang. *Natural Language Processing*) w ogólności nie mogły dawać zadowalających wyników.

Podobnie sprawa miała się z innymi metadanymi, które powinny być dostępne dla automatycznej analizy. Choć w przypadku wydobywania słów kluczowych znanymi wcześniej metodami ważenia terminów wyniki były obiecujące, to jednak nie uwzględniały specyfiki WWW i nie mogły się równać ze słowem kluczowym przypisanym ręcznie przez autora lub eksperta. Na zmianę sytuacji miały wpływ dwa zdarzenia: wprowadzenie dynamicznych stron WWW i popularyzacja systemów zarządzania treścią - CMS (ang. *Content Management System*). Dzięki wykorzystaniu dynamicznych stron WWW kod HTML nie musi być edytowany ręcznie, tylko generowany najczęściej przez skrypt po stronie serwera. Dzięki temu, mając informację o autorze, system może wstawić tą informację w odpowiednie znaczniki generowanego HTML'a, dając możliwość jej automatycznego przetwarzania. Z drugiej strony, dzięki spopularyzowaniu idei CMS w blogach i serwisach typu Web 2.0, gdzie każdy użytkownik ma własne konto, informacja o autorze tworzącym treść jest zawsze obecna – wystarczy ją tylko odpowiednio wyeksponować. Podobnie można wyeksponować i umożliwić automatyczny dostęp do tagów, etykiet czy po prostu słów kluczowych przypisywanych w takich systemach przez autorów danej treści.

Mając dostęp do tego typu metadanych można w prosty sposób wykorzystać je w analizie WWW. Do tej pory robiono to niezależnie dla autorów w Kopel i Kazienko 2007) oraz dla pojęć w Kopel i Daniłowicz 2004a) i (Daniłowicz i Kopel 2003). Na potrzeby niniejszej pracy model WWW składa się z trzech typów obiektów:

1. **dokument** – tradycyjnie modelowany obiekt WWW,
2. **autor** – który w WWW nie tylko tworzy dokumenty, ale jako **użytkownik** wchodzi w związek z innymi dokumentami, np. oceniając je, czy dodając do ulubionych; w opisywanej analizie terminy **autor** i **użytkownik** będą używane zamiennie,
3. **pojęcie** – czyli uogólnienie **słowa kluczowego**, **tagu** przypisanego jako reprezentanta treści.

Umożliwienie łatwego dostępu do metadanych to jeden z wymogów Sieci Semantycznej. Drugi, o wiele trudniejszy, to udostępnienie ontologii, która umożliwi automatyczne przetwarzanie i wnioskowanie na podstawie dostępnych metadanych. Ten etap rozwoju Sieci Semantycznej ciągle nie został osiągnięty, choć część takiej funkcjonalności jest już dziś dostępna. Na przykład dzięki użyciu ontologii FOAF i XFN możliwe jest półautomatyczne wnioskowanie na temat typu związku między użytkownikami WWW. Dzięki ontologii zbudowanej na podstawie WordNet'u możliwe jest obliczenie semantycznej odległości między pojęciami. Tego typu informacje pozwalają wyznaczać związki między modelowanymi obiektami WWW. Aby metody analizy spójności i zgodności mogły objąć szerszy aspekt świata WWW, niż tradycyjnie dokumenty i związki między nimi, przyjęto jako model graf DAC.

Graf DAC to rozszerzenie tradycyjnego **grafu ważonego**, zgodnego z definicją 2.4.5, o różne rodzaje węzłów. W grafie DAC, jak wskazuje nazwa, mogą występować wymienione wyżej **trzy typy węzłów**: **dokument** (ang. *document*), **autor** (ang. *author*) i **pojęcie** (ang. *concept*). Istnieniem i obciążaniem krawędzi pomiędzy tymi węzłami modelowane jest **pięć rodzajów związków** opisanych w podrozdziale 5.1 Związki w WWW s. 71. Choć, jak wykazano wcześniej, część związków może być skierowana, to jednak na potrzeby analizy spójności i zgodności graf DAC jest grafem **nieskierowanym**. Przykładowy graf DAC widać w dolnej części rysunku 5.1. W związku z tym należy zauważyć, że graf DAC nie należy do popularnej klasy grafów DAG, czyli skierowanych grafów acyklicznych (ang. *directed acyclic graph*).

Definicja 5.2.1. Graf DAC

Grafem DAC nazywamy dwójkę (V, E), w której $V = \{d_1,...,d_i, a_1,...,a_j, c_1,...,c_k\}$ jest zbiorem węzłów trzech typów D, A i C, a $E = \{(v_r, v_s)\}$ jest zbiorem krawędzi incydentnych z węzłami v_r i v_s. Graf DAC jest nieskierowany więc $(v_r, v_s) = (v_s, v_r)$. Na zbiór E nałożone jest dodatkowe ograniczenie: nie może on zawierać krawędzi (a_r, c_t).

Definicja 5.2.2. Graf DA i graf DC

Grafem DA będziemy nazywać podgraf grafu DAC, powstały poprzez usunięcie z DAC wszystkich węzłów typu C i wszystkich krawędzi z nimi incydentnych. Analogicznie, graf DC to graf powstały przez usunięcie z DAC węzłów A i krawędzi z nimi incydentnych.

Definicja 5.2.1 formalnie określa graf, który w pracy używany jest do modelowania obiektów: dokument, autor, pojęcie i związków pomiędzy nimi. Ograniczenie zbioru krawędzi wynika z faktu, że związki między autorami i pojęciami nie są modelowane bezpośrednio. Ponieważ analiza spójności i zgodności dotyczy przeważnie podgrafów DAC, dlatego w definicji 5.2.2 najczęściej używane podgrafy otrzymały własne nazwy. Ze względu na fakt, że DAC jest szczególnym przypadkiem nieskierowanego grafu nieważonego, można zauważyć w nim kilka szczególnych prawidłowości. Te prawidłowości, dotyczące pełności grafu DAC i maksymalnej liczby krawędzi, nie są prawdziwe w ogólności, dla grafów o tylko jednym typie węzłów, dlatego zostały tu przytoczone. Z drugiej strony wszystkie prawidłowości ogólne dotyczące nieskierowanych grafów nieważonych dotyczą również grafu DAC.

Przy założeniu, że DAC zawiera przynajmniej, po jednym węźle typu A i C, z ograniczenia w definicji 5.2.1 wynika wprost, że taki graf nigdy nie będzie grafem pełnym, w którym każde 2 węzły muszą być połączone krawędzią. Skoro DAC nie jest grafem pełnym, to można spróbować określić maksymalną liczbę krawędzi w takim grafie.

Załóżmy, że d, a i c oznaczają odpowiednio liczności zbiorów węzłów typu D, A, i C. Gdyby DAC był grafem pełnym maksymalna liczba krawędzi wynosiłaby $\frac{1}{2}(d+a+c)(d+a+c-1)$. Jednak DAC nie jest grafem pełnym, ponieważ nie występują w nim krawędzie incydentne z węzłami A i C. Gdyby istniały, krawędzi incydentnych z węzłem A i C byłoby w DAC maksymalnie $\frac{1}{2}((a+c)(a+c-1)-a(a-1)-c(c-1))=ac$. Odejmując od maksymalnej liczby krawędzi pełnego grafu maksymalną liczbę potencjalnych krawędzi między węzłami A i C mamy:

$$\frac{1}{2}(d+a+c)(d+a+c-1)-ac$$

$$=\frac{1}{2}(d^2+a^2+c^2+2\mathrm{da}+2\mathrm{dc}+2\mathrm{ac}-d-a-c-2\mathrm{ac})$$

$$= \frac{1}{2}((d+a)(d+a-1)+c^2+2\mathrm{dc}-c)$$

$$= \frac{1}{2}((d+a)(d+a-1)+(d+c)(d+c-1)-d^2+d)$$

$$= \frac{1}{2}((d+a)(d+a-1)+(d+c)(d+c-1)-d(d-1))$$

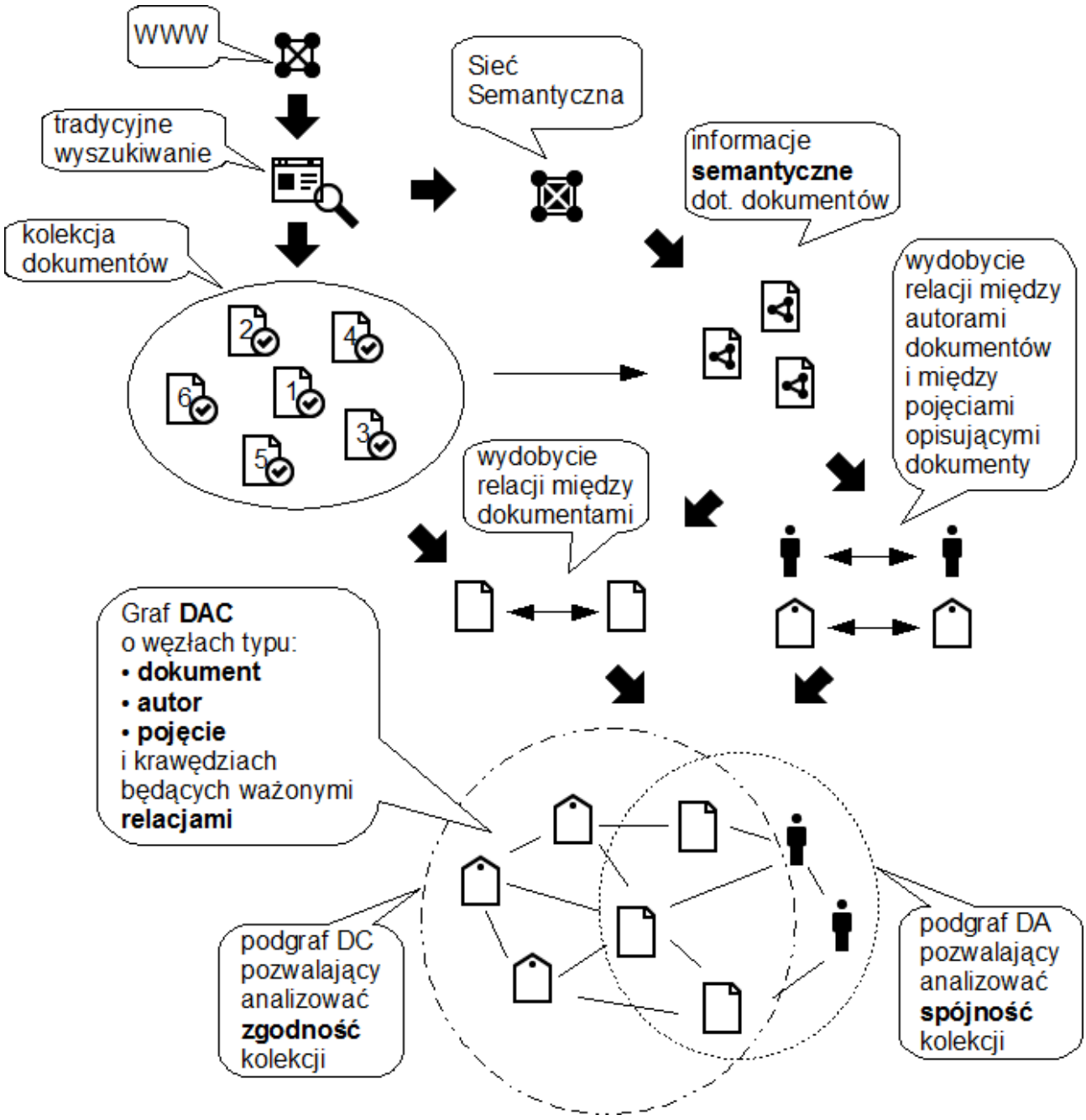
Czyli, jeżeli że d, a i c oznaczają odpowiednio liczności zbiorów węzłów typu D, A, i C, to zbudowany na tych węzłach graf DAC może posiadać co najwyżej $\frac{1}{2}((d+a)(d+a-1)+(d+c)(d+c-1)-d(d-1))$ krawędzi.

Pojęcie grafu DAC zostało wprowadzone, aby można było przy jego użyciu modelować świat rzeczywisty, jakim jest środowisko WWW. Rysunek 5.1 przedstawia schemat procesu modelowania obiektów WWW za pomocą grafu DAC. W szczególności modelowana jest kolekcja dokumentów WWW oraz powiązane z nią obiekty autorów i pojęć. Najprostszym sposobem otrzymania kolekcji dokumentów w rozumieniu definicji 5.1 jest użycie wyszukiwarki internetowej. Zakładając, że wyszukiwarka działa indeksując terminy ważone, otrzymany wynik będzie kolekcją dokumentów WWW powiązanych przynajmniej związkiem podobieństwa treści, mierzonego metodami ważenia terminów.

Alternatywą dla tradycyjnego wyszukiwania, w celu pozyskania wejściowej kolekcji dokumentów może być subskrypcja kanału RSS bloga czy witryny. Również filtrowanie dokumentów w serwisach społecznościowych według ocen, tagów, popularności, itd. może być sposobem pozyskania kolekcji wejściowej.

Równolegle do tworzenia kolekcji wejściowej z „semantycznej części współczesnej WWW" wydobywane są informacje na temat obiektów autorów i pojęć oraz związków pomiędzy tymi obiektami. Związki te można pozyskać np.: bezpośrednio z metadanych umieszczonych w dokumentach, z grafu RDF opisanego w dokumentach powiązanych z dokumentami kolekcji. Związki można też wydobyć ze źródeł nie „podlinkowanych" bezpośrednio do dokumentów, ale przechowujących informacje semantyczne na wyższym poziomie, np. na temat agregacji dokumentów.

Dokumenty oraz obiekty autorów i pojęć możemy zamodelować jako węzły DAC. Z kolei wydobyte z informacji semantycznej związki modelujemy jako krawędzie grafu. Przypisanie wag krawędziom zależne jest od algorytmu obliczania siły związków, który z kolei determinowany jest przez dostępność informacji na temat poziomów i typów związków. Konkretne wzory na wartości wag krawędzi w grafie DAC oraz na miary spójności i zgodności zostały zaproponowane w (Kopel i Zgrzywa 2008). Zweryfikowane wersje wzorów dla tych miar zostaną przedstawione w kolejnych rozdziałach.



Rysunek 5.1. Schemat modelowania WWW za pomocą grafu DAC
ikony zaczerpnięto z (Bilgil 2009)

5.3 Spójne i zgodne kolekcje

Celem modelowania kolekcji dokumentów WWW, nie tylko jako obiektów dokumentów, ale również powiązanych obiektów autorów i pojęć, była możliwość szerszej analizy kolekcji. Dodatkowe modelowanie autorów i pojęć ma umożliwić analizę spójności i zgodności. Jak przyjęto w definicji 5.2 spójność kolekcji dotyczy związków pomiędzy autorami dokumentów. Na przykład: dokumenty jednego autora będą tworzyły bardziej spójną kolekcję niż dokumenty różnych autorów. I podobnie: kolekcja dokumentów stworzona przez pracowników jednego instytutu, będzie bardziej spójna niż kolekcja artykułów z jednej sesji międzynarodowej konferencji. Zgodnie z definicją 5.3

podobna intuicja występuje w przypadku pojęć opisujących dokumenty i związków między nimi. Dokumenty mające przypisane jednakowe słowa kluczowe czy tagi powinny tworzyć bardziej zgodną kolekcję niż dokumenty o różnych słowach kluczowych. I podobnie: mając dwie kolekcje dokumentów o różnych słowach kluczowych bardziej zgodna będzie ta kolekcja, w której przypisane dokumentom terminy będą synonimami i wyrazami bliskoznacznymi, niż kolekcja, w której każdy dokument „otagowany" jest pojęciami z różnych dziedzin.

Skoro na spójność mają wpływ autorzy, a na zgodność – pojęcia, mając jako model graf DAC, do analizy spójności używany będzie graf DA, a analiza zgodności dotyczyć będzie grafu DC. Aby porównać spójność dwóch kolekcji dokumentów należy wyznaczyć miarę spójności. Wartości miary spójności dla dwóch kolekcji pozwolą stwierdzić która kolekcja jest bardziej spójna. Analogiczna miarę należy wyznaczyć dla analizy zgodności. Ponieważ modelem kolekcji jest graf, dlatego do wyznaczania miar spójności i zgodności używane są algorytmy operujące na grafach. Część algorytmów działa na grafach ważonych, jakim jest DAC, ale niektóre mogą przyjąć na wejściu jedynie zwykły graf. Zakładając, że brak związku między dwoma węzłami można modelować jako krawędź o wadze 0, stosowanie algorytmów dla grafów nieważonych nie miało by sensu, ponieważ biorąc pod uwagę samo istnienie krawędzi wejściowe grafy DA i DC zawsze byłyby grafami pełnymi. Dlatego, przed obliczeniem miar spójności lub zgodności algorytmem nie uwzględniającym wag, a jedynie istnienie krawędzi, należy grafy DA i DC przekształcić do grafów nie ważonych używając progu istotności krawędzi.

Próg istotności krawędzi to parametr jaki należy przyjąć obliczając spójność lub zgodność na podstawie nieważonych grafów. Jeśli algorytm wybrany do wyznaczania miary spójności lub zgodności działa jedynie na grafie nie ważonym, grafy wejściowe dla algorytmu wyznaczane są w następujący sposób:

- jeśli waga krawędzi jest nie mniejsza niż przyjęty próg istotności – krawędź pozostaje w grafie,
- w przeciwnym przypadku krawędź jest usuwana.

W ten sposób otrzymany graf modeluje jedynie związki wystarczająco istotne, aby zostały uwzględnione przez algorytm, który traktuje związki binarnie: jest lub nie ma związku. W szczególnych przypadkach można stosować próg istotności krawędzi również w algorytmach dla grafów ważonych. Jeśli zachodzi potrzeba można zastosować próg istotności, ale zamiast odrzucić krawędzie z wagami poniżej progu należy przeskalować wagi w przedział [próg istotności; waga maksymalna]. Takie podejście stosuje się przy badaniu podobieństwa strukturalnego, gdzie trzeba zastosować próg istotności, ale informacja o wszystkich krawędziach jest istotna.

5.4 Algorytmy wyznaczania miar spójności i zgodności (podejście ogólne)

Na wartość miar spójności i zgodności mają bezpośrednio wpływ związki pomiędzy obiektami *D*, *A* i *C*. Algorytmy wyznaczające te miary mogą brać pod uwagę sam fakt istnienia związku, czyli krawędzi łączącej węzły odpowiadające poszczególnym obiektom lub siłę takiego związku, który formalnie zapisany jest jako waga krawędzi. Spójność wyznaczana jest przez algorytmy operujące na związkach dotyczących spójności, czyli tych z grafu DA. Z kolei miara zgodności obliczana jest w oparciu o związki pomiędzy obiektami dokumentów i pojęć, czyli tych z grafu DC. Same algorytmy działające na grafach DA i DC mogą być jednakowe, jedynie interpretacja wyniku będzie różna. Pomimo że grafy DA i DC są heterogeniczne (różne typy węzłów), to często można zastosować dla nich algorytmy dla typowych grafów.

5.4.1 Współczynnik zgrupowania grafu DAC

Miarą pozwalającą ocenić stopień „pełności" grafu jest wprowadzony w (Watts i Strogatz 1998) współczynnik zgrupowania. Współczynnik ten został wprowadzony dla oceny małych

światów w kontekście sieci społecznych. Pozwala on ocenić jak daleko węzłowi i jego sąsiadom w grafie jest do grafu będącego kliką (grafem pełnym). W odniesieniu do autorów i dokumentów modelowanych przez DA, klika oznaczałaby maksymalnie spójną kolekcję dokumentów. Można to również odnieść do grafu DC i maksymalnej zgodności. Jednak należy tu dodatkowo rozważyć kwestię skierowania i siły związku.

Sąsiedztwo węzła w grafie to zbiór zawierający wszystkich sąsiadów tego węzła, czyli węzły, z którymi jest on połączony bezpośrednio krawędzią. Ponieważ każdy sąsiad musi być incydentny z krawędzią incydentną również z centralnym węzłem tworzącym sąsiedztwo, dlatego dla grafu nieskierowanego liczba sąsiadów węzła jest równa stopniowi tego węzła (każdej krawędzi incydentnej z węzłem odpowiada jeden sąsiad). Taka zależność będzie prawdziwa, tylko dla grafów, w których między dwoma węzłami może występować tylko jedna krawędź. Takim grafem jest DAC, dlatego pojęcie współczynnika zgrupowania zawęzimy do tego tylko przypadku. Stopień „pełności" takiego sąsiedztwa, czyli współczynnik zgrupowania dla tego węzła to stosunek liczby istniejących krawędzi pomiędzy węzłami sąsiedztwa do wszystkich możliwych krawędzi, jakie mogłyby istnieć między sąsiadami. Stwierdzenie „krawędzie występujące między sąsiadami" dotyczy krawędzi, dla których oba węzły, z którymi są one incydentne, są węzłami z sąsiedztwa. Zakładając, że d_i oznacza stopień węzła v_i (i przez to również liczbę jego sąsiadów), między sąsiadami v_i może wystąpić maksymalnie $\frac{d_i(d_i-1)}{2}$ krawędzi. Dlatego w przypadku grafu DAC współczynnik zgrupowania dla węzła v_i można wyrazić wzorem (7):

$$CC(v_i)=\frac{2|E(S_i)|}{d_i(d_i-1)} \tag{7}$$

gdzie:

$E(S_i)$ - zbiór krawędzi między sąsiadami v_i ,

d_i - stopień węzła v_i .

Miarą dla całego grafu (DA lub DC, w zależności czy liczymy spójność czy zgodność) w najprostszym przypadku może być wtedy uśredniony współczynnik zgrupowania dla wszystkich węzłów. Można to wyrazić wzorem (8):

$$\overline{CC}=\frac{1}{n}\sum_{i=1}^{n} CC(v_i) \tag{8}$$

gdzie:

n - liczba węzłów/obiektów w kolekcji.

W razie potrzeby, gdyby dostępność informacji o związkach między różnymi typami węzłów nie była zbalansowana lub zależałoby nam na uwypuklaniu pewnego typu związków, można by użyć jako miary spójności/zgodności średnią ważoną. Wtedy obliczając uśredniony współczynnik zgrupowania należałoby przypisać składowym współczynnikom dla różnych typów węzłów odpowiednie wagi.

Takie podejście do współczynnika zgrupowania nie uwzględnia siły związków, a jedynie ich strukturę. Aby wykorzystać wagi krawędzi modelujących siły związków można nieznacznie zmodyfikować wzór na współczynnik zgrupowania dla węzła. Przy założeniu, że wagi są znormalizowane do przedziału <0;1> można zamiast samej liczby krawędzi w sąsiedztwie uwzględnić sumę wag tych krawędzi. Wtedy we wcześniej procedurze wzór (7) należy zastąpić wzorem (9):

$$CC_w(v_i) = \frac{2\sum_{j=1}^{k_i} weight(e_j^{S_i})}{d_i(d_i - 1)} \tag{9}$$

gdzie:

k_i - liczba krawędzi w sąsiedztwie węzła v_i ,
e_j^{Si} - j-ta krawędź w sąsiedztwie węzła v_i ,
weight - waga krawędzi,
d_i - stopień węzła v_i .

W przypadku, gdyby wagi krawędzi nie były znormalizowane, należałoby dodatkowo w mianowniku wzoru (9) umieścić iloczyn liczby krawędzi i maksymalnej wagi krawędzi w sąsiedztwie v_i. Alternatywnie zamiast lokalnej normalizacji można normalizować globalnie: zamiast maksymalnej wagi krawędzi w sąsiedztwie v_i należy do iloczynu wziąć maksymalną wagę z całego grafu.

5.4.2 Grupowanie grafu DAC

W (Kopel i Zgrzywa 2008) zaproponowano miary spójności (wzór (10)) i zgodności (wzór (11)) obliczane na podstawie spójnych i zgodnych podkolekcji:

$$consistency = \frac{|cons - subc(C)|}{|C|} \tag{10}$$

$$conformance = \frac{|conf - subc(C)|}{|C|} \tag{11}$$

gdzie:

cons-subc(C) - spójna podkolekcja kolekcji C,
conf-subc(C) - zgodna podkolekcja kolekcji C.

Spójne i zgodne podkolekcje to największe grupy z wyników grupowania odpowiednio grafów DA i DC. Miara spójności lub zgodności, w zależności od wejściowego grafu, zwraca stosunek liczby dokumentów największej grupy (podkolekcji) do wszystkich dokumentów. Ogólnie taka miara byłyby maksymalna, gdyby algorytm grupujący zwrócił całą kolekcję dokumentów jako jedną grupę. Z kolei, gdyby każdy dokument kolekcji po grupowaniu znalazł się w osobnej grupie miara byłaby minimalna i równa odwrotności liczby dokumentów. W tej metodzie mierzenia spójności i zgodności do grupowania grafu użyto algorytmu MCL, ze względu na powszechny dostęp do jego implementacji.

W przeciwieństwie do używania współczynnika zgrupowania, gdzie rzeczywiste grupowanie nie ma miejsca, tutaj przed ustaleniem miar spójności i zgodności algorytm grupowania musi zakończyć działanie. Mając wynikowe podgrafy można tworzyć miary zależne od specyfiki związków modelowanych przez krawędzie DAC. Na przykład jako miarę można przyjąć, szczególnie dla spójności, **odwrotność liczby wynikowych grup**. Można również, przez analogię do wcześniej przytoczonych wzorów, określić miary jako **stosunek średniej liczby obiektów w grupach do całkowitej liczby obiektów**.

Ponieważ w pracy modelem jest graf, a składową analizy spójności i zgodności ma być grupowanie, potrzebny jest a algorytmu grupującego, który uwzględni wszystkie aspekty informacji niesionej przez DAC. Dlatego najlepiej w grupowaniu powinny się sprawdzić algorytmy grupowania grafów, jak: MCL, ICC czy GMC. Jednak, w zależności od dostępności informacji o sile związku, węzły grafu DAC można przekształcić w zbiór obiektów, a wagi krawędzi DAC w prosty sposób zamienić na odległość. Mając takie dane można zastosować dowolny algorytm grupujący zbiory na podstawie funkcji odległości.

5.4.3 Kliki w grafie DAC

Alternatywnym podejściem dla wyznaczenia miar spójności i zgodności wzorami (10) i (11) może być zasugerowana w Kopel i Zgrzywa 2008) metoda, której podstawą zamiast grupowania może być wyznaczanie klik. Klika to, wywodzący się z analizy sieci społecznych, termin oznaczający obiekty (osoby), dla których między każdymi dwoma istnieje związkek (znajomość). W odniesieniu do grafu w Moon i Moser 1965) klika zdefiniowana jest jako „maksymalny pełny podgraf". „Maksymalny" oznacza, że po dodaniu do kliki dowolnego węzła grafu podgraf ten przestaje być kliką. Można zauważyć, że skoro klika jest grafem (podgrafem) pełnym, więc średni współczynnik zgrupowania dla kliki zawsze będzie równy 1. Proces znajdowania klik można porównać do grupowania aglomeracyjnego. Na początku zakładamy, że każdy węzeł jest podgrafem pełnym, a następnie dodajemy do nich węzły do momentu, gdy dodanie kolejnego węzła spowoduje „niepełność" podgrafu. W przypadku klik istnieje jednak problem: które węzły łączyć? W przypadku tradycyjnego grupowania łączyło się węzły o maksymalnym podobieństwie/minimalnej odległości. Tutaj takiej informacji nie ma, dlatego znajdowanie kliki jest problemem bardziej złożonym obliczeniowo. Gdyby jednak założyć, że zamiast algorytmu grupowania używamy dla wyznaczania spójności i zgodności kolekcji klik wyznaczonych dla grafów DA i DC można stworzyć wzory analogiczne do tych z podrozdziału 5.4.2 Grupowanie grafu DAC. W przytoczonej już pracy (Moon i Moser 1965) oszacowano, że w grafie o n węzłach może istnieć co najwyżej $3^{n/3}$ klik. Wykorzystując tą informację do normalizacji miar spójności i zgodności można dla grafu odpowiednio DA i DC użyć wzoru (12):

$$cq = 1 - \frac{|clique(C, s)|}{3^{n/3}} \tag{12}$$

gdzie:

C - graf DA lub DC reprezentujący kolekcję dokumentów,
clique - zbiór klik wyznaczonych w grafie C,
s - parametr >3 pozwalających ograniczyć kliki zwrócone przez operator clique tylko do tych, o liczbie obiektów <=s.

Jak widać miara spójności/zgodności wyznaczona w oparciu o wzór (12) będzie maksymalna, gdy cały podgraf DA czy DC będzie tworzył klikę. Dodatkowo wprowadzony parametr *s* pozwala ograniczyć wpływ klik powyżej zadanej wielkości. Parametr ten może być szczególnie przydatny przy liczeniu **odchylenia standardowego wielkości klik**, które z kolei też może być podstawą dla miar analizy kolekcji. Niskie odchylenie standardowe wielkości klik świadczy o wysokiej jednorodności grafu, a więc może być interpretowane jako wyższa spójność/zgodność.

Ponieważ wyznaczanie klik z założenia odbywa się dla grafów nieważonych, jedyną miarą dla kliki może być jej wielkość (liczba węzłów). W przypadku grafu ważonego, podobnie jak w przypadku współczynnika zgrupowania i algorytmów grupowania, najpierw należy przekształcić graf w graf nie ważony. Najpopularniejsze przekształcenie to odrzucenie krawędzi poniżej zadanego progu istotności i usunięcie wag z krawędzi. W przypadku badania spójności i zgodności można by jednak rozważyć pozostawienie wag krawędzi. Wtedy poza samą wielkością kliki można by wyznaczyć inne cechy charakterystyczne. Na przykład, zamiast wielkości kliki przy liczeniu odchylenia standardowego można by zastosować odchylenie standardowe wag krawędzi w klice. Czyli miarą jednorodności byłoby **odchylenie standardowe odchyleń standardowych wag krawędzi w klikach**. Idąc dalej można by zastanowić się nad zasadnością kombinacji wielkości klik i cech wynikających z ich struktury w jednej mierze. Podobne rozważania można by przeprowadzić w stosunku do przedstawionych wcześniej algorytmów grupowania.

Takie rozważania i ich weryfikacja wykraczają jednak poza zakres niniejszej pracy. Dlatego w dalszej części pracy przyjęto konkretne ograniczenia i uszczegółowienia problemu pozwalające uściślić problem i umożliwiając jego formalną weryfikację.

5.5 Zastosowanie metody analizy spójności i zgodności do wyszukiwania (podejście szczegółowe)

Głównym zastosowaniem analizy spójności i zgodności kolekcji WWW jest zwiększanie skuteczności wyszukiwania. Zakładając, że nasze wyszukiwanie ogranicza się do zbioru WWW, który możemy modelować grafem DAC, wykorzystanie tego grafu daje nowe funkcjonalności. Miar spójności i zgodności można użyć do sterowania grupowaniem wyników. Zaprezentowane wyniki mogą być posortowane np. według liczności podkolekcji. Liczność spójnych i zgodnych podkolekcji regulowana jest progami spójności i zgodności, np. za pomocą suwaków. Przesunięcie suwaka maksymalnie w lewo, czyli ustawienie minimalnego progu oznacza zaprezentowanie wszystkich wyników jako jednej grupy. Z drugiej strony, ustawienie suwaka na maksymalny próg oznacza przedstawienie każdego wyniku jako osobnej, jednoelementowej grupy. Wartości pośrednie progów powodują zmiany liczby i liczności grup, czyli spójnych i zgodnych podkolekcji.

Inną funkcjonalnością, wynikającą z zastosowania grafu DAC jest możliwość zmiany perspektywy wyszukiwania. Perspektywa jest punktem widzenia dokumentów kolekcji z wybranego węzła grafu DAC. Perspektywą może być dowolny węzeł grafu, a więc: dokument, autor lub pojęcie. Wybierając dla danego zapytania konkretną perspektywę przesuwamy niejako ranking w kierunku tego węzła. To znaczy: pozycja wyniku wyszukiwania zależeć będzie nie tylko od relewancji względem pytania, ale również od siły związku z węzłem – perspektywą (np. ważona długość ścieżki). Taki reranking pozwala zadać wyszukiwarce dodatkowe kryterium. Jeśli perspektywą będzie dokument – funkcjonalność będzie podobna to znanej z Google czy WordPress'a funkcji „pokaż podobne”, z tym, że będą to „podobne”, relewantne z pytaniem. Jeśli perspektywą będzie autor, to funkcjonalność będzie przypominała personalizację wyników względem profilu użytkownika znaną z popularnych wyszukiwarek. Jednak w tej wersji będzie można „przełączyć się” na profil dowolnego użytkownika. Jeśli perspektywą będzie pojęcie, to wyniki wyszukiwania będą „zawężone” do tematyki określonej przez to pojęcie, ale nie w kwestii występowania w nich słowa kluczowego, tylko w kontekście ontologii.

Jako empiryczną weryfikację metod analizy spójności i zgodności kolekcji dokumentów WWW wybrano reranking wyników tradycyjnej wyszukiwarki przy użyciu analizy spójności i zgodności. Rerankowanie polega na analizie i ponownym ustaleniu kolejności dokumentów w porządku wynikającym z pierwotnej kolejności oraz z dodatkowych informacji, w tym przypadku spójności i zgodności. Aby zastosować spójność i zgodność do rerankingu potrzeba:

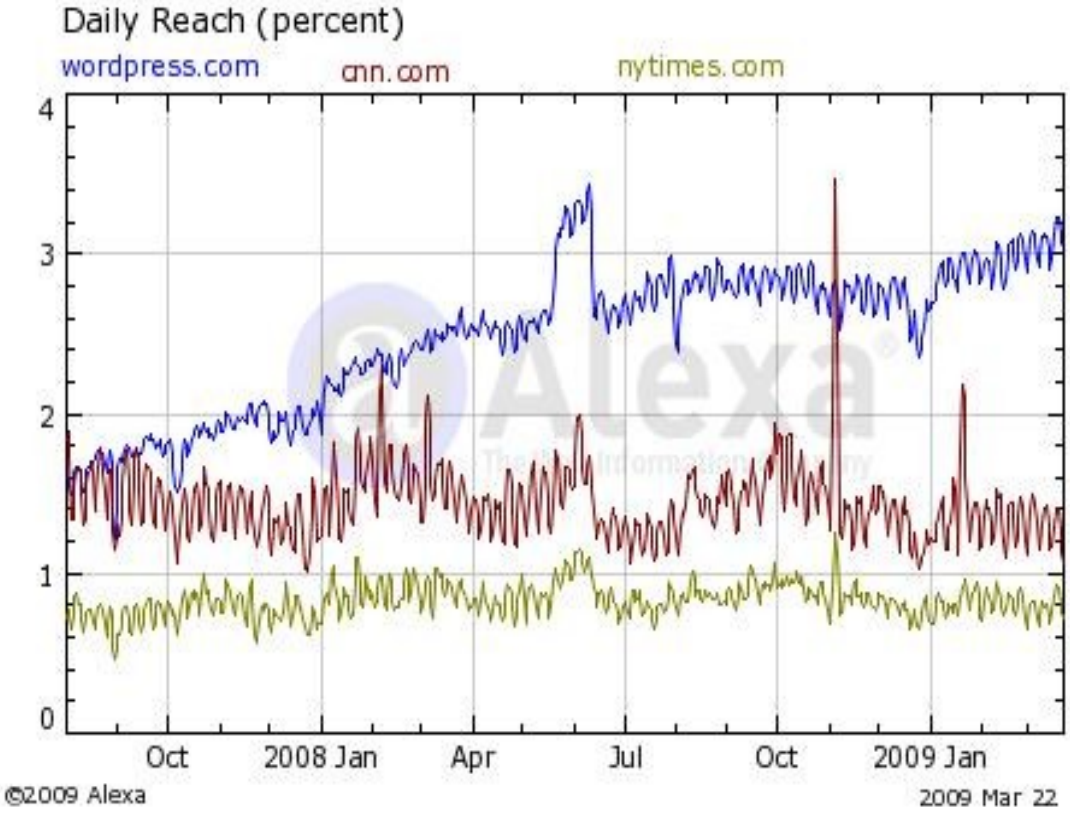
1. źródła, które umożliwili zaindeksowanie danych i metadanych semantycznych,
2. silnika wyszukiwarki, który zaindeksuje dokumenty i umożliwi wyszukiwanie,
3. aplikacji, która rozszerzy możliwości tradycyjnej wyszukiwarki o możliwość analizy spójności i zgodności oraz dokona rerankingu wyników wyszukiwania w oparciu o graf DAC.

5.6 Źródło danych i metadanych

Jako źródło dokumentów WWW przyjęto popularny serwis blogów wordpress.com. Za jego wyborem przemawia kilka faktów. Po pierwsze jest to serwis blogów. To blogi umożliwiły na szeroką skalę zwykłym użytkownikom publikowanie i zarządzanie swoimi treściami oraz ich formą. Do tej pory to webmasterzy byli osobami, które nadawały publikowanej treści ostateczną formę, niezależnie kto był jej autorem. Wraz z blogami, poza automatyczną identyfikacją autora publikującego treści, pojawiła się możliwość autorskiej ingerencji w informacje semantyczne, takie jak tagi, kategorie, daty publikacji. Dzięki temu bezpośrednio autor ma wpływ nie tylko na ostateczną formę prezentacji swojej treści, ale również na to w jaki sposób będzie ona wyszukiwana i indeksowana. Matt Mullenweg, główny developer WordPress'a od początku kładł duży nacisk na semantykę publikowanych treści. Wordpress zawdzięcza swoją popularność pionierskiemu podejściu do adaptowania takich technologii jak folksonomie czy semantyczne blogrolle. Dzięki

interfejsowi z podpowiadaniem popularnych tagów podczas tagowania posta serwis dba o utrzymanie jednolitej folksonomii. Z kolei udostępnienie podczas edycji linków w blogroll odpowiedniej formatki spowodowało, że już nie tylko użytkownicy znający składnię HTML i XFN, ale każdy może „wyklikać" i dodać do linka informację o swoim związku z likowanym użytkownikiem.

Z drugiej strony blogi to obecnie najpopularniejsze treści informacyjne WWW. Jak widać na schemacie 5.2 z serwisu Alexa badającego ruch w WWW, dzienna liczba odwiedzin wordpress.com jest kilka krotnie większa od liczby odwiedzin serwisów telewizji CNN i gazety New York Times. Choć takie dane mogą nie być miarodajne, to jednak wydają się wskazywać realny trend.



Rysunek 5.2. Porównanie popularności serwisów wordpress.com, cnn.com i nytimes.com źródło: (Alexa 2009)

Większość treści blogów zaindeksowanych na potrzeby eksperymentu jest w języku angielskim, jednak jak się okazało w badaniu przeprowadzonym przez gazeta.pl w połowie 2008 roku blogi czyta 10% Polaków. Raport (Rówińska 2008) pokazuje, że popularna jest również interakcja z blogami: 3% Polaków przyznaje się do komentowania postów w czytanych blogach, a 2% Polaków – do pisania samych postów we własnym blogu.

Jak wspomniano, serwis WordPress stawia na dostępność danych i metadanych, dlatego pobranie danych do zaindeksowania w wyszukiwarce nie było trudne. Przez kanał RSS (w formacie Atom) pobrane zostały posty (dokumenty) z wyszczególnionymi informacjami o: tytule, autorze, dacie publikacji i modyfikacji, streszczeniu (najczęściej całej treści tekstowej) i tagach/kategoriach. Taki kanał udostępniał minimum 10 najnowszych postów bloga, co jak się okazało jest aż nadto. Dodatkowo dla każdego posta, przez osobny kanał RSS, pobierane były komentarze również z semantyczną informacją. Co ciekawe, dzięki obsłudze *pingback*, komentarze do posta to nie tylko treści pisane przez użytkowników w serwisie bezpośrednio pod postem, ale również konteksty (fragment tekstu) linków do tego posta z innych postów, nierzadko z innych serwisów. Dzięki temu dostępna była informacja o linkach między postami. Z drugiej strony nasuwa się pytanie: czy

komentarze nie powinny być traktowane jako autonomiczne dokumenty WWW podlinkowane do posta? W implementacji jednak wykorzystano jedynie pierwsze zastosowanie komentarzy.

5.7 Indeksowanie obiektów WWW

Indeksowanie blogów z serwisu WordPress odbyło się dwuetapowo. Najpierw dane pobierane z WWW umieszczane były w lokalnej bazie danych. W tej bazie przeliczane były związki między obiektami DAC. Następnie zawartość bazy została przesłana do silnika wyszukiwarki, który automatycznie indeksuje dokumenty metodami ważenia tf-idf i pozwala zadawać zapytania w standardowym języku wyszukiwarek.

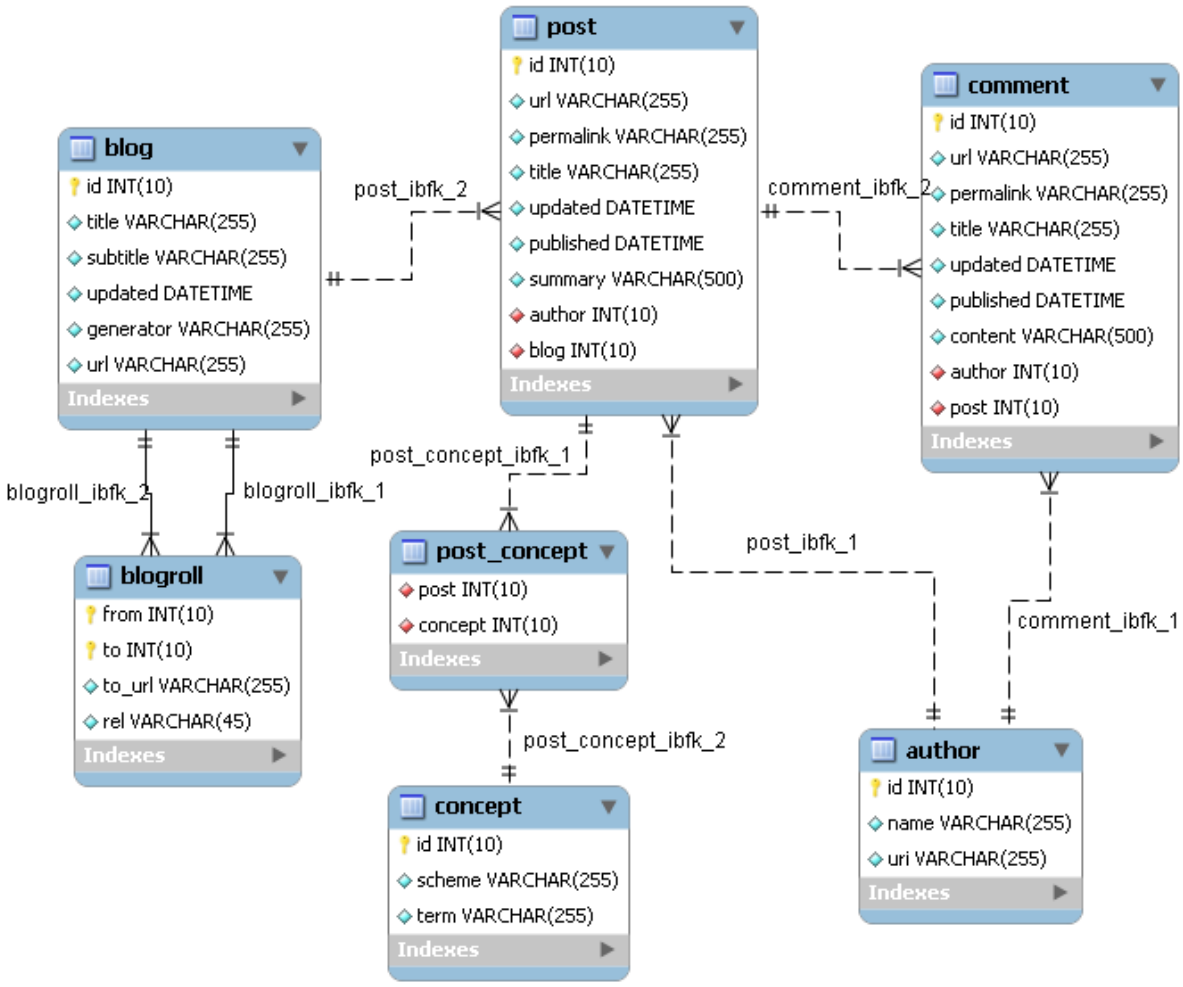
5.7.1 Relacyjna baza danych

W pierwszym etapie informacje z pobieranych i parsowanych plików XML i HTML były umieszczane w relacyjnej bazie danych o schemacie jak na rysunku 5.3. W osobnych relacjach przechowywane są informacje o blogu, należących do niego postach i przypisanych do postów komentarzach. Zarówno do postów i do komentarzy przypisani są autorzy, których URI najczęściej równało się URLowi ich bloga. Pojęcia (tagi) przechowywane są w relacji o związku wiele-do-wielu z relacją posty. Te wszystkie dane udało się wyciągnąć z kanału RSS.

Aby uzyskać informację semantyczną o związkach między autorami trzeba było parsować HTML stron głównych blogów, na których w blogroll mogły znajdować się linki opisane przez XFN. Wszystkie odnośniki do innych blogów zapisywane były w relacji blogroll, która tworzyła powiązanie między dwoma blogami. Niestety linki do tych samych blogów nie zawsze były jednakowe lub nie linkowały do głównej strony danego bloga, więc pojawił się problem redundancji informacji i potrzeby dopasowania danych dotyczących tego samego obiektu.

Oczywiście problem literówek w adresach, nazwach, tagach to ogólny problem dokumentów tworzonych w sposób zdecentralizowany. Niestety istnienie tego problemu pogarsza znacząco jakość i spójność danych, które należy wziąć pod uwagę przy ocenie działania metod wykorzystujących te dane. Za pomocą blogroll'i pozyskiwane były kolejne adresy blogów do zaindeksowania, jednak przydatne były tylko te z domeny wordpress.com. Taki sposób sięgania po kolejne blogi, znany z działania tradycyjnych pająków sieciowych, był uzupełniony przez użycie adresów sugerowanych przez sam serwis (przycisk NEXT). Te adresy jednak okazały się bardzo powtarzalne i dotyczyły tylko niewielkiej części najpopularniejszych blogów.

Sam proces indeksowania zajął kilka tygodni. Indeksowane treści postanowiono ograniczyć do publikacji z maja 2008. Takie ograniczenia: z jednej strony tylko do domeny wordpress.com, a z drugiej do postów opublikowanych tylko w jednym miesiącu, miało umożliwić kilka rzeczy. Po pierwsze, indeksowanie dokumentów sprzed miesięcy dawało prawie pewność, że dane są stabilne, czyli, np.: posty nie będą aktualizowane, nie pojawią się przy nich nowe komentarze, itp. Po drugie, porównanie działania wyszukiwarki z wyszukiwarką firmy Google. Po trzecie, realne objęcie pewnej klasy treści, w której można by stwierdzić, że w zadanych ograniczeniach indeks tych treści jest kompletny.



Rysunek 5.3. Schemat bazy danych przechowującej indeks obiektów z blogów WordPress.com

(wygenerowane w MySQL Workbench)

Jak się okazało w praktyce drugie i trzecie założenie było zbyt optymistyczne. Po tygodniach parsowania kanałów RSS z 26 719 blogów udało się zaindeksować około 129 445 postów. Google Blog Search na pytanie „blogurl: wordpress.com" z ograniczeniem czasowym do maja 2008 pokazuje, że znalazł 2 097 816 postów. Jakkolwiek liczby wyników w odpowiedzi deklarowane przez wyszukiwarki Google są przeważnie zawyżone, to jednak różnica rzędu wielkości uniemożliwia równanie się z indeksem tych wyszukiwarek.

5.7.2 Silnik tradycyjnej wyszukiwarki z parserem zapytań

Zaindeksowanie informacji pobranych z blogów WordPress pozwoliło na zorientowanie się w objętości danych i przeliczenie semantycznych związków. Jednak odpytywanie bazy możliwe było tylko w języku SQL. Dlatego sięgnięto po gotowy parser tradycyjnych zapytań do wyszukiwarek i połączony z tym projektem silnik indeksujący dokumenty terminami ważonymi.

Lucene – parser zapytań stworzony przez Apache dał początek wielu projektom tworzącym narzędzia do wyszukiwania danych. Jednym z takich projektów jest Solr – napisany w Javie silnik wyszukiwania, który oferuje m.in.:

- wyszukiwanie pełnotekstowe w oparciu o biblioteki Lucene
- definiowanie schematu silnika z podziałem na pola różnych typów danych, np.: autor, data

- obsługę faset
- komunikację za pomocą XML przez HTTP
- cach'owanie, replikację i inne pozwalające na skalowalne rozproszenie

Na potrzeby zaindeksowania informacji o postach zebranych z WordPress'a do standardowych pól schematu Solr zostały dodane deklaracje jak na rysunku 5.4.

Takie pola jak `permalink`, `author_uri`, `blog_ulr`, czyli identyfikatory indeksowane były jako `string`, które w odróżnieniu od `text` nie są tokenizowane, aby wyszukiwać wewnątrz pola (jak np. pole `summary`). Ze względu na możliwość przypisania do posta wielu tagów pole `tag` ma ustawiony atrybut pola wielowartościwego. Aby zawężać zapytania według czasu, pola publikacji i aktualizacji są typu `date`. Ponieważ Sorl pozwala na dynamiczne tworzenie faset dla każdego zaindeksowanego pola (nawet dat z określonym interwałem), dlatego wszystkie powyższe pola mają ustawione indeksowanie. Ustawiony atrybut `stored` oznacza, że wszystkie pola są dostępne podczas wyszukiwania (bezpośrednio w wynikach). Aby umożliwić wyszukiwanie bez podawania pola, ustawiono domyślne, wirtualne pole, które jest sumą pól: `title`, `author`, `tag`, `blog`, `summary`. Czyli nie określając pola przy zadawaniu pytania, wyszukiwanie odbywało się dla tych pól.

```
<field name="id" type="string" indexed="true" stored="true" required="true" />
<field name="permalink" type="textTight" indexed="true" stored="true" omitNorms="true"/>
<field name="title" type="text" indexed="true" stored="true"/>
<field name="titleSort" type="string" indexed="true" stored="false"/>
<field name="alphaTitleSort" type="alphaOnlySort" indexed="true" stored="false"/>
<field name="author" type="string" indexed="true" stored="true" omitNorms="true"/>
<field name="blog" type="text" indexed="true" stored="true" omitNorms="true"/>
<field name="tag" type="string" indexed="true" stored="true" multiValued="true" omitNorms="true"/>
<field name="summary" type="text" indexed="true" stored="true"/>
<field name="author_uri" type="string" indexed="true" stored="true" />
<field name="blog_url" type="string" indexed="true" stored="true" />
<field name="published" type="date" indexed="true" stored="true" default="NOW" multiValued="false"/>
<field name="updated" type="date" indexed="true" stored="true" default="NOW" multiValued="false"/>
```

Rysunek 5.4. Rozszerzenie standardowego schematu Solr o pola dla blogów

Samo indeksowanie danych w działającym na Tomcat'cie silniku Solr polegało na przesłaniu odpowiednio sformatowanych dokumentów XML odpowiadającym poszczególnym postom. W tym celu należało scalić rozłożone w tabelach informacje z powrotem w posty z odpowiednim oznaczeniem pól ustawionych w schemacie. Ponieważ Solr jest aplikacją RESTful[29], komunikacja z silnikiem wyszukiwania odbywa się za pomocą HTTP poprzez użycie odpowiednich adresów URL. Wykorzystany klient Solr w PHP, tworzył odpowiedni obiekt dokumentu XML na podstawie perspektywy w bazie danych i przekazywał go pod odpowiedni URL aplikacji Solr. Mając tak zaindeksowany silnik wyszukiwarki, Solr na zapytanie parsowane przez Lucene zwraca wyniki w postaci pliku XML. Co więcej wyniki są od razu rankowane domyślnym algorytmem na podstawie ważenia tf-idf. Jak widać na rysunku 5.5 przykładowy, pierwszy dokument ze 121 dokumentów w wyniku zapytania `sql OR tag:java` otrzymał w rankingu ocenę 7,5937257.

Ponieważ zapytanie składało się z 2 członów połączonych operatorem `OR`, ocena jest sumą ocen dla relewancji dokumentu względem teminu `sql` w domyślnym polu `text` i terminu `java` w polu `tag`. Same składniki końcowej oceny są iloczynem wag zapytania i wag pól, względem obu członów. Czynniki tych iloczynów to z kole iloczyny tf-idf i wartości normalizujących. Na przykład ocena składowa 1,5776818 wynika z tego, że termin `sql` wystąpił w tym dokumencie 4 razy (*tf*), a w ogóle wystąpił w 77 dokumentach (*idf*) oraz z normalizacji domyślnego pola `text`.

29 RESTful – to przymiotnik opisujący aplikację spełniającą wymogi REST. REST (ang. *Representational state transfer*) to styl architektury oprogramowania dla rozproszonych systemów hipermedialnych. Przykładem takiej architektury jest architektura WWW - (R. T. Fielding 2000)

```
7.5937257 = (MATCH) sum of:
  1.081148 = (MATCH) weight(text:sql in 12391), product of:
    0.6852763 = queryWeight(text:sql), product of:
      8.414303 = idf(docFreq=77)
      0.08144184 = queryNorm
    1.5776818 = (MATCH) fieldWeight(text:sql in 12391), product of:
      2.0 = tf(termFreq(text:sql)=4)
      8.414303 = idf(docFreq=77)
      0.09375 = fieldNorm(field=text, doc=12391)
  6.5125775 = (MATCH) weight(tag:java in 12391), product of:
    0.7282831 = queryWeight(tag:java), product of:
      8.94237 = idf(docFreq=45)
      0.08144184 = queryNorm
    8.94237 = (MATCH) fieldWeight(tag:java in 12391), product of:
      1.0 = tf(termFreq(tag:java)=1)
      8.94237 = idf(docFreq=45)
      1.0 = fieldNorm(field=tag, doc=12391)
```

Rysunek 5.5. Wyjaśnienie oceny pierwszego dokumentu w odpowiedzi na zapytanie „sql OR tag:java”

Aby zweryfikować empirycznie analizę spójności i zgodności należało, dla tak skonfigurowanego silnika wyszukiwarki, stworzyć aplikację interfejsową. Aplikacja interfejsowa to aplikacja WWW, za pomocą której użytkownik przeprowadza analizę spójności i zgodności wyników oraz może ocenić jej jakość.

5.8 Aplikacja wykorzystująca algorytmy analizy spójności i zgodności

Budowa aplikacji wyszukiwarki WWW miała na celu umożliwienie szybkiej analizy spójności zwykłemu, niezaawansowanemu użytkownikowi WWW. Zakładając, że potrzebą informacyjną użytkownika jest nie tyko wyszukanie konkretnej informacji, ale również - jak zdefiniowano we wstępie, możliwość analizy wyników wyszukiwania - słowo wyszukiwarka przestaje być wystarczająco relewantne. Nasuwa się analogiczny neologizm „analizerka”. Być może właśnie tak powinny się nazywać narzędzia WWW, które oferują założone tu funkcjonalności. Uściślając: aplikacja, tworzona na potrzeby weryfikacji metod opisanych w pracy, powinna pozwolić:

- uzależniać wyniki zapytania od przyjętej perspektywy,
- ustalać spójność i zgodność wynikowej kolekcji na podstawie ustawionych progów istotności,
- grupować dokumenty kolekcji,
- poprawiać ranking dokumentów na ważeniu terminów.

Uzależnianie wyników od perspektywy danego węzła DAC w stworzonej aplikacji sprowadza się do zawężania/poszerzania wyników danego zapytania poprzez używanie faset, jak na rysunku 5.6.

Fasety są tworzone dla autorów i tagów, czyli węzłów *A* i *C*. Węzły *D* wizualizowane są w tradycyjny sposób jako lista tytułów dokumentów (postów) wraz z fragmentem treści (streszczeniem) i metadanymi: autor, data aktualizacji, blog, z którego pochodzi post i przypisane tagi. Analiza spójności odbywa się poprzez regulowanie progów istotności związków. W przypadku zgodności odpowiednim suwakiem można ustawić próg istotności związku między pojęciami (tagami). W przypadku spójności podobny próg ustawiany jest dla związków autorów.

Grupowanie dokumentów odbywa się poprzez grupowanie grafu DAC. Ponieważ grupowanie ma wpływ na ranking, musi ono być realizowane dla każdej kolekcji wynikowej, a więc przy każdym zadaniu pytania i przy każdej zmianie perspektywy. Grupowanie każdorazowo dotyczy innej kolekcji, a więc i innego grafu DAC, dlatego DAC jest budowany na bieżąco na podstawie: kolekcji wynikowej (zadane zapytanie i ustawiona perspektywa) i ustawionych progów istotności dla spójności i zgodności.

Rysunek 5.6. Interfejs autorskiej wyszukiwarki umożliwiającej analizę spójności i zgodności oraz reranking w oparciu o DAC

5.8.1 Wagi krawędzi grafu DAC

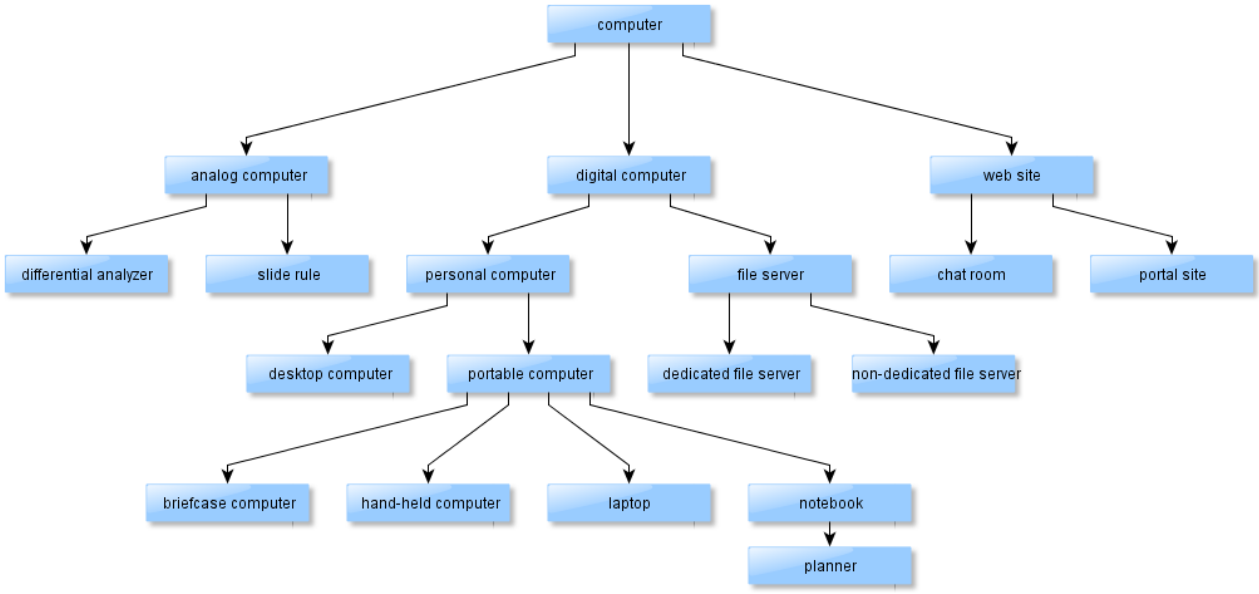
Węzły DAC są dostępne wprost z silnika wyszukiwarki jako pola wyników wyszukiwania. Natomiast przeliczone związki przechowywane są niezależnie w bazie danych. Kluczowe dla analizy spójności związki między autorami wyznaczane są na podstawie deklaracji autorów w postaci semantycznych linków w blogroll'ach. W celu uproszczenia obliczeń przyjęto, że siłą wagą związku między autorami jest odwrotność najkrótszej odległości między węzłami typu A, liczonej jako liczba krawędzi. Odległość jednej krawędzi między autorami wyznaczają same linki. Pozostałe krawędzie wraz z wagami zostały dodane poprzez znajdowanie pośrednich powiązań węzłów *A* za pomocą łańcuchów (węzłów *A* i krawędzi parami incydentych).

W na podobnej zasadzie przyjęto siły związków między pojęciami. Podobnie, wagą krawędzi takiego związku jest odwrotność odległości w grafie, jednak podstawą nie są semantyczne linki, tylko graf ontologii stworzony przy użyciu WordNet'u. Użyta biblioteka WordNet::Similarity daje możliwości wyznaczenia podobieństwa między pojęciami metodami opisanymi w różnych pracach naukowych, co podsumowano w Pedersen, Patwardhan, i Michelizzi 2004). Jednak, przez analogię do spójności, za wagi między pojęciami, wpływające na zgodność, wybrano miarę podobieństwa jako odwrotność długości ścieżki pomiędzy węzłami pojęć w ontologii zbudowanej na podstawie WordNet'u.

Na rysunku 5.7 przedstawiono przykładowy fragment omawianej ontologii. Widać z niego, że odległość między komputerem osobistym („personal computer") a laptopem to 2 krawędzie, czyli siła związku między tymi pojęciami to ½. Z kolei długość ścieżki między laptopem i serwerem plików („file server") to 4, więc siła związku między nimi wynosi ¼ .

Dla ponad 130 000 majowych postów w WordPress użyto ponad 400 000 różnych tagów. Po znormalizowaniu (małe litery) i przefiltrowaniu tagów według wyrażenia regularnego `"^[a-z ]*$"`, czyli tylko takich, które zawierają małe litery i spacje, zostało ich ok. 154 000. Ze względów obliczeniowych nie można było przeliczyć wag między taką liczbą węzłów. Stosując *stemming* sprawdzono ile z nich znajduje się w WordNet'cie: ~21 000. To wciąż było za dużo. Ostatecznie z

~21 000 wybrano takie, które zostały użyte więcej niż kilka razy. To pozwoliło wybrać najpopularniejszych ~2 000 tagów, między którymi wyznaczono podobieństwa.



Rysunek 5.7: Fragment ontologii "jest rodzajem" Wordnet'u: drzewo hiponimii dla węzła-pojęcia „komputer”

Dodatkowym uproszczeniem, które wymusiło ograniczenie wydajnościowe był wybór znaczenia terminu. Funkcje z biblioteki WordNet::Similarity liczące podobieństwa wymagają podania terminu, określenia części mowy, którą on reprezentuje i numer znaczenia. Ponieważ popularne terminy mogą mieć nawet do kilkunastu znaczeń i występować zarówno jako rzeczownik, czasownik czy przymiotnik, przyjęto ograniczenie zgodne z intuicją słów kluczowych. Przy ustalaniu słów kluczowych, a więc i przy tagowaniu wymaga się przeważnie, że takie wyrażenie jest rzeczownikiem w mianowniku liczby pojedynczej. W związku z tym funkcja obliczająca podobieństwo tagów zakładała, że są to rzeczowniki i brała pod uwagę tylko ich pierwsze znaczenie.

Głównym elementem wpływającym na siłę związku między dokumentami był fakt wzajemnego cytowania, czyli wystąpienie hiperłącza w jednym dokumencie do drugiego. Dzięki technice *pingback* nie potrzeba było parsować treści każdego posta w poszukiwaniu linków. Jest to szczególnie ważne, ze względu na fakt, że w kanale RSS dostępne są jedynie streszczenia postów. Zatem, aby przeszukać treści całych postów trzeba by ściągnąć i parsować 130 000 stron HTML, to z przytoczonych wcześniej ograniczeń sprzętowych nie wchodziło w grę. Na szczęście rozwiązaniem okazało się wizualizacja *pingback* w WordPress'sie. Okazało się, że każdy post zawierający odnośnik do innego posta jest umieszczany, wraz z fragmentem tekstu otaczającego ten link, pod linkowanym postem jako komentarz. Post cytujący, i przez to pojawiający się jako komentarz, nie musi być nawet z tego samego serwisu blogów, wystarczy, że oba serwisy obsługują *pingback*. Aby odróżnić zwykły komentarz wpisywany przez użytkownika pod postem od komentarza stworzonego automatycznie przez system z innego posta na podstawie informacji *pingback* wystarczy sprawdzić czy URI autora komentarza nie jest URLem posta. Ponieważ serwisy blogów powszechnie używają *pretty permalink*ów[30], jako URLi dla postów, co oznacza, że w URL zawiera najczęściej datę publikacji posta, łatwo można odróżnić adresy postów od URI autorów.

Czyli: jeśli URI autora komentarza jest permalinkiem (URLem) posta oznacza to, że ten post

30 Pretty permalink – semantyczny, stały URL

cytuje lub co najmniej zawiera link do posta, pod którym znajduje się taki komentarz. Ten fakt pozwala na łatwe znalezienie powiązań między dokumentami. Analogicznie do przypadku autorów wagą może być również pośrednie powiązanie. Jeśli dwa posty powołują się na trzeci, to odległość między nimi wynosi 2 krawędzie, czyli ich związek to krawędź o wadze ½. Jednak, ponieważ takie cytowanie nie jest częste, gdy tylko ono miało wpływ na siły związków dokument – dokument, większość była by równa zero. Dlatego drugim czynnikiem, wpływającym na wagę krawędzi incydentnej z dwoma węzłami typu *D*, jest ranking wyszukiwarki tradycyjnej (ważącej terminy). Siłę tego powiązania dokument–dokument wyraża wzór (13):

$$w_s(d_i, d_j) = 1 - \frac{|rank(d_i) - rank(d_j)|}{range(rank)} \tag{13}$$

gdzie:

rank(d_i) - ocena dokumentu d_i w rankingu kolekcji dokumentów,
range(rank) - zakres wartości rankingu kolekcji dokumentów.

Jaki widać ze wzoru (13) składnik wagi krawędzi związku między dokumentami uwzględnia podobieństwo terminów ważonych w dokumentach. Liczony jest jako bezwzględna różnica rang dokumentów, znormalizowana według zakresu rang w rankingu, dopełniona do jedynki. Czyli ten składnik wagi jest tym większy, im bliżej siebie znajdują się dokumenty w rankingu, a przyjmie zero dla pierwszego i ostatniego dokumentu w rankingu.

Składnik wagi tf-idf użyty został ze współczynnikiem ¼, czyli na ostateczną wagę krawędzi ma wpływ 3 razy mniejszy niż fakt istnienia linków między dokumentami.

Wagi krawędzi incydentych z różnymi typami węzłów, czyli *D–C* i *D–A* dotyczyły semantyki, która jest binarna, czyli: obiekt typu *A* może być lub może nie być autorem dokumentu, tag może opisywać lub nie być przypisany do dokumentu. Aby siły związków nie były binarne przyjęto następującą intuicję: im więcej tagów opisuje dokument, tym mniejsza jest siła ich związku z dokumentem. W praktyce wszystkie wagi krawędzi incydentych z tagami i jednym dokumentem były jednakowe i równe odwrotności liczby tych tagów przypisanych do dokumentu. W przypadku autorów można by przyjąć jednakową intuicję. A nawet posunąć się dalej, bo o ile przypisane tagi najczęściej są posortowane alfabetycznie, o tyle kolejność autorów dokumentu najczęściej ma znaczenie. Jednak w praktyce blogów każdy post może mieć dokładnie jednego autora, więc waga związku *D–A* pozostała binarna.

Przyjmując powyższe wagi w algorytmie tworzenia grafu DAC, analizę spójności i zgodności wyników wyszukiwarki można sparametryzować progiem istotności dla wag krawędzi. Przyjęty próg istotności wpływa na strukturę grafu, co odzwierciedlane jest przez miary spójności i zgodności. Ze względu na grupowanie grafu używane do rerankingu, najlepszymi miarami spójności i zgodności byłyby te proponowane w podrozdziale 5.4 Algorytmy wyznaczania miar spójności i zgodności (podejście ogólne) s. 77. Jednak ponownie praktyka zmusza do uproszczeń ze względu na ograniczenia obliczeniowe. Ponieważ miary te muszą być liczone w czasie rzeczywistym dla każdej zmiany perspektywy, przyjęto prostsze w obliczeniach wzory. Zakładając, że informacja strukturalna nie jest krytyczna w przypadku spójności i zgodności, zamiast średniego współczynnika zgrupowania za miary spójności i zgodności przyjęto odpowiednio: gęstości grafów DA i DC. Gęstość wyznaczana wzorem (14) jest tradycyjną gęstością nieskierowanego grafu nieważonego:

$$g = \frac{2|E(G)|}{n(n-1)} \tag{14}$$

gdzie:

E(G) - zbiór krawędzi grafu DA lub DC,
n - liczba węzłów typu D i A lub typu D i C.

. Rozważano użycie gęstości dla grafu ważonego zdefiniowanej w analogiczny sposób do średniego współczynnika zgrupowania dla grafu ważonego (wzór (9)), jednak ze względu małą liczbę niezerowych krawędzi grafów DAC taka miara spójności i zgodności dawałaby jeszcze niższe wartości niż i tak te niewielkie zwracane przez *g* dla krawędzi binarnych.

Zastanawiając się, czy gęstość może być jakąś aproksymacją średniego współczynnika zgrupowania można łatwo zauważyć, że granicą bezwzględnej różnicy gęstości i średniego współczynnika zgrupowania przy grafie o stałej liczbie węzłów dążącym do grafu pełnego jest zero, co można wrazić wzorem (15):

$$\lim_{k \to \frac{n(n-1)}{2}} |g - \overline{CC}| = 0 \tag{15}$$

gdzie:

k - liczba krawędzi,

n - liczba węzłów.

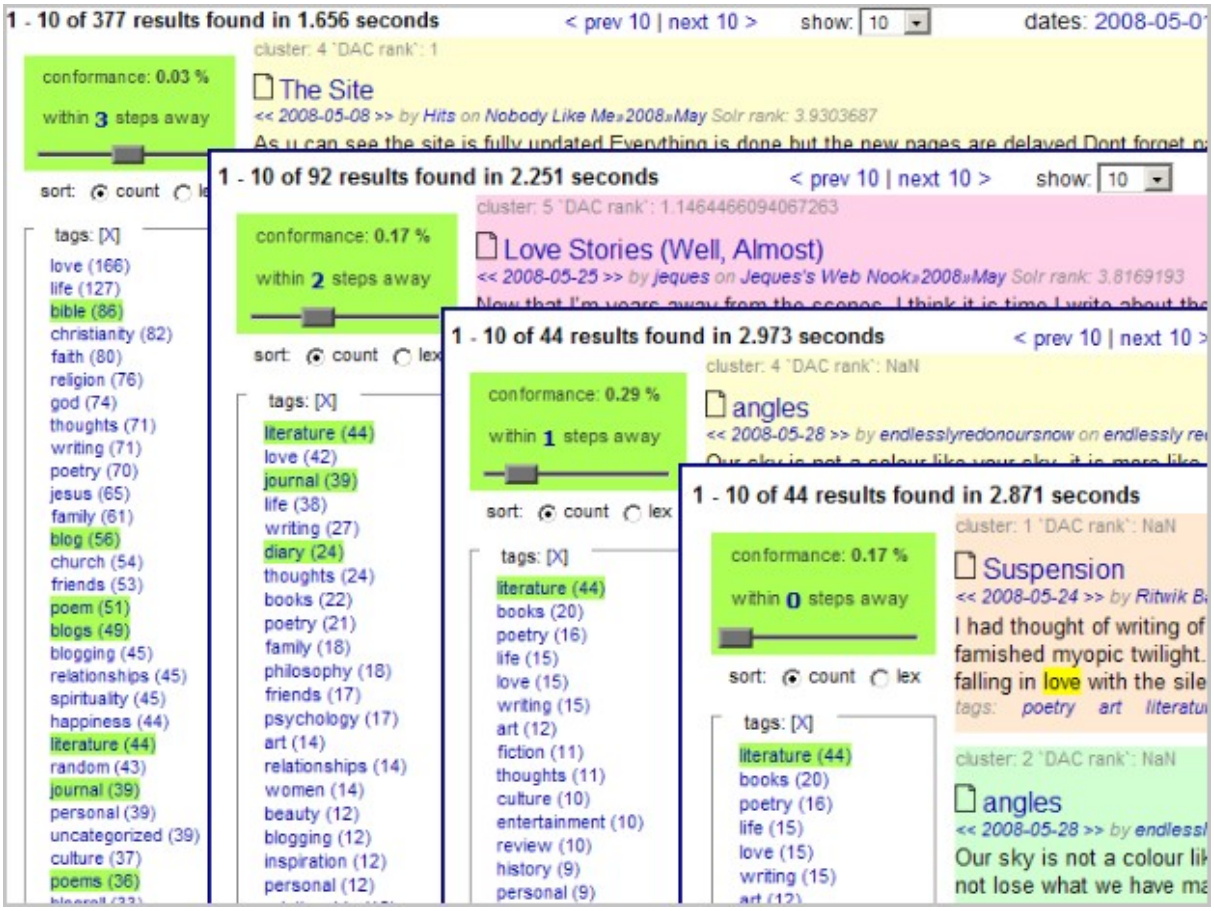
Przy założeniu, że $g - \overline{CC}$ jest funkcją liczby krawędzi *k*, monotoniczność tej funkcji jest zmienna. Weźmiemy jednak pod uwagę średni stopień węzłów grafu, który ma bezpośredni wpływ na współczynnik zgrupowania, ale nie dotyczy samej struktury. Skoro gęstość zależy liniowo od średniego stopnia węzła dla stałej liczby węzłów, pokazuje ona dobrze ten aspekt współczynnika zgrupowania, który nie dotyczy struktury grafu. Poza tym gęstość, jako miara spójności i zgodności, lepiej spełnia postulaty dobrej miary określone w Pipino, Lee, i Wang 2002), do której należą również: przejrzystość definicji, możliwość prezentacji i wnikania w wyższy poziom szczegółowości.

Oczywiście gęstość jako miara dla grafu DAC uwzględnia tylko te krawędzie, które przekraczają próg istotności. Im mniejszy próg istotności, tym większa gęstość, a więc spójność i zgodność. Chcąc dać użytkownikowi proste i intuicyjne narzędzie analizy spójności i zgodności, ustawianie progu zgodności połączone zostało z rozszerzaniem/zawężaniem zapytania do wyszukiwarki. Suwak, którym kontroluje się próg wag krawędzi A-A (w przypadku spójności) i krawędzi C-C (w przypadku zgodności), automatycznie rozszerza/zawęża wybrane perspektywy – odpowiednio – węzłów autorów i węzłów pojęć. Zmiana perspektywy węzła C to wybranie taga z listy faset jak na rysunku 5.8. Zgodnie z intuicją faset wybranie jednego elementu zawęża wyniki do liczby podanej w nawiasie. W przypadku tagów wybranie jednego z listy faset powoduje dodanie do zapytania „AND tag: *wybrany tag*". Oznacza to, że wyniki dodatkowo zostają ograniczone do tylko takich postów, które mają przypisany *wybrany tag*. *Wybrany tag* staje się wtedy perspektywą analizy zgodności. Analogicznie działa to w przypadku faset autorów i analizy spójności.

Ponieważ miary wagi jako odwrotność liczby krawędzi trudno zdyskretyzować jako równomierne odległości na pasku suwaka, suwak kontroluje odwrotność progu istotności, czyli po prostu liczbę krawędzi między węzłami. W związku z tym: 0 oznacza maksymalny próg istotności, 6 oznacza minimalny próg istotności = 1/6. Takie przedstawienie progu wartości wiąże się również z intuicją zawężania czy poszerzania. Tradycyjne zawężanie fasetami dostępne jest przy ustawieniu 0, czyli maksymalnego progu. Przy przesuwaniu suwaka w kierunku większych wartości, czyli zmniejszaniu progu dla krawędzi *A–A* czy *C–C* , czyli odpowiednio zmniejszaniu wymaganej spójności czy zgodności, do wybranych perspektyw dodawane są odpowiednio autorzy lub tagi, którzy znajdują się grafie w odległości nie większej niż wartość suwaka.

Prześledźmy to na przykładzie z rysunku 5.8. Rysunek przedstawia cztery widoki wyników zapytania „*love*" z wybraną perspektywą zgodności „*literature*". W prawym dolnym rogu widać przypadek, gdy interesują nas tylko węzły grafu DC zwrócone przez wyszukiwarkę. Gęstość grafu DC, czyli zgodność wynosi 0,0017 (czyli 0,17%). Przesunięcie suwaka na 1 praktycznie nie zmienia progu istotności, ponieważ odwrotność jedynki nadal jest maksymalną możliwą wagą krawędzi. Zmienia się jednak to, że do analizowanego grafu DC dodane zostają węzły *C*, które znajdują się w DAC w odległości jednej krawędzi. Jak się okazuje poprawiają one gęstość grafu

DC, a więc zgodność, ale nie zmieniają liczby znalezionych dokumentów, ponieważ nie są przypisane do żadnego z postów zwracanego przez samo zapytanie „*love*”.



Rysunek 5.8. Analiza zgodności wyników zapytania "love" z perspektywy "tag: literature" dla różnych progów istotności

Przy kolejnym przesunięciu suwaka w prawo, na pozycję 2, tagi będące węzłami w odległości nie większej niż dwie krawędzie od węzłów tagów przypisanych do pierwotnie zwróconych przez wyszukiwarkę postów, są dodawane jako alternatywne kryterium dla perspektywy zgodności. Czyli ograniczenie postów do tych otagowanych jako „*literature*” (literatura) rozszerzono do takich, które mogą też być otagowane jako „*journal*” (pismo), „*diary*” (pamiętnik) czy inny termin znajdujący się nie dalej niż 2 krawędzie od „*literature*” w grafie WordNet. W pewnym sensie można traktować zwiększanie zgodności jako używanie synonimów, wyrazów bliskoznacznych i terminów o coraz mniej podobnych znaczeniach. Przez dodanie nowych tagów do zapytania zwiększyła się liczba dokumentów wynikowych z 44 do 92. Automatycznie zwiększyła się liczba węzłów w grafie DAC, przez co zmieniła się gęstość grafu DC – zgodność zmniejszyła się z 0,29% do 0,17%.

Przy przesunięciu suwaka na pozycję 3 analogicznie dodano do zapytania jeszcze więcej tagów (o sile związku z „*literature*” $\geq \frac{1}{3}$), co poszerzyło listę wyników do 377, a zgodność zmniejszyło do 0,03%. Podobna sytuacja ma miejsce w przypadku suwaka spójności. Ustawiając nim wartości progu istotności dla wag krawędzi między autorami mamy wpływ na spójność, czyli gęstość grafu DA. Możemy również dodawać do wyników posty autorów będących w określonym związku z autorem wybranym jako perspektywa spójności. Różnica w przypadku faset spójności jest taka, że w – odróżnieniu od tagów – post może mieć tylko jednego autora. Z tego wynika, że wybranie węzła *A* jako perspektywy spójności przy wysokim progu spójności najczęściej skraca listę faset autorów do tej wybranej, jednej pozycji. Jest to jedynak jedynie niewielkie utrudnienie

nawigacyjne.

Co bardzo istotne, niezależnie regulowane progi istotności dla spójności i zgodności nie powodują rozdzielenia miar spójności i zgodności. Zwiększanie się liczby postów w wyniku, spowodowane zmniejszaniem progu zgodności, ma wpływ na cały graf DAC: zmienia się nie tylko gęstość DC, czyli zgodność, ale i DA, czyli spójność. To pokazuje **sens modelowania WWW jako jednego grafu DAC, a nie dwóch niezależnych DA i DC**. Widać to dobrze na wykresie zależności liczby postów w wyniku od progów spójności i zgodności na rysunku 1 w dodatku C s. 131. Podobną zależność można zauważyć na rysunku 2, z tą różnicą, że na osi Z zamiast węzłów *D* są węzły C.

Uniezależnienie progów istotności dla wag krawędzi A-A (między autorami) i C-C (między pojęciami) stawia dodatkowy problem: jakie przyjąć progi istotności dla pozostałych 3 rodzajów związków: D-A, D-C i D-D? Na potrzeby eksperymentu arbitralnie przyjęto jednakowy próg równy ½.

5.8.2 Algorytm rerankingu

Aby stworzyć miarę pozwalającą ocenić jakość analizy spójności i zgodności kolekcji postów zwróconych przez tradycyjną wyszukiwarkę zaproponowano algorytm rerankingu tych wyników. Dzięki temu możliwe jest użycie tradycyjnych miar porównujących rankingi – jak współczynniki korelacji Kendalla i Spearmana – do oceny przydatności i jakości analizy spójności i zgodności kolekcji dokumentów WWW. Ogólny plan działania tego algorytmu przewiduje następujące etapy:

1. Tradycyjne wyszukiwanie w oparciu o terminy ważone,
2. Budowanie grafu DAC dla wyników wyszukiwania,
3. Analiza spójności i zgodności DAC,
4. Grupowanie grafu DAC,
5. Tworzenie rerankingu na podstawie rankingu tradycyjnej wyszukiwarki i pogrupowanych dokumentów.

Pierwsze trzy etapy zostały opisane w poprzednim podrozdziale. Czwarty etap sprowadzał się do użycia algorytmu MCL opisanego w 4.3 Grupowanie grafu s. 67. Za pomocą skryptu PHP, graf – w postaci trójek: węzeł1, węzeł2 i waga krawędzi z nimi incydentnej – został przepisany z bazy MySQL do formatu tekstowego, wejściowego dla aplikacji `mcl`. Następnie skrypt wywoływał polecenie linii komend `mcl`, którego – poza plikiem z grafem wejściowym – jedynym parametrem był *inflation*. Za wartość tego parametru, pozwalającego sterować ziarnistością grupowania, przyjęto 2, zgodnie z zaleceniami autora algorytmu.

Wynikowe grupy węzłów przefiltrowano odrzucając węzły inne niż D, czyli niż posty z pierwotnego rankingu. Następnie grupy ustawiono w kolejności zgodnej z następującymi przesłankami:

1. Skoro głównym i intuicyjnym dla użytkownika sposobem wyszukiwania konkretnych dokumentów jest zgodność terminów ważonych, to pierwotny ranking wyników jest dla użytkownika istotny.
2. Skoro progi spójności i zgodności zostały zadane przez użytkownika, to znaczy, że spodziewa się on w wynikach nie tylko postów relewantnych pod względem tf-idf, ale również spójnych i zgodnych z postami pasującymi do pytania bezpośrednio przez terminy ważone.
3. Skoro kryterium grupowania są spójność i zgodność, to posty wewnątrz jednej grupy stanowią kolekcję bardziej spójną i zgodną, niż posty z różnych grup. Dzięki temu posty spójne i zgodne z postami o wysokiej ocenie według pierwotnego rankingu tworzą dla nich spójny i zgodny kontekst.

Mając na uwadze powyższe przesłanki zaproponowano algorytm rerankingu według założeń:

1. Wewnątrz grupy kolejność dokumentów jest zgodna z pierwotnym rankingiem.

Dokumentom posortowanym malejąco według ocen rankingu przypisywane są rosnące numery porządkowe czyli: numer 1 zostaje przypisany dokumentowi o największej ocenie rankingu, a ostatni numer porządkowy dokumentowi o najmniejszej ocenie. Numery porządkowe używane są następnie zamiast ocen.

2. Kolejność grup w wynikach wyszukiwania wyznacza funkcja zapisana wzorem (16). Grupy sortowane są rosnąco według wartości funkcji, a więc grupy o najmniejszej wartość znajdują się na początku listy wyników.

$$R(C_i) = \overline{r(C_i)} - \frac{\sigma(r(C_i))}{max(\sigma(r(C_i)))} \tag{16}$$

gdzie:

$\overline{r(C_i)}$ *- średnia numerów porządkowych dokumentów grupy C_i według pierwotnego rankingu,*

$\sigma(r(C_i))$ *- odchylenie standardowe numerów porządkowych dokumentów grupy C_i według pierwotnego rankingu.*

Głównym elementem wpływającym na ocenę grupy tej funkcji jest, zgodnie z pierwszym punktem powyższych przesłanek, pierwotny ranking liczony jako średnia numerów porządkowych dokumentów w grupie. Ta średnia pomniejszona jest o znormalizowane odchylenie standardowe numerów porządkowych. Jest to zgodne z drugim i trzecim punktem przesłanek. Dzięki takiemu pomniejszeniu na najwcześniejszych pozycjach listy wyników znajdują się zarówno dokumenty wysoko ocenione w pierwotnym rankingu, jak i spójne i zgodne z nimi dokumenty najgorzej ocenione w pierwotnym rankingu. Tą własność wzoru (16) można zobrazować w następujący sposób:

1. pierwotnie wysoko ocenione dokumenty nie zmieniają swojej względnej pozycji w rerankowanych wynikach wyszukiwania;
2. pierwotnie nisko ocenione dokumenty, które znalazły się w grupie z pierwotnie dobrze ocenionymi dokumentami, są w wynikach wstawiane zaraz pod te dobrze ocenione dokumenty, tworząc dla nich spójny i zgodny kontekst.

Pseudokod na rysunku 5.9 przedstawia działanie algorytmu bardziej szczegółowo. Pierwsza funkcja *Aktualizuj_podobieństwa()* (linie 1-8), działa po stronie serwera SQL. Jej zadaniem jest przeliczenie podobieństw pomiędzy zaindeksowanymi obiektami WWW, czyli wag krawędzi modelujących związki między dokumentami, autorami i pojęciami. Działanie tej funkcji wymaga utrzymywania trzech homogenicznych grafów:

- D – modelującego związki między postami na podstawie linków,
- A – modelującego związki między autorami blogów na podstawie linków XFN,
- C – modelującego związki między pojęciami na podstawie ontologii WordNetu.

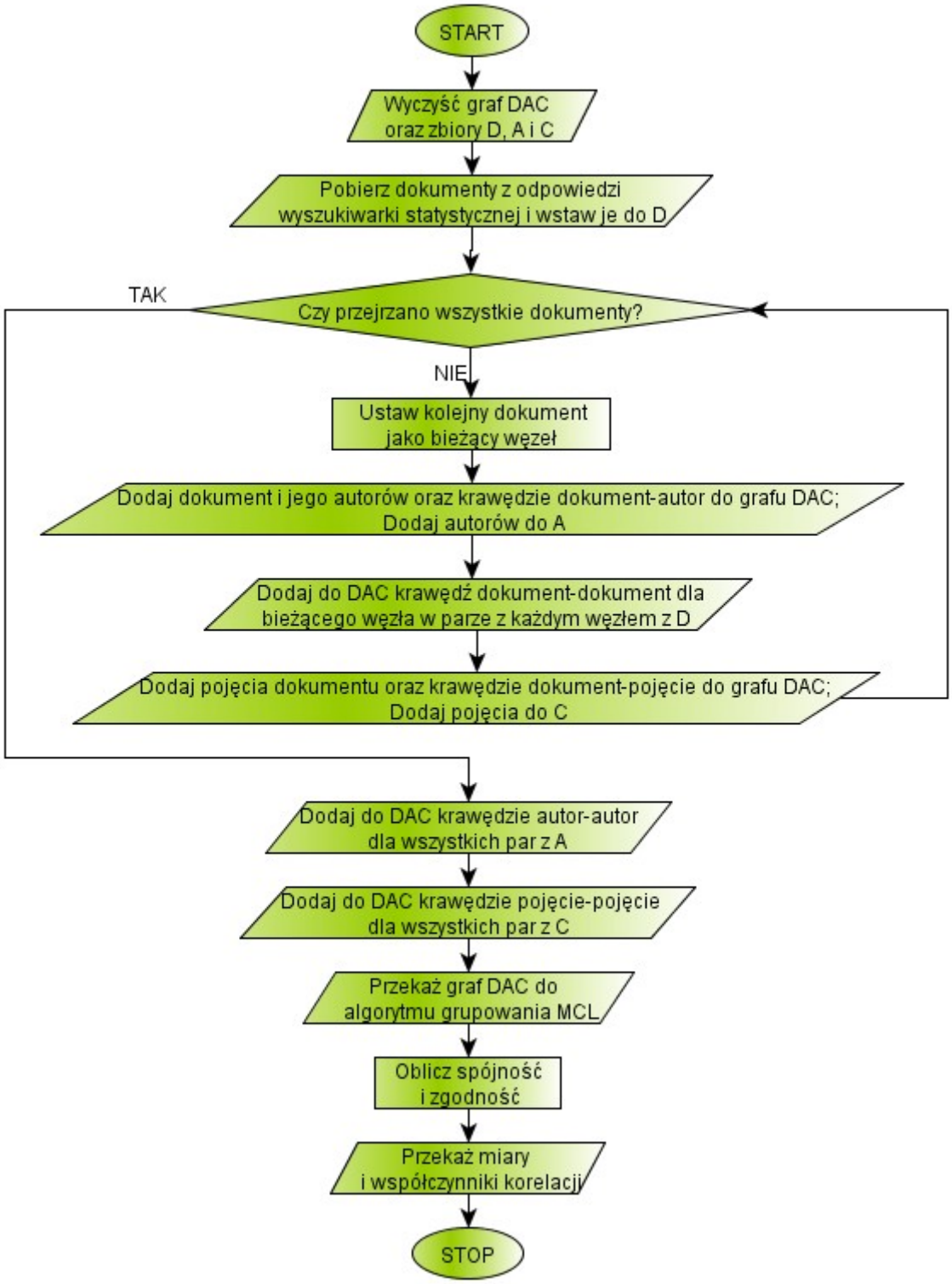
```
1.  // po stronie serwera (SQL)
2.  Aktualizuj_podobieństwa() {
3.    Dla całej bazy indeksu {
4.      Wyznacz wagi krawędzi DD(Pingback do 2 stopni oddalenia);
5.      Wyznacz wagi krawędzi AA(XFN do 6 stopni oddalenia);
6.      Wyznacz wagi krawędzi CC(Wordnet do 6 stopni oddalenia);
7.    }
8.  }
9.
10. // po stronie serwera (PHP)
11. Grupuj_DAC() {
12.   Jeśli (zaktualizowano indeks) {Aktualizuj_podobieństwa;}
13.   Wyczyść graf DAC;
14.   Pobierz dokumenty D z odpowiedzi tradycyjnej wyszukiwarki;
15.   Dla każdego dokumentu D{
16.
17.     Dodaj do DAC krawędzie DA=alfa*(1/liczba_autorów_d1);
18.     Dodaj A do zbioru_autorów, gdzie A jest autorem D;
19.
20.     Dla każdej pary dokumentów D {
21.       Dodaj do DAC DD=beta*(wartość wyliczona wg wzoru (13))+DD(Pingback);
22.     }
23.
24.     Dla każdego pojęcia C opisującego D {
25.       Dodaj do DAC DC= gamma*(1/liczba_pojęć_opisujących_D);
26.       Dodaj C do zbioru_pojęć;
27.     }
28.   }
29.   Dodaj do DAC AA(XFN), gdzie oboje A ∈ zbioru_autorów;
30.   Dodaj do DAC CC(Wordnet), gdzie obydwa C ∈ zbioru_pojęć;
31.
32.   MCL(DAC);
33.   Oblicz(spójność i zgodność, wg wzoru (14));
34.   Pobierz miary;
35. }
36.
37. // po stronie klienta (JavaScript)
38. Rerankuj_wyniki_wyszukiwania() {
39.   Grupuj_DAC;
40.   Dla każdej grupy z grup dokumentów {
41.     Sortuj(dokumenty w grupie, wg pierwotnych numerów porządkowych);
42.     Oblicz(rangę grupy, wg wzoru (16));
43.   }
44.   Sortuj(grupy, wg rangi grupy);
45. }
```

Rysunek 5.9: Pseudokod algorytmu rerankingu za pomocą grafu DAC

Graf *D* zawiera krawędzie związku między dokumentami wydobytego z informacji Pingback. Dwa stopnie oddalenia oznaczają 2 linki w ścieżce między dwoma dokumentami. Są to jednak tylko takie ścieżki, w których linki mają przeciwny zwrot. To znaczy nie są modelowane związki wynikające z linku tranzytywnego, dla A i C byłby to np.: A → B → C, tylko wynikające ze ścieżek typu A ← B → C (dokument linkuje do dwóch innych) lub A → B ← C (dwa dokumenty linkują do tego samego).

W przypadku grafu A wagami krawędzi są odwrotności długości ścieżek tworzonych z linków

XFN. Ponieważ związki XFN najczęściej są zwrotne, więc kierunek linków nie ma tu znaczenia. Ograniczenie do 6 linków w ścieżce jest zabiegiem wynikającym z oszczędności zasobów oraz z hipotezy 6 stopni oddalenia (Milgram 1967). Graf C modelowany jest analogicznie do grafu A, jednak wagi krawędzi są odwrotnościami długości ścieżek w ontologii WordNet.



Rysunek 5.10: Schemat blokowy algorytmu konstrukcji grafu DAC

Kolejna funkcja pseudokodu – *Grupuj_DAC()* (linie 10-35) – jest wykonywana na serwerze dla

każdego zapytania. Funkcja ma za zadanie stworzyć graf DAC dla konkretnej kolekcji dokumentów zwróconych przez tradycyjną wyszukiwarkę, a następnie ten graf pogrupować algorytmem MCL.

Linia 12 jest teoretycznym uproszczeniem. Ze względu na zużycie zasobów wymagane przez funkcję *Aktualizuj_podobieństwa()* w praktyce jest ona uruchamiana okresowo, a nie dla każdej zmiany indeksu w momencie tworzenia grafu DAC.

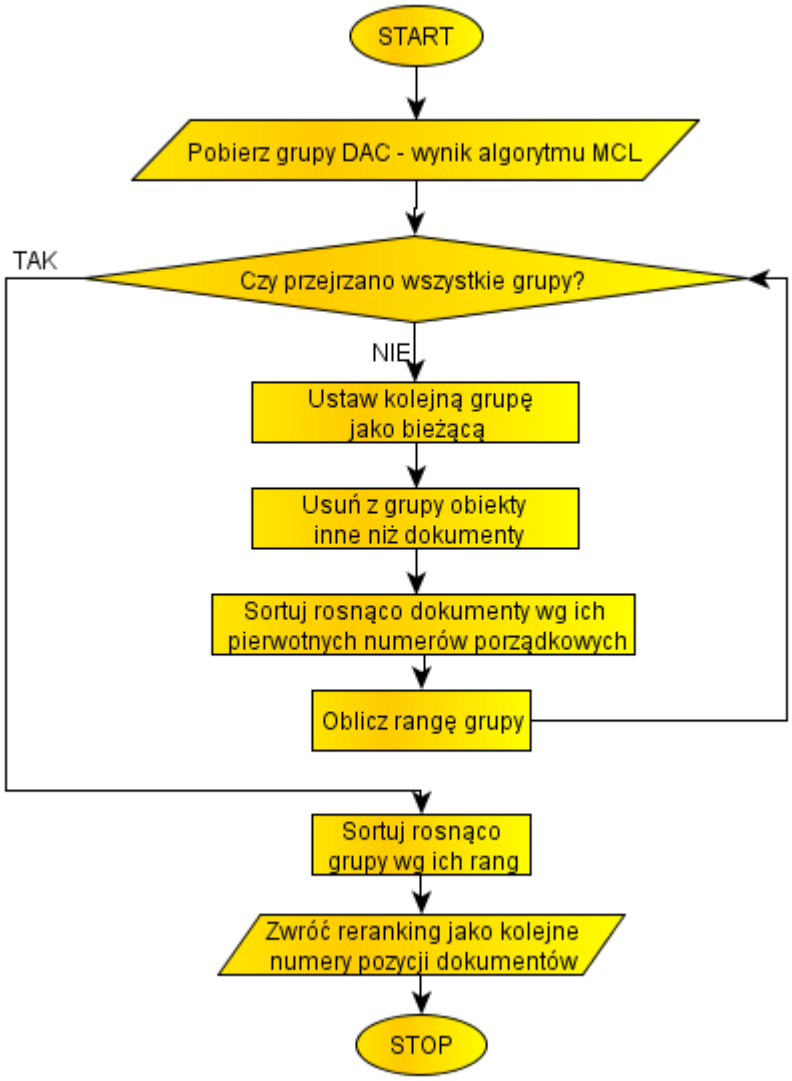
Budowanie grafu odbywa się w pętli iterowanej dla każdego dokumentu z odpowiedzi (linie 15-28). W pętli tej obliczane są wagi krawędzi DAC (linie 17, 21, 25) oraz zapamiętywana informacja o krawędziach do pobrania z grafów homogenicznych A i C (linie 18 i 26), co jest czynione po wykonaniu pętli (linie 29-30). W przypadku grafu *D* informacja jest wykorzystywana na bieżąco do modyfikacji wagi DD (linia 21), ponieważ – w odróżnieniu od krawędzi AA i CC – w pętli *D* nie ma ryzyka dublowania operacji dodawania krawędzi. Parametry alfa, beta i gamma, to stałe wyznaczane empirycznie w celu „wyważenia" stosunków między związkami różnych typów. Przykładowo, ponieważ, w praktyce platformy blogów Wordpress, każdy post ma dokładnie jednego autora, w badaniach alfa ustawiono na jeden, a gamma na średnią liczbę tagów opisujących jeden post, czyli trzy[31].

Ze względu na skrótowość zapisu w pseudokodzie sam algorytm konstrukcji grafu może być nieczytelny stworzono jego alternatywną wersję w postaci schematu blokowego (rysunek 5.10). Schemat blokowy, będący alternatywną wersją linii 13-34 pseudokodu, opisuje algorytm niezależnie od struktur danych – operując na wyłącznie zbiorach. Stad nieco inna semantyka oznaczeń D, A i C w pseudokodzie i w schemacie blokowym: w pseudokodzie D, A i C oznaczają konkretne obiekty, natomiast w schemacie zbiory tych obiektów. Czyli blokowe A odpowiadają pseudokodowemu *zbiorowi_autorów*, a C – *zbiorowi_pojęć*.

Linie 32-34 odpowiadają za uruchomienie algorytmu grupowania dla stworzonego grafu DAC oraz obliczenie i archiwizowanie danych potrzebnych w późniejszej analizie (patrz 6.1 Przyjęte miary s. 98).

Ostatnia funkcja w pseudokodzie – *Rerankuj_wyniki_wyszukiwania()* linie (37-45) – wykorzystująca dwie poprzednie ma za zadanie już po stronie przeglądarki, po przesłaniu wyników, obliczyć rangi w rerankingu i posortować dokumenty zgodnie z wcześniejszym opisem. Ponownie, alternatywnie do pseudokodu, ten fragment algorytmu opisano również schematem blokowym (rysunek 5.11).

31 Trzy to średnia tagów na jeden post po przefiltrowaniu postów ze splogów, które mogą mieć po kilkadziesiąt tagów, przez co znacznie zawyżają średnią



Rysunek 5.11: Schemat blokowy ustalania rerankingu po grupowaniu grafu DAC

6 Weryfikacja metod analizy spójności i zgodności

Przydatności wyników po rerankingu nie da się ocenić w sposób bezwzględny, podobnie jak w przypadku przydatności każdych wyników wyszukiwania. Stworzono co prawda naukowe miary, które pozwalają w jakimś stopniu ocenić relewancję wyników. Jednak, ponieważ oceny te bazują na empirycznych danych uzyskanych dzięki użytkownikom oceniającym wynikowe dokumenty, sam fakt oceny relewancji dokumentu względem pytania może być rzeczą subiektywną. Czasami zakłada się, że oceniającymi relewancję dokumentów są eksperci w dziedzinie, której dotyczą dokumenty, co ma decydować o obiektywności ocen, jednak problemem staje się ewentualna niezgodność ocen ekspertów. Dlatego w pracy zdecydowano się ograniczyć do badania pertynencji, czyli relewancji kognitywnej wyników, która – w odróżnieniu od relewancji tematycznej – nie ma badać zgodności dokumentu z zapytaniem, tylko przydatność dokumentu dla użytkownika. Dzięki temu nie powstaje problem niezgodności „obiektywnych" ocen, ponieważ ocena pertynencji ma być subiektywna.

Wybór pertynencji, zamiast tradycyjnej relewancji tematycznej, do empirycznej oceny rerankingu pozwala również ominąć praktyczny problem związany z językami dokumentów. Użytkownik nie jest w stanie ocenić relewancji dokumentu w języku, którego nie zna. W przypadku blogów na WordPress'ie bardzo popularnymi językami są m.in. indyjski i hiszpański, które nie są specjalnie popularne wśród polskich użytkowników. Napotykając dokument w nieznanym języku, użytkownik podczas testów empirycznych nie jest wstanie ocenić jego relewancji (chyba, że wyłącznie po występowaniu słów kluczowych), natomiast pertynencja automatycznie jest znana i równa zero.

Eksperyment, będący weryfikacją empiryczną rerankingu w oparciu o analizę spójności i zgodności, sprowadza się do następujących kroków:

1. Użytkownik zadaje zapytanie i uzyskuje dokumenty ułożone według rerankingu.
2. Użytkownik zaznacza dokumenty pertynentne.
3. Na podstawie podzbioru zaznaczonych dokumentów obliczane są miary weryfikujące przydatność rerankingu.
4. Dla tych miar przeprowadzane są testy istotności statystycznej.

6.1 Przyjęte miary

Ponieważ reranking poprzez analizę spójności i zgodności ma poprawiać pierwotny ranking, dlatego zaproponowane miary weryfikujące skupiają się na porównaniu tych dwóch rankingów ze względnym rankingiem idealnym. Pierwotny ranking uzyskany z silnika Solr na potrzeby weryfikacji nazywany będzie skrótowo rankingiem Solr. Reranking otrzymany na podstawie analizy spójności i zgodności używając grafu DAC będzie skrótowo nazywany rankingiem DAC. Ranking idealny czy doskonały to ranking stworzony na podstawie rankingu, który badamy i wyboru dokumentów pertynentnych przez użytkownika. Ranking idealny tworzony jest według następującego algorytmu:

1. Weź ranking weryfikowany z zaznaczonymi dokumentami pertynentnymi.
2. Wszystkie dokumenty zaznaczone jako pertynentne przesuń na początek rankingu zachowując ich względną kolejność.
3. Przypisz dokumentom rangi zgodne z ich numerem porządkowym w takim ustawieniu.

Dzięki takiemu zabiegowi ranking idealny zawiera na najwyższych pozycjach dokumenty pertynentne w kolejności w jakiej otrzymał je użytkownik oceniający pertynencję. Niepertynentne dokumenty pozostają w rankingu doskonałym również w tej samej względnej kolejności co pierwotnie, ale najwyżej oceniony dokument niepertynentny w weryfikowanym rankingu jest w rankingu idealnym zawsze niżej niż najgorzej oceniony dokument pertynentny. Z zachowania względnej kolejności dokumentów w rankingu idealnym wynika brak możliwości stworzenia

jednego bezwzględnego rankingu idealnego dla danej odpowiedzi wyszukiwarki. Aby porównanie dwóch rankingów, dla danej kolekcji w odpowiedzi wyszukiwarki, było sprawiedliwe należy dla każdego z nich niezależnie wyznaczyć względny ranking idealny. Na szczęście w praktyce użytkownik nie musi oceniać pertynencji kolekcji wynikowej dwa razy: raz dla jednego rankingu, a potem dla drugiego. Ponieważ kolejność prezentacji dokumentów nie ma wpływu na sam fakt, czy dany dokument jest pertynentny czy nie, wystarczy sam ten fakt zapamiętać. W weryfikacji rerankingu użytkownik zaznacza pertynentne dokumenty podane w kolejności rerankingu i na tej podstawie wyznaczany jest ranking idealny dla rankingu DAC. W tle kolekcja dokumentów razem z zaznaczonymi dokumentami sortowana jest według pierwotnego rankingu i tworzony jest ranking idealny dla rankingu Solr.

Dla tak przygotowanych rankingów najprostszą metodą ich porównania są tradycyjne współczynniki korelacji rankingów. Pierwszym z nich jest współczynnik τ zdefiniowany w Kendall 1938) wyrażany wzorem (17):

$$\tau = \frac{2(n_c - n_d)}{n(n-1)} \tag{17}$$

gdzie:

n_c - liczba par zgodnych (ang. concordant) w porównywanych rankingach,
n_d - liczba par niezgodnych (ang. disconcordant) w porównywanych rankingach,
n - liczba elementów w rankingu.

Zgodność i niezgodność par w pojęciu współczynnika τ dotyczy względnej kolejności dwóch elementów w jednym rankingu w stosunku do względnej kolejności dwóch elementów w drugim rankingu. Para jest zgodna, jeśli różnica rang obu elementów w parze jest dla obu par dodatnia lub dla obu ujemna. Jeśli różnica dla jednej pary jest o znaku przeciwnym niż dla drugiej – pary są niezgodne.

Drugim uznanym współczynnikiem korelacji rankingów jest, jeszcze starszy od τ, współczynnik ρ opisany w Spearman 1904) i wyrażany wzorem (18):

$$\rho = 1 - \frac{6\sum d_i^2}{n(n-1)} \tag{18}$$

gdzie:

$d_i = x_i - y_i$ - różnica dwóch rang elementu w rankingach X_i i Y_i ,
n - liczba elementów w rankingu.

Ze wzorów (17) i (18) widać, że oba współczynniki korelacji liczone są dla wszystkich par rang w obu rankingach, stąd normalizacja jednakowym mianownikiem. Ponieważ ρ używa potęgi przy wyrażaniu odległości między rangami, przy dopełnieniu do jedynki zawsze będzie zwracało wartości większe niż τ, które traktuje odległości między rangami liniowo. Poza tym, oba współczynniki, dzięki jednakowej normalizacji, zwracają podobne wartości z przedziału [-1,1]. Wartość 1 oznacza idealną zgodność rankingów, -1 oznacza, że jeden ranking jest odwróceniem drugiego. Zero oznacza brak korelacji między rankingami.

Dodatkowo, oprócz korelacji rankingów, zaproponowano inne miary, które zgodnie z zaleceniami (Pipino, Lee, i Wang 2002) mają przejrzyste definicje, dotyczą mierzalnych aspektów rankingu i dobrze je prezentują. Tradycyjnymi miarami relewancji wyników są dokładność (ang. *precision*) i kompletność (ang. *recall*) opisane np. w Manning, Raghavan, i Schütze 2008), a pierwszy raz użyte w Kent et al. 1955). Dokładność to stosunek liczby wyszukanych dokumentów relewantnych do wszystkich wyszukanych. Kompletność to stosunek liczby wyszukanych dokumentów relewantnych do wszystkich relewantnych. Ponieważ te stosunki liczbowe są znormalizowane do przedziału [0,1], można je również interpretować jako prawdopodobieństwa warunkowe. Np. prawdopodobieństwo dokładności jest maksymalne i równe 1, gdy liczba

wyszukanych relewantnych dokumentów jest równa liczbie wszystkich wyszukanych. Warto zauważyć, że dokładność=1 to nie jest jeszcze idealna sytuacja. Użytkownik dostaje co prawda same relewantne dokumenty, ale nie ma gwarancji, że są to wszystkie relewantne. O tą część dba kompletność. Idealna sytuacja zachodzi, gdy obie miary: dokładność i kompletność są maksymalne i równe 1. Alternatywnie można używać miary trafności (ang. *accuracy*), która łączy oba aspekty dokładności i kompletności.

Pomimo że dokładność i kompletność mają większy sens, gdy stosuje się je łącznie, w praktyce rzadko można to zrobić. Załóżmy, że miary dokładności i kompletności dla relewancji analogicznie zdefiniujemy dla – badanej w eksperymencie na potrzeby tej pracy – pertynencji. Kompletności nie jesteśmy w stanie wyznaczyć, ponieważ nie mamy informacji o wszystkich pertynentnych dokumentach zaindeksowanych. Użytkownik zaznacza pertynentne dokumenty tylko w ramach wyszukanej kolekcji. Brakuje więc mianownika ze wzoru na kompletność, czyli liczby wszystkich dokumentów pertynentnych dla danego zapytania.

Możemy natomiast wyznaczyć dokładność. Sama dokładność jest niezależna od rankingu. To znaczy: niezależnie w jakiej kolejności będą ułożone dokumenty, dokładność będzie taka sama. Aby wykorzystać dokładność do porównania rankingów przyjęto jej zmodyfikowaną wersję, zwaną *P@n*, czyli dokładnością na *n* pozycjach (ang. *precision at n*). Za *n* przyjęto liczbę pertynentnych dokumentów wskazanych w odpowiedzi przez użytkownika. Tak zdefiniowaną dokładność można wyrazić wzorem wzór (19):

$$P@n = \frac{|pertinent(top(n))|}{n} \tag{19}$$

gdzie:

n - liczba pertynentnych dokumentów w rankingu,

pertinent(top(n)) - zbiór dokumentów pertynentnych na pierwszych n pozycjach rankingu.

P@n może być różna dla różnych rankingów, ale nie musi. Prawdopodobieństwo wychwycenia różnicy przez tą miarę wzrasta wraz z liczbą pertynentnych dokumentów w wynikach wyszukiwania. Jednak, jak widać w wynikach testów (tabela 6.2) wartości tej miary dla dwóch rankingów częściej są jednakowe niż różne. Dlatego postanowiono zaproponować kolejną miarę – modyfikację *P@n* – która dotyczy *n* dokumentów, ale obejmuje wszystkie pertynentne dokumenty, a nie tylko te na pierwszych *n* pozycjach. Jest to kompletność porządkowa wyrażana wzorem (20):

$$R@n = \frac{\sum_{i=1}^{n} i}{\sum_{i=1}^{n} r_i} = \frac{n(n+1)}{2\sum_{i=1}^{n} r_i} \tag{20}$$

gdzie:

n - liczba pertynentnych dokumentów w rankingu,

r_i - nr porządkowy dokumentu w rankingu.

Jak widać z pierwszej części wzoru (20), *R@n* jest sumą pozycji dokumentów pertynentnych w idealnym rankingu w stosunku do rzeczywistych pozycji tych dokumentów w badanym rankingu. Druga część wzoru jest uproszczeniem wykorzystującym znany wzór na sumę ciągu arytmetycznego o pierwszym elemencie równym 1, ostatnim równym *n* i różnicy między kolejnymi elementami ciągu równej 1. W tabeli 6.2 można zauważyć, że *R@n* częściej daje różne wartości dla dwóch rankingów niż jednakowe, czyli wydaje się być bardziej przydatna niż *P@n*.

Ostatnią miarą, którą zaimplementowano w aplikacji wyszukiwarki jest miara uwzględniająca specyfikę rerankingu opartego na grupowaniu. Miarę tę, nazwaną dopasowanie (ang. *fitness*) można wyrazić wzorem (21):

$$F = \frac{\sum_{i=1 \wedge |C_i|>1}^{n} |pertinent(C_i)| + \sum_{i=1 \wedge |C_i|>1 \wedge |pertinent(C_i)|=0}^{n} |C_i|}{\sum_{i=1 \wedge |C_i|>1}^{n} |C_i|} \quad (21)$$

gdzie:

n - liczba pertynentnych dokumentów w rankingu

r_i - nr porządkowy dokumentu w rankingu.

W uproszczeniu można uznać dopasowanie za średnią ważoną stopnia wypełnienia grup dokumentami pertynentnymi lub nie. Miara ta zwraca maksymalną wartość, równą 1, gdy każda z grup wynikowej kolekcji dokumentów zawiera albo same pertynentne dokumenty, albo same niepertynentne. Ponieważ własność dopasowania zawsze zachodzi dla grup jednoelementowych, dlatego miara dopasowania nie bierze ich pod uwagę (warunek $|C_i|>1$). W przypadku, gdy grupa nie jest w pełni dopasowana (pełna dokumentów pertynentnych albo niepertynentnych), pod uwagę brany jest stopień dopasowania, czyli stosunek dokumentów pertynentnych do wszystkich w grupie. Dopasowanie ma na celu sprawdzenie, czy ma sens wizualizacja grup wyników. Aplikacja koloruje jednakowo tła postów z jednej grupy. Jeśli ranking uzyskuje wysokie wartości dopasowania, to oznacza, że jeśli jeden dokument z grupy jest pertynentny, to istnieje duże prawdopodobieństwo, że pozostałe również będą. Podobnie jeśli pierwszy dokument z grupy okazał się niepertynentny, to pozostałe zapewne też takie będą. Ta własność może zostać użyta do skutecznego filtrowania spamu, który w blogosferze stanowi duży problem w postaci splogów[32].

Co ciekawe klasyfikowanie spamu w kategoriach pertynencji wydaje się o wiele lepsze niż w kategoriach relewancji. Spam, podobnie jak pertynencja jest pojęciem subiektywnym, dlatego gdyby mierzyć go relewancją, to może się okazać, że reklama produktu zgodnego z wyszukiwaniem jest relewantna, ale jeśli użytkownika nie interesują reklamy, to na pewno nie będzie pertynentna. W związku z tym odfiltrowanie dokumentów potencjalnie niepertynentnych na podstawie wartości dopasowania będzie zawsze skuteczniejszą walką ze spamem, niż odfiltrowywanie dokumentów nierelewantnych.

6.2 Wyniki eksperymentu i dyskusja

Tabele 6.1, 6.2 i 6.3 zawierają wyniki miar obliczonych przez aplikację w czasie eksperymentów. Eksperyment, przeprowadzony według schematu opisanego na początku rozdziału 6 (s. 98), obejmował ocenę pertynencji wyników wyszukiwarki na zadane przez użytkownika zapytanie. W eksperymentach udział wzięło 17 osób – pracowników i studentów Politechniki Wrocławskiej. Łącznie ocenili oni wyniki 38 zapytań do wyszukiwarki.

Eksperyment przeprowadzono według idei „ślepych testów". To znaczy, że użytkownicy oceniający pertynencję nie znali metody, według której wyniki zostały im zaprezentowane. Progi spójności i zgodności zostały arbitralnie ustawione na odpowiednio ¼ i ⅓. Takie wartości wydawały się najlepiej pasować do charakterystyki indeksowanych danych, czyli blogów. Użytkownicy nie wiedzieli na czym polega analiza spójności i zgodności, co oznaczają pokolorowane grupy dokumentów, ani że oceniają reranking.

Ponieważ reranking opiera się na grupowaniu grafu DAC, odpowiedzi zawierające mniej niż 20 dokumentów nie kwalifikowały się do rerankingu. Po odrzuceniu zapytań zwracających takie odpowiedzi do oceny rerankingu użyto 28 zapytań z ocenionymi odpowiedziami.

32 Splog (ang. *spam blog*) - blog stworzony przez robota internetowego wykorzystywany do spamowania wyników wyszukiwarek internetowych, czyli nieprawdziwego pozycjonowania stron. Zawartość splogów często jest nonsensownym zlepkiem przypadkowych słów i zdań, przeplecionym linkami. Teksty są mechanicznie kopiowane z innych stron i blogów - (Contributors 2008)

pytanie	D	A	C	D-D	D-A	D-C	A-A	C-C
skiing in austria___3_4	69	54	795	0	69	113	3	585
polish food___3_4	82	67	907	3300	82	153	3	904
formula and one___3_4	109	117	783	0	100	219	38	914
e-learning___3_4	41	28	264	0	41	70	3	276
multimedia_video__3_4	24	14	356	0	24	10	2	123
hendrix jimi___3_4	21	20	184	0	21	33	1	46
tsunami___3_4	49	49	627	0	49	79	3	536
u2 music___3_4	38	56	717	0	38	73	19	618
semantic web___3_4	40	46	516	0	40	66	19	325
social AND network_social net	45	40	707	0	45	93	4	494
semantic AND web___3_4	24	31	315	0	24	29	19	139
ajax___3_4	22	31	219	0	22	41	11	115
record AND video___3_4	53	99	1091	0	53	88	46	595
skype___3_4	49	42	747	0	49	106	1	596
garden_flowers__3_4	47	39	521	0	47	87	4	373
pregnancy_baby__3_4	35	53	538	0	35	77	30	458
groove music___3_4	45	46	566	0	45	75	10	562
nobel___3_4	49	68	725	0	49	90	20	411
liverpool___3_4	74	89	949	0	74	108	27	859
god_novel__3_4	46	31	937	0	46	50	1	622
radiohead_radiohead__3_4	29	27	427	0	29	56	5	353
upgrade ubuntu___3_4	140	118	751	0	100	207	23	628
rock_music__0_4	200	145	607	0	100	175	76	5
CSS___0_4	82	108	381	0	82	140	51	1
validator___0_4	81	188	407	0	81	96	110	11
google labs___0_4	850	94	338	0	100	248	8	7
nintendo___3_4	139	184	851	0	100	222	111	784
kung fu___0_4	169	85	421	0	100	230	28	12
średnia	94,7	70,3	594,5	117,9	58,8	108,4	24,1	405,4

Tabela 6.1. Liczby węzłów i krawędzi w grafach DAC konstruowanych dla kolekcji testowych w weryfikacji empirycznej

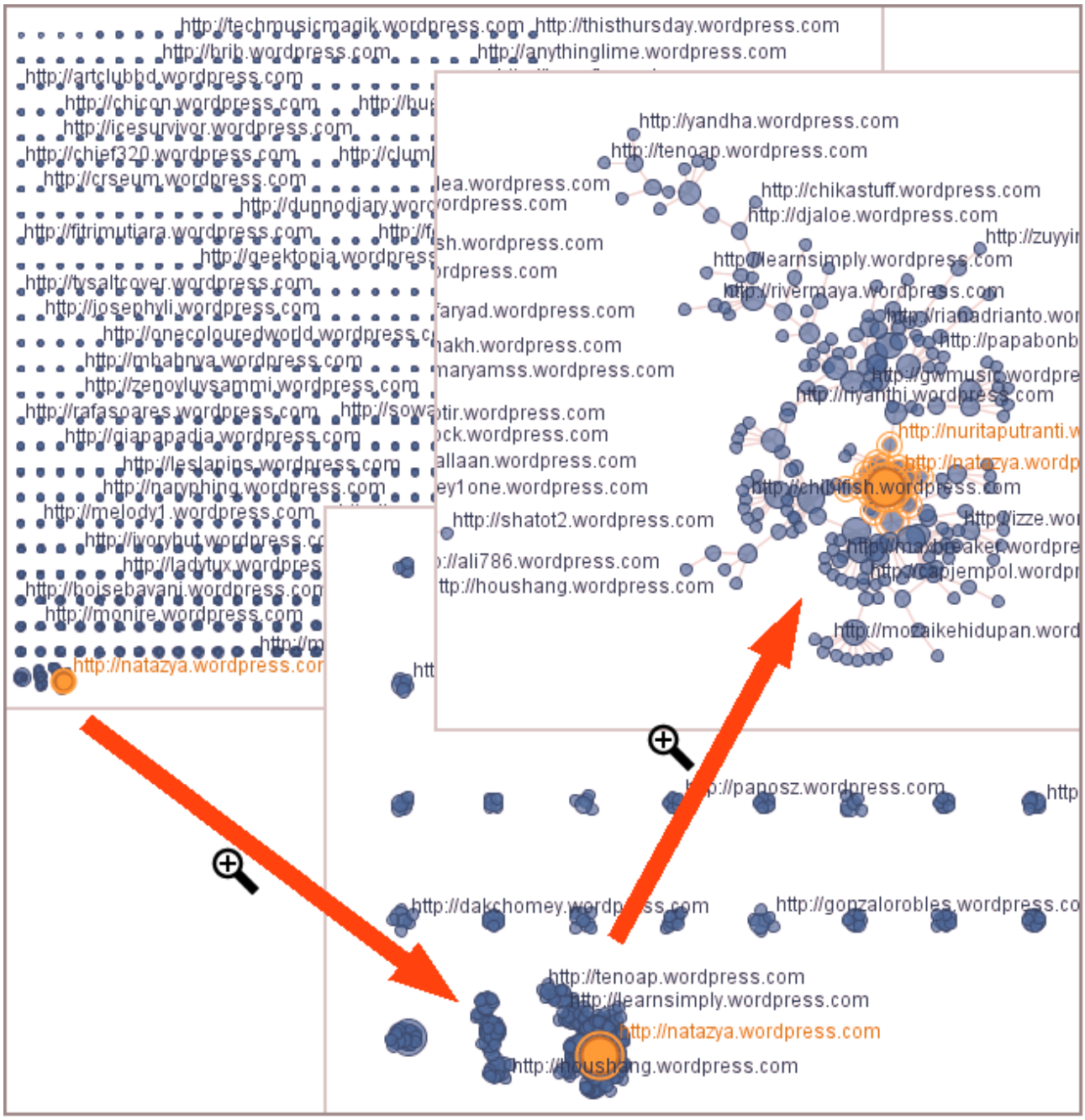
Jak pokazuje tabela 6.1, dla zadawanych pytań wyszukiwarka zwracała średnio kolekcję poniżej 100 dokumentów. W praktyce, przy analizie spójności i zgodności, pod uwagę brane było tylko pierwsze 100 dokumentów. Jeśli więc założyć, że na każde zapytanie wyszukiwarka zwróciła maksymalnie 100 dokumentów, to średnia zmniejsza się do 58,8. Średnie liczby autorów i tagów, jako węzłów w grupowanych grafach DAC, tworzonych niezależnie dla każdego zapytania, to odpowiednio: 70,3 i 594,5. Przy założeniu, że każdy post może mieć jednego autora, może dziwić fakt, że średnia liczba autorów jest większa od średniej liczby dokumentów poddanych analizie. Należy jednak pamiętać, że ze względu na niski próg spójności, średnia nie dotyczy wyłącznie autorów wyszukanych dokumentów, ale również tych o wystarczająco silnych związkach z nimi. Podobna sytuacja dotyczy tagów, jednak tu jest to trudniej zauważalne, ponieważ jeden post może mieć wiele tagów. Średnia tagów na jednego posta dla całej bazy indeksowej to 5,3081. Widać więc, że podczas analizy brano pod uwagę ponad dwa razy więcej tagów niż było przypisanych do postów i to tylko przy założeniu, że każdy post miał przypisane inne tagi, co w praktyce jest bardzo rzadkie.

pytanie	zgodność	spójność	dopasowanie – F	dokładność P	P@n – S	P@n – D	R@n – S	R@n – D
skiing in austria___3_4	0,0019	0,0096	1	0,01	0	0	0,5	0,5
polish food___3_4	0,0089	0,3070	0,01	0,01	0	0	0,14	0,14
formula and one___3_4	0,0029	0,0054	1	0,19	0,32	0,37	0,24	0,24
e-learning___3_4	0,0075	0,0188	1	0,12	0,2	0,2	0,1	0,1
multimedia_video__3_4	0,0018	0,0370	0,94	0,04	1	1	1	1
hendrix jimi___3_4	0,0038	0,0268	1	0,24	0,2	0,2	0,36	0,36
tsunami___3_4	0,0027	0,0109	1	0,12	0,17	0,17	0,26	0,28
u2 music___3_4	0,0024	0,0130	1	0,13	0,8	0,8	0,47	0,5
semantic web___3_4	0,0025	0,0161	1	0,18	0,71	0,71	0,58	0,7
social AND network_social netv	0,0021	0,0137	0,89	0,2	0,78	0,67	0,79	0,7
semantic AND web___3_4	0,0029	0,0290	0,67	0,38	0,56	0,56	0,56	0,51
ajax___3_4	0,0054	0,0239	0,75	0,32	0,29	0,43	0,46	0,44
record AND video___3_4	0,0010	0,0086	1	0,15	0,13	0,13	0,26	0,27
skype___3_4	0,0022	0,0122	0,73	0,14	0,29	0,29	0,19	0,18
garden_flowers__3_4	0,0029	0,0140	0,9	0,47	0,64	0,64	0,59	0,59
pregnancy_baby__3_4	0,0033	0,0170	1	0,54	0,58	0,58	0,57	0,57
groove music___3_4	0,0034	0,0134	0,87	0,11	0,4	0,2	0,41	0,43
nobel___3_4	0,0017	0,0102	1	0,08	0,5	0,5	0,53	0,53
liverpool___3_4	0,0018	0,0076	0,86	0,35	0,65	0,73	0,68	0,71
god_novel__3_4	0,0014	0,0161	0,73	0,02	0	0	0,08	0,05
radiohead_radiohead__3_4	0,0039	0,0221	0,8	0,24	0,57	0,57	0,54	0,57
upgrade ubuntu___3_4	0,0021	0,0037	0,97	0,07	0,43	0,43	0,47	0,51
rock_music__0_4	0,0006	0,0030	1	0,15	0,47	0,6	0,5	0,66
CSS___0_4	0,0013	0,0074	1	0,05	0,25	0,25	0,18	0,2
validator___0_4	0,0009	0,0053	1	0,06	0	0,2	0,15	0,22
google labs___0_4	0,0004	0,0002	1	0,1	0,4	0,6	0,42	0,51
nintendo___3_4	0,0021	0,0041	0,97	0,07	0,57	0,71	0,45	0,7
kung fu___0_4	0,0014	0,0040	1	0,07	0,71	0,86	0,68	0,68
średnia	0,003	0,024	0,896	0,165	0,414	0,442	0,434	0,459
test znaku					0,109		0,189	
test Wilcoxona dla par					0,141		0,046	
test Kołmogorowa-Smirnowa					1,66E-06	1,66E-06	2,83E-07	5,63E-07
test t-Studenta					0,048		0,027	

Tabela 6.2. Miary spójności i zgodności oraz dopasowania, dokładności i kompletności rankingów dla kolekcji testowej w weryfikacji empirycznej

(wykresy 1, 2, 3 i 4 w dodatku E s. 136 oraz 6.2)

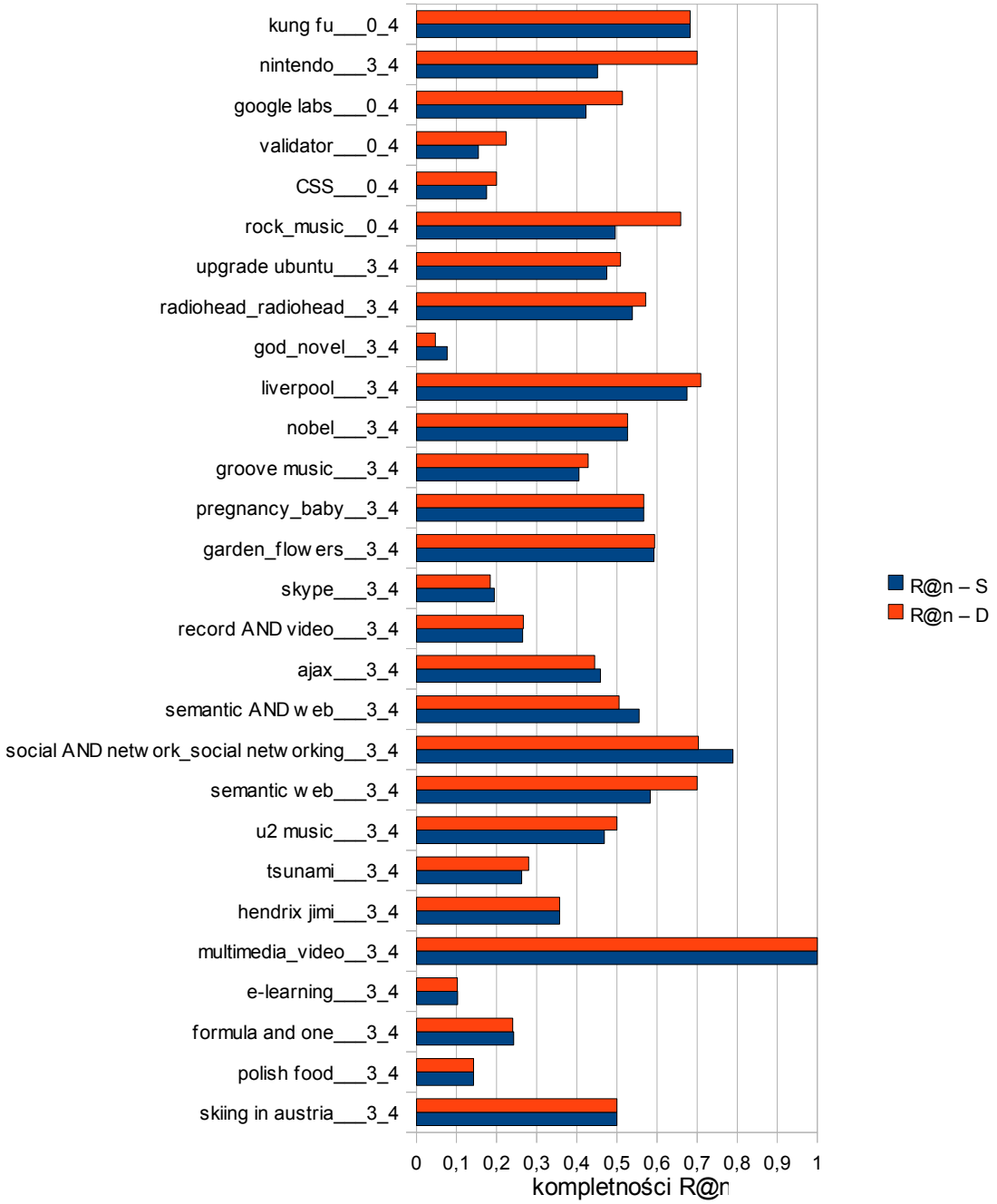
Charakterystyczną kolumną, zawierającą prawie same zera w tabeli 6.1 jest kolumna liczby krawędzi incydentnych z dwoma węzłami typu D. Zera te wskazują, że jak podkreślał Tim Berners-Lee w Tim Berners-Lee 2007a) linkowanie między dokumentami wciąż nie jest tak popularne jak by się mogło wydawać. Między postami nie było linków, a siła związku między postami wynikająca z rankingu Solr nie przekraczała progu istotności, więc związki D-D nie były modelowane w DAC. Jedyna wartość niezerowa wynika z faktu, że na pytanie „polish food" Solr zwrócił kolekcję jednakowo ocenionych dokumentów, co z kolei wpłynęło na siły związków między postami w DAC tak, że przekroczyły one próg istotności. Również związki między autorami okazały się dosyć rzadkie. W całej zaindeksowanej części Wordpress'a użytkownicy deklarujący związki za pomocą linków XFN w blogroll'ach tworzyli graf jak na rysunku 6.1. Jak widać graf jest niespójny i tworzą go głównie kilku węzłowe podgrafy. Tylko kilka ze spójnych podgrafów składa się z więcej niż kilku węzłów.



Rysunek 6.1. Graf użytkowników i związków XFN w zaindeksowanej części serwisu wordpress.com

(wygenerowane w serwisie manyeyes.alphaworks.ibm.com)

Jak widać w pierwszych kolumnach tabeli 6.2 spójność i zgodność wynikowych kolekcji utrzymywały się na poziomie kilku procent czy nawet promili. Średnia zgodność kolekcji to 0,002, natomiast spójność – 0,024, a po odrzuceniu jednej wyjątkowo dużej wartości – 0,13. Na wykresie wykresie 1 w dodatku E s. 136 można zauważyć, że zgodność jest w każdym, poza jednym przypadkiem, mniejsza od spójności pomimo narzucenia wyższego progu dla zgodności niż dla spójności. To wskazuje na celowość wstępnego zróżnicowania progów istotności spójności i zgodności dla ocenianych kolekcji.



Rysunek 6.2. Wykres porównujący kompletność porządkową R@n dla rankingów Solr i DAC z tabeli 6.2 s. 103

Miara dopasowania (wykres 2 w dodatku E s. 137) dała raczej zadowalający wynik 89,6%, jednak należy zastanowić się nad jej statystyczną istotnością w tym przypadku. Ze względu na dużą ziarnistość grupowania w wynikowych kolekcjach rzadko występowała więcej niż jedna grupa większa niż dwa posty. Dwuelementowe grupy też zdarzały się maksymalnie kilka razy w całej kolekcji wynikowej. Wobec powyższego zasadne wydaje się dodatkowo ograniczyć używanie miary dopasowania tylko do przypadków, gdy liczba grup wieloelementowych przekracza określony próg lub uzależnić tę miarę od stosunku grup wieloelementowych do wszystkich grup w wynikach.

Kolejną miarą jest dokładność P (wykres 3 dodatku E s. 138), która jednak nie mierzy jakości

metody rerankingu, tylko samego silnika Solr. Jak zaznaczono wcześniej dokładność P jest niezależna od kolejności prezentowanych wyników. Wartość tej miary na poziomie 0,165 pokazuje zasadność używania rankingu: skoro średnio tylko co siódmy dokument w odpowiedzi jest pertynentny, to warto je pokazać na najwyższych pozycjach. Jak widać z kolejnych dwóch miar – dokładności *P@n* (wykres 4 dodatku E s. 139) i kompletności porządkowej *R@n* (wykres 6.2) - dla rankingów Solr i DAC, rankingi poprawiają dokładność i kompletność. Choć nie można tych miar porównywać bezpośrednio, to w przybliżeniu można powiedzieć, że przy użyciu rankingów dokładność jest 2 razy lepsza. Widać też, że dla tych dwóch miar dokładności w porównaniu ranking Solr : ranking DAC w obu przypadkach wygrywa ranking DAC, mając o około 3% lepszy wynik.

Aby pokazać, że te 3% jest statystycznie istotne przeprowadzono testowanie hipotezy zerowej dla średnich dokładności i kompletności porządkowych obu rankingów. Tabela 6.2 zawiera również wyniki 4 takich testów: 3 nieparametrycznych i parametrycznego. Pierwsze dwa to test znaku i test lewostronny Wilcoxona dla par obserwacji. O ile p-wartości dla testu znaku nie pozwalają odrzucić hipotezy zerowej, o tyle test Wilcoxona dla kompletności rankingów Solr i DAC zwrócił p-wartość mniejszą od 0,05. Oznacza to, że są podstawy do odrzucenia hipotezy zerowej.

Kolejny nieparametryczny test przeprowadzono po to, by stwierdzić, czy można wykonać test parametryczny. Ponieważ testy parametryczne wymagają rozkładu normalnego badanych próbek. Test Kołmogorowa-Smirnowa ma sprawdzić czy badane wartości *P@n* i *R@n* mają rozkład normalny. P-wartości testów dla obu miar rzędu 10^{-6} potwierdziły, że parametryczny test T-studenta może zostać przeprowadzony. Wyniki testu T-studenta dla obu miar zwróciły p-wartości poniżej 5%, co dało podstawę do odrzucenia hipotezy zerowej o równości średnich dla rankingów Solr i DAC. Oznacza to, że zarówno dokładność i kompletność odpowiedzi wyszukiwarki po rerankingu jest statycznie lepsza niż przed rerankingiem, a parametryczny test T-studenta potwierdza istotność statystyczną wyników eksperymentu.

Jeśli wziąć po uwagę współczynniki korelacji rankingów z tabeli 6.3 widać, że Solr i DAC układają dokumenty w podobnej kolejności. Współczynnik korelacji τ Kendalla dla tych rankingów na podstawie 28 testowych przypadków wynosi 0,893, natomiast ρ Spearmana – 0,956. Oznacza to, że przy rerankowaniu względna kolejność postów nie zmieniała się. Jest to zgodne z wcześniejszą uwagą, że grupy często były jednoelementowe. W skrajnym przypadku, gdy nie ma grup większych niż jednoelementowe – oba rankingi, przed i po rerankowaniu, mają jednakową kolejność dokumentów. Dzieje się tak ponieważ ranga jednoelementowej grupy jest równa pierwotnej randze jedynego dokumentu w tej grupie. Wartości współczynników, widoczne na wykresach 6.3 i 5 w dodatku E s. 108, pokazują, że korelacja tych rankingów z rankingiem idealnym jest często jeszcze wyższa niż ich wzajemna. Wynika to z względności rankingu idealnego. Jak opisano wcześniej ranking idealny tworzony jest na podstawie rankingu, z którym ma być porównywany.

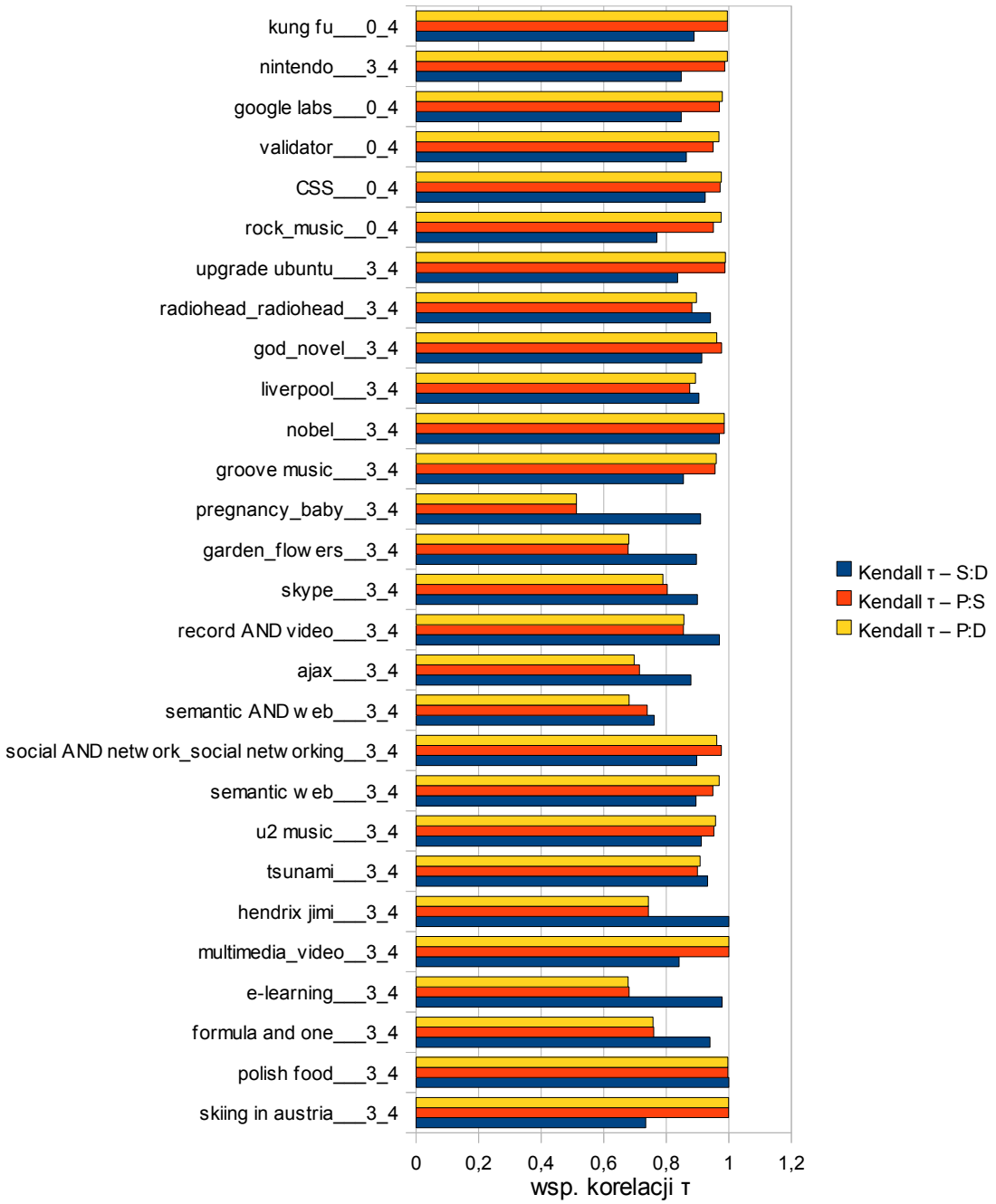
Same wartości τ i ρ są średnio o około 1‰ lepsze dla, odpowiednio, DAC i Solr. Oznacza to, że wg współczynnika Kendalla reranking przybliża kolejność dokumentów w odpowiedzi do kolejności idealnej (najpierw pertynentne). Natomiast według współczynnika Spearmana grupowanie i ponowne rankowanie wyników pogarsza sprawę. Różnice te są jednak tak niewielkie, że żaden z testów nie potwierdził ich istotności statystycznej. Żaden z testów nie zwrócił p-wartości mniejszej niż 0,1. Poza testem na rozkład normalny, którego wynik pozwolił przeprowadzić test T-studenta. W związku z tym, w przeciwieństwie do testów przeprowadzonych dla dokładności i kompletności porządkowej, nie ma podstaw do odrzucenia hipotezy zerowej. To znaczy, że badane próbki nie pozwalają stwierdzić, że średnie współczynników korelacji dla rankingów Solr i DAC w ogólności będą różne. Jednak brak podstaw do odrzucenia hipotezy zerowej nie oznacza jednocześnie, że reranking nie poprawia sytuacji z kolejnością dokumentów pertynentnych w odpowiedzi wyszukiwarki. Po protu te testy tego nie potwierdzają, ale też nie zaprzeczają.

pytanie	Kendall τ – S:D	Kendall τ – P:S	Kendall τ – P:D	Spearman ρ – S:D	Spearman ρ – P:S	Spearman ρ – P:D
skiing in austria___3_4	0,73	1	1	0,85	1	1
polish food___3_4	1	1	1	1	1	1
formula and one___3_4	0,94	0,76	0,76	0,98	0,79	0,78
e-learning___3_4	0,98	0,68	0,68	1	0,59	0,59
multimedia_video__3_4	0,84	1	1	0,94	1	1
hendrix jimi___3_4	1	0,74	0,74	1	0,81	0,81
tsunami___3_4	0,93	0,9	0,91	0,97	0,95	0,96
u2 music___3_4	0,91	0,95	0,96	0,97	0,97	0,97
semantic web___3_4	0,89	0,95	0,97	0,97	0,98	0,99
social AND network_social netv	0,9	0,98	0,96	0,95	0,99	0,99
semantic AND web___3_4	0,76	0,74	0,68	0,9	0,82	0,74
ajax___3_4	0,88	0,71	0,7	0,95	0,79	0,78
record AND video___3_4	0,97	0,85	0,86	0,99	0,91	0,91
skype___3_4	0,9	0,8	0,79	0,96	0,82	0,8
garden_flowers__3_4	0,9	0,68	0,68	0,97	0,74	0,74
pregnancy_baby__3_4	0,91	0,51	0,51	0,96	0,53	0,53
groove music___3_4	0,85	0,96	0,96	0,91	0,99	0,99
nobel___3_4	0,97	0,98	0,98	0,99	1	1
liverpool___3_4	0,9	0,87	0,89	0,96	0,93	0,95
god_novel__3_4	0,91	0,98	0,96	0,98	0,99	0,97
radiohead_radiohead__3_4	0,94	0,88	0,9	0,99	0,94	0,95
upgrade ubuntu___3_4	0,84	0,99	0,99	0,93	1	1
rock_music__0_4	0,77	0,95	0,97	0,9	0,98	0,99
CSS___0_4	0,92	0,97	0,98	0,98	0,99	0,99
validator___0_4	0,86	0,95	0,97	0,94	0,98	0,99
google labs___0_4	0,85	0,97	0,98	0,94	0,99	1
nintendo___3_4	0,85	0,99	1	0,94	1	1
kung fu___0_4	0,89	0,99	0,99	0,95	1	1
średnia	0,8929	0,8836	0,8842	0,9562	0,9098	0,9076
test znaku		0,189			0,523	
test Wilcoxona dla par		0,305			0,808	
test Kołmogorowa-Smirnowa		7,13E-13	7,13E-13		2,02E-12	2,16E-12
test t-Studenta		0,413			0,229	

Tabela 6.3. Współczynniki korelacji rankingów dla kolekcji testowych w weryfikacji empirycznej

(wykresy 6.3 i 5 w dodatku E s. 140)

Skoro testy istotności statystycznej potwierdziły, że wyniki po rerankingu zawsze mają większą dokładności i relewancję porządkową, więc testy współczynników τ i ρ również powinny potwierdzić zasadność stosowania ponownego rankowania. Fakt, że się tak nie stało, wynika najpewniej z mniejszego odchylenia standardowego wartości dla tych współczynników niż dla *P@n* i *R@n*.



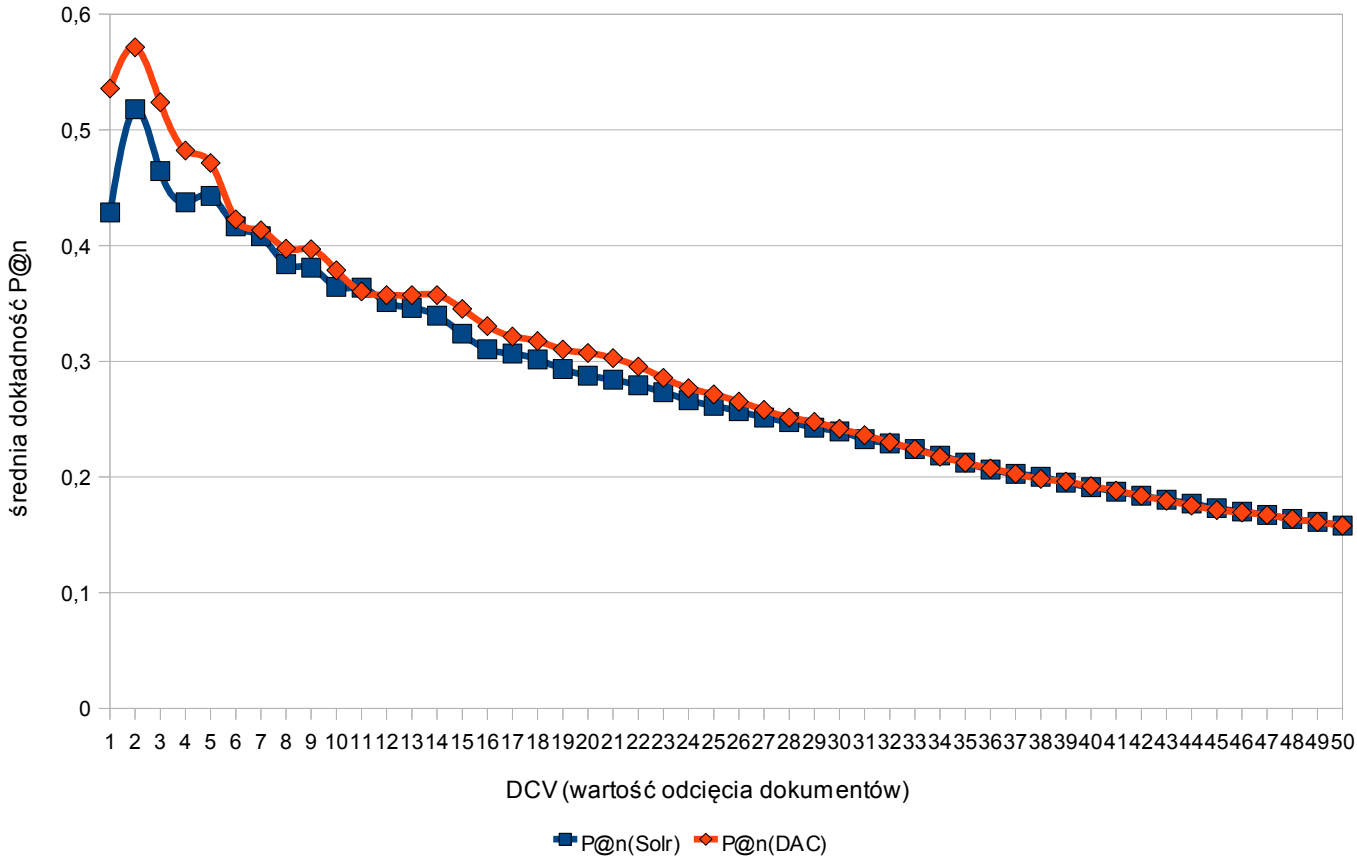
Rysunek 6.3. Wykres porównujący współczynniki korelacji Kendalla w tabeli 6.3 s. 107

6.2.1 Wizualizacja miar za pomocą grafów DCV

Jako dodatkową wizualizację dokładności i kompletności wyników z rankingami Solr i DAC wykonano wykresy wzorowane tych z pracy (Gordon i Pathak 1999). W przytoczonej pracy porównywano efektywność ośmiu wyszukiwarek, a konkretnie ich dokładność i kompletność. Miary te nie były liczone raz dla każdej odpowiedzi wyszukiwarki, ale dla całego przedziału DCV.

DCV (ang. *document cut-off value*), czyli „wartość odcięcia dokumentów", to numer dokumentu, powyżej którego dokumenty nie są brane pod uwagę przy liczeniu dokładności i kompletności. Czyli np. dla DCV=5 miary liczone są dla kolekcji 5 dokumentów – pozostałe dokumenty są „odcinane". W przytoczonej pracy wykresy tworzone były dla skali DCV: 1-20, ponieważ

w badaniach oceniana była relewancja pierwszych 20 dokumentów zwracanych przez wyszukiwarki. W przypadku badania rankingu DAC pod uwagę brane było pierwsze 100 dokumentów. Jednak, po obcięciu odpowiedzi >100 dokumentów, średnia liczba dokumentów w odpowiedzi badanych zapytań to 58,8. Dlatego wykresy 6.4 i 6.5 na osi DVC mają wartości z przedziału <1, 50>.



Rysunek 6.4: Średnie dokładności P@n rankingów Solr i DAC dla różnych wartość odcięcia dokumentów

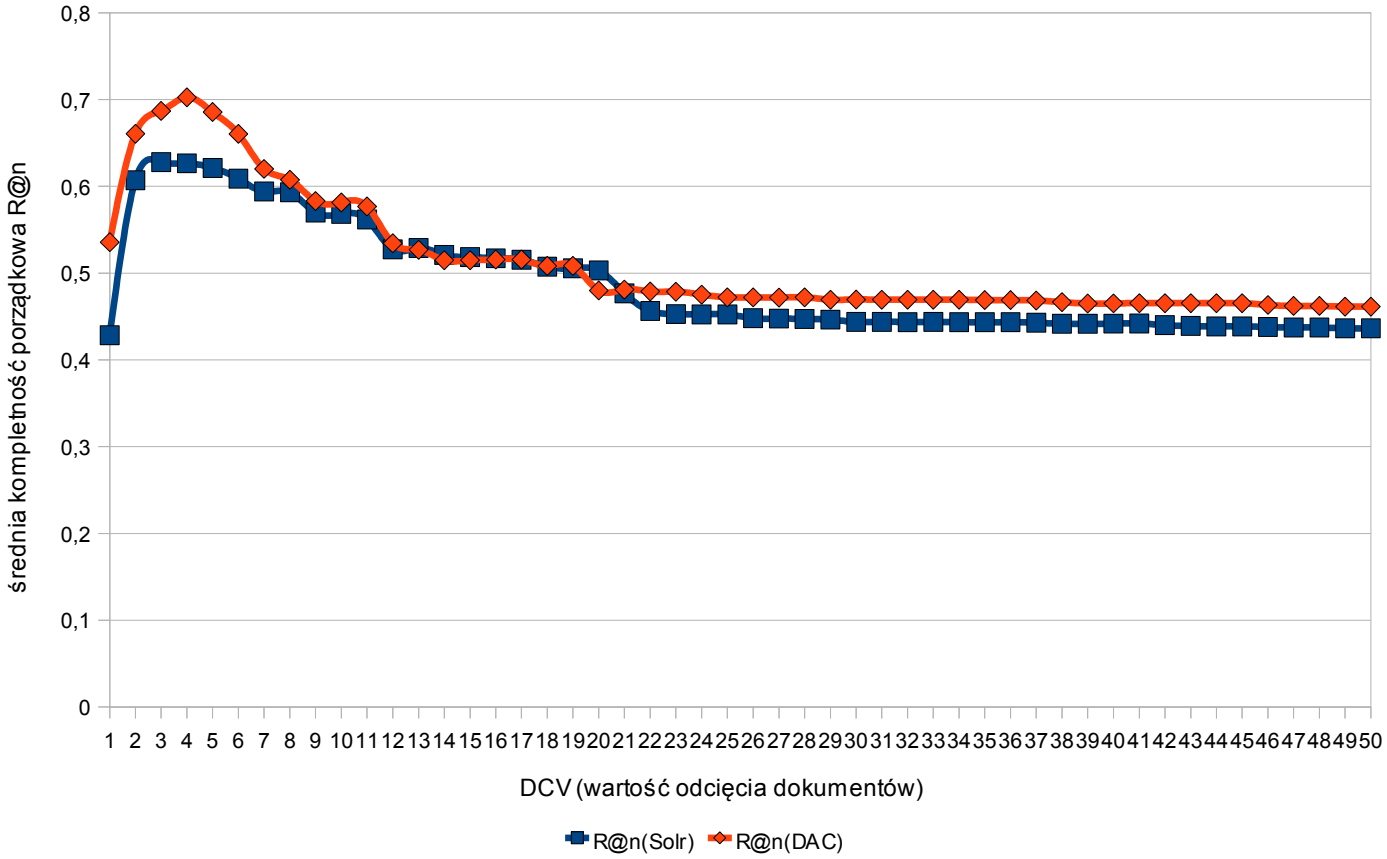
Wykres 6.4 prezentuje średnie dokładności *P@n* wyciągnięte z 28 zapytań tak jak w tabeli 6.2, z tym, że *n* w tych *dokładnościach na n pozycjach* nie jest jak wcześniej stałe i równe liczbie pertynentnych dokumentów w odpowiedzi na zapytanie. Tutaj *n* przyjmuje wartości od 1 do 50. Stąd zamiast jednej średniej dokładności *P@n* dla każdego z rankingów mamy ich po 50. Widać więc, że DCV w przypadku *P@n* oznacza zmienne *n*.

Jak wykazano wcześniej dokładność obu rankingów Solr i DAC w ogólności jest jednakowa, ponieważ oba bazują na tej samej kolekcji dokumentów w odpowiedzi. Stąd potrzeba wprowadzenia *dokładności na n pozycjach*. Skoro więc oba rankingi mają jednakową dokładność, to ich *P@n* przy n dążącym do nieskończoności (czy raczej do liczby dokumentów w kolekcji) jest jednakowa. Tą tendencję widać dokładnie na powyższym wykresie.

Dla użytkownika ważne jest natomiast, żeby znaleźć poszukiwany dokument jak najszybciej, a więc najważniejsze są dokładności na pierwszych kilku pozycjach. Jak widać na wykresie 6.4, pomimo że średnia *P@n*(DAC) dla niektórych *n* jest mniejsza od średniej *P@n*(Solr), to na pierwszych pozycjach jest zawsze znacząco wyższa.

Wykres 6.5 jest analogiczny do poprzedniego, ale obrazuje średnie kompletności porządkowe *R@n* dla obu rankingów. Podobnie jak w przypadku poprzedniego wykresu widać wyraźnie zauważoną wcześniej własność *R@n*, mówiącą, że w odróżnieniu od *P@n*, przy *n* dążącym do nieskończoności kompletności porządkowe obu rankingów będą miały stałą różnicę – niekoniecznie

równą 0.



Rysunek 6.5: Średnie kompletności porządkowe R@n rankingów Solr i DAC dla różnych wartość odcięcia dokumentów

Jak widać na wykresie badane kolekcje wynikowe zapytań miały średnią kompletność R@n w rankingu DAC lepszą niż R@n(Solr), dla całego przedziału DCV, poza drugą dziesiątką dokumentów w odpowiedzi.

6.2.2 Empiryczne wyznaczanie parametrów grupowania DAC

Dwa główne parametry grupowania DAC, a więc i samego rerankingu to próg istotności dla krawędzi grafu i parametr algorytmu MCL – *inflation*. Ponieważ parametry te należało dobrać empirycznie, przeprowadzono dodatkowe badania wcześniejszych miar w różnymi kombinacjami tych 2 parametrów. W tym celu zasymulowano użytkownika i przeliczono dokładność, kompletność i współczynniki korelacji dla tych kombinacji.

Zbadane zostały progi istotności z przedziału <0,3; 0,9> z krokiem 0,2. Wartości mniejsze powodowały znaczny wzrost wielkości grafu DAC i czas działania stawał się niedopuszczanie długi dla pracy online. Parametr *inflation* testowano przedziale <1,2; 4,4> z krokiem 0,4.

Do wyboru najlepszych parametrów posłużono się deltami średnich: $\Delta(\overline{P@n})$, $\Delta(\overline{R@n})$, $\Delta(\overline{Kendall})$, $\Delta(\overline{Spearman})$ liczonych analogicznie do wzoru (22):

$$\Delta(\overline{P@n}) = avg(P@n(DAC)) - avg(P@n(Solr)) \quad (22)$$

gdzie:

avg() - średnia,

P@n(ranking) -dokładność liczona wg wzoru (19).

Każda średnia jest średnią 28 wartości (dokładności, kompletności porządkowej, współczynnika

korelacji Kendalla rankingu z rankingiem idealnym oraz analogicznego współczynnika korelacji Spearmana) odpowiadających 28 zapytaniom z tabel 6.1 - 6.3.

infl	Δ (P@n)	Δ (R@n)	Δ (τ)	Δ(ρ)
1,2	-0,11140	-0,08295	-0,02143	-0,01715
1,6	0,02528	0,01514	0,00312	0,00266
2	0,02192	0,00898	-0,00376	-0,00559
2,4	0,01086	0,00322	-0,00449	-0,00629
2,8	0,00780	-0,00235	-0,00910	-0,01144
3,2	-0,00240	-0,00136	-0,00723	-0,00933
3,6	-0,00240	-0,00184	-0,00741	-0,00946
4	-0,00955	-0,00625	-0,00823	-0,00987
4,4	-0,01653	-0,00155	-0,00926	-0,01094
1,2	-0,07781	-0,05612	-0,01653	-0,01416
1,6	0,03773	0,01482	0,00294	0,00107
2	0,02787	0,02489	0,00065	-0,00223
2,4	0,01920	0,01661	-0,00062	-0,00307
2,8	0,01886	0,01818	0,00169	-0,00114
3,2	0,02243	0,01565	0,00175	-0,00081
3,6	0,02243	0,01475	0,00105	-0,00170
4	0,01290	0,00927	0,00028	-0,00216
4,4	0,01102	0,01240	0,00090	-0,00120
1,2	0,00989	-0,01816	-0,00723	-0,00815
1,6	0,01876	-0,00632	-0,00201	-0,00268
2	0,02591	0,02214	0,00430	0,00107
2,4	0,03101	0,02478	0,00335	0,00009
2,8	0,04019	0,02642	0,00599	0,00231
3,2	0,04019	0,02412	0,00610	0,00243
3,6	0,03622	0,02290	0,00545	0,00201
4	0,02908	0,01843	0,00488	0,00166
4,4	0,02908	0,02719	0,00533	0,00237
1,2	0,02228	-0,00642	-0,00675	-0,00878
1,6	0,02277	-0,00487	-0,00252	-0,00349
2	0,03736	0,01757	0,00239	-0,00022
2,4	0,03736	0,01594	0,00209	-0,00062
2,8	0,02715	0,01793	0,00310	0,00003
3,2	0,02715	0,01448	0,00283	-0,00013
3,6	0,02715	0,01372	0,00270	-0,00020
4	0,02001	0,00972	0,00178	-0,00080
4,4	0,01763	0,01712	0,00199	-0,00026

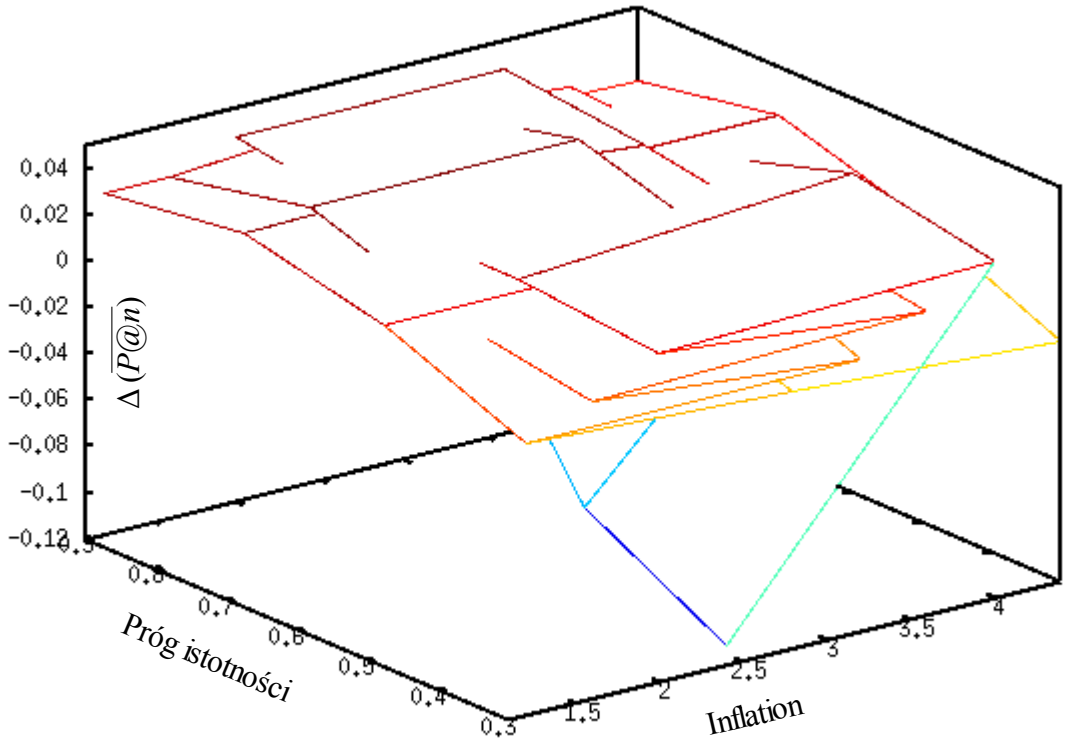
Tabela 6.4. Wartości różnic średnich miar dla rankingów badanych zapytań w zależności od przyjętych parametrów grupowania DAC: progu istotności i inflation; zielone pola oznaczają wartości maksymalne w kolumnie, czerwone – minimalne

(wykresy 6.6 i 1, 2, 3 w dodatku D s. 133 oraz 6.7 i 6.8)

Tabela 6.4 przedstawia delty średnich dla różnych kombinacji parametrów grupowania DAC. Wyróżnione zostały wartości największe i najmniejsze. Jak widać intuicyjnie przyjęte wartości parametrów: progu istotności=0,5 i *inflation*=2 dają całkiem dobre wyniki. W przypadku kompletności *R@n* delta średnich dla tych parametrów jest trzecia co do wielkości, zaraz po parach parametrów (0,7; 4,4) i (0,7; 2,8). W przypadku dokładności *P@n* delta dla (0,5; 2) jest siódma do co wielkości, ale za to drugą jest delta dla (0,5; 1,6). Gorzej jest w przypadku współczynników τ i ρ. Dla wartości delty średnich obu współczynników dla parametrów (0,5; 2) mieszczą się w połowie

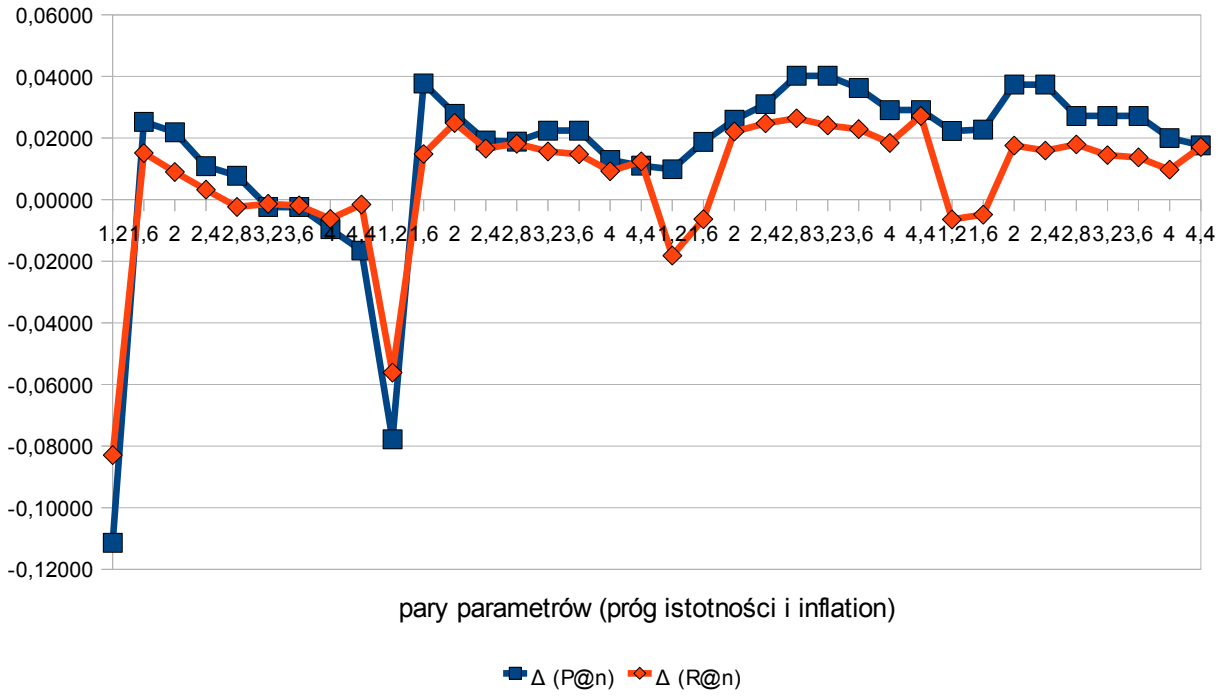
przedziału wszystkich delt.

Wartości funkcję dwóch zmiennych najlepiej wizualizować na wykresie w przestrzeni. Takim wykresem dla delt średnich dokładności *P@n* jest wykres 6.6. Ponieważ delty jako wartości funkcji progu istotności i *inflation* nie są monotoniczne – wykres w przestrzeni jest nieczytelny. Podobnie dzieje jest w przypadku pozostałych, badanych delt. Ich przestrzenne wykresy znajdują się się w dodatku D s. 133.

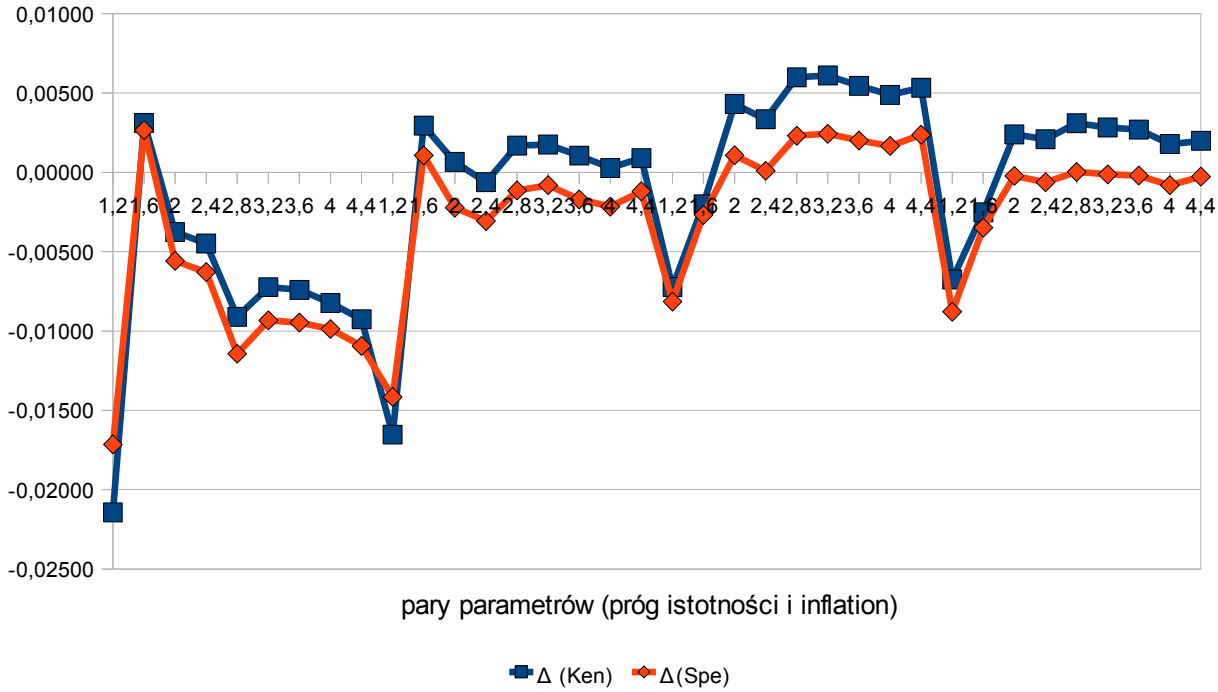


Rysunek 6.6: Wykres zależności różnicy średnich dokładności P@n od przyjętych parametrów grupowania DAC: progu istotności i inflation (na podstawie tabeli 6.4)

Aby znaleźć i przedyskutować wartości parametrów, dla których wartości delt są największe przedstawiono je na wykresach płaskich 6.7 i 6.8. Wykresy te są okresowe: na osi *x* wartości *inflation* powtarzają się 4 razy – dla każdej wartości progu istotności: odpowiednio 0,3; 0,5; 0,7 i 0,9. Również na tych wykresach widać, że wartości delt będąc funkcjami parametrów grupowania nie są monotoniczne. Nie jest również tak, że wszystkie delty są maksymalne dla jednej pary parametrów.



Rysunek 6.7: Delty średnich dokładności i kompletności porządkowych dla par parametrów grupowania DAC



Rysunek 6.8: Delty średnich współczynników korelacji Kendalla i Spearmana dla par parametrów grupowania DAC

W przypadku delt Kendalla widać wyraźnie, że najwyższe wartości osiągane są dla progu istotności = 0,7, niezależnie od *inflation*. W przypadku delt Spearmana jest jest to już tak wyraźne, ponieważ wysoko wybijają się też pary (0,3; 1,6) i (0,5; 1,6). W pobliżu *inflation* 1,6 znajdują się też największe amplitudy na wykresie, ponieważ dla 1,2 wartości delt są lokalnie najmniejsze.

Podobnie, choć mniej wyraźnie, wygląda to w przypadku *P@n* i *R@n*.

Ostatecznie, skoro nie można wybrać jednej pary, dla której wszystkie cztery typy delt są maksymalne, można spróbować wyznaczyć przedziały parametrów grupowania DAC, dających najlepsze wyniki. Próg istotności z pewnością najlepiej przyjąć 0,7, *inflation* w parze z takim progiem daje najczęściej dobre wyniki. W niektórych przypadkach – zwłaszcza z małymi wartościami *inflation* – dokładność i kompletność mogą być niewielkie. Poza tym wysokie wartości delt można uzyskać dla progu 0,5 i środkowych wartości badanego przedziału *inflation*, co potwierdza poprawność intuicyjnego, pierwotnego wyboru parametrów grupowania DAC.

7 Podsumowanie

W pracy zaproponowano metody analizy spójności i zgodności kolekcji dokumentów WWW. Metody te łączą klasyczne elementy wyszukiwania informacji, jak indeksy terminów ważonych i rankingi oparte na ważeniu terminów, z aktualnymi trendami w dziedzinie analizy dokumentów WWW, jak analiza semantyki dokumentów i ich kontekstu, np. autorów, czyli analiza sieci społecznej w WWW. Metody zostały zaproponowane na bardzo ogólnym poziomie, dzięki czemu mogą mieć one szerokie zastosowanie. Weryfikację przeprowadzono dla wybranej metody na konkretnym przypadku zastosowania analizy spójności i zgodności, mianowicie na poprawie wyników tradycyjnej wyszukiwarki. Pojęcie „tradycyjna wyszukiwarka" oznacza silnik indeksujący dokumenty za pomocą terminów ważonych i tworzący ranking na podstawie miar typu tf-idf. Poprawa wyników ma na celu poprawienie dokładności rankingu, tak aby szybciej zaspokoić potrzebę informacyjną użytkownika. Poprzez analizę semantyki związków między obiektami WWW dokumenty potencjalnie najciekawsze dla użytkownika po zadanym zapytaniu umieszczane są na jak najwyższych pozycjach, aby użytkownik dotarł do nich jak najwcześniej. Ze względu na dostępność informacji semantycznej o związkach między dokumentami, autorami i słowami kluczowymi na platformie WordPress, zdecydowano weryfikację przeprowadzić dla blogów publikowanych w serwisie wordpress.com.

Weryfikacją zastosowania metod spójności i zgodności było porównanie rankingu generowanego przez silnik Solr z zaimplementowanym, autorskim algorytmem rerankingu. Weryfikację empiryczną musiała poprzedzić możliwość oceny pertynencji przez użytkowników. W tym celu zaimplementowano aplikację, która poza poprawianiem pierwotnego rankingu pozwalała m.in.:

- analizować spójność i zgodność w czasie rzeczywistym
- poszerzać/zawężać analizę kontrolując progi istotności miar spójności i zgodności
- zmieniać perspektywę analizy za pomocą faset
- prezentować pogrupowane wyniki, umożliwiając analizę na różnych poziomach ziarnistości.

Aplikacja obliczała również przyjęte miary weryfikacji, czyli dokładności i kompletności porządkowej oraz współczynniki korelacji rankingów tak, że po zakończeniu testów zebrane dane i wyliczenia nie wymagały dodatkowej obróbki. Wyniki wykazały, że w zastosowanie rerankingu okazało się słuszne. Grupowanie grafu DAC i ponowne rankowanie dokumentów w odpowiedzi wyszukiwarki poprawia dokładność i kompletność porządkową zwróconej kolekcji. Odpowiednie testy potwierdziły istotność statyczną tych wyników dając podstawy do odrzucenia hipotezy zerowej. Tak więc, w ogólności dokumenty posortowane według rankingu semantycznego DAC, cechują się większą dokładnością i kompletnością porządkową niż posortowane według wyłącznie rankingu Solr.

Dodatkowo przeprowadzono badanie współczynników korelacji rankingów Solr i DAC z rankingiem idealnym. Jednak wyniki okazały się nie być istotnymi statystycznie. Średnie dokładności *P@n* i kompletności *R@n* zostały dodatkowo zilustrowane wykresami na osi wartości odcięcia dokumentów DCV. Dyskusję wyników zamknięto weryfikacją intuicyjnie dobranych parametrów grupowania DAC w kolekcjach testowych.

7.1 Weryfikacja tezy

Na początku pracy postawiona została teza: „Dzięki wykorzystaniu spójności i zgodności, wynikających z informacji semantycznej na temat obiektów WWW i ich związków, możliwe jest poprawienie dokładności i kompletności oraz sposobu prezentacji wyników wyszukiwania". Jak wskazano w podrozdziale 1.5.1 Teza s, 19, pertynencja jest szczególnym przypadkiem relewancji: relewancji dokumentu również względem użytkownika, a nie jak się tradycyjnie przyjmuje tylko względem zapytania. Można więc uznać, że weryfikacja rerankingu opartego na metodach analizy

spójności i zgodności potwierdza tę część tezy dotyczącą poprawy dokładności i kompletności. Jeśli chodzi o poprawę sposobu prezentacji wyników, to stworzenie aplikacji wyszukiwarki analitycznej było dowodem koncepcji (ang. *proof of concept*) prezentacji umożliwiającej analizę. Wspomniane wcześniej możliwości rerankingu, kontroli progów istotności i przez to zakresu analizy oraz zmiany perspektywy według faset to według najlepszej wiedzy autora pionierskie zestawienie funkcjonalności w jednej aplikacji. Dodatkowo, aspekt poprawy prezentacji, który próbowano zmierzyć to wizualizacja grup w wynikowych kolekcjach. Jak zbadano za pomocą miary dopasowania, heurystyka, zakładająca relewancję/pertynencję bądź jej brak dla wszystkich dokumentów z grupy na podstawie jednego reprezentanta, może być bardzo efektywna. Należy jednak pamiętać, że w dyskusji wskazano, że stosowanie tej miary może wymagać dodatkowych ograniczeń.

7.2 Możliwe kierunki dalszych badań

Szerokie zastosowanie metod analizy spójności i zgodności dokumentów WWW, wynikające ze zdefiniowania ich na dosyć ogólnym poziomie daje ogromne możliwości badawcze w różnych dziedzinach analizy WWW. Przedstawione metody analizy zostały zainspirowane dostępnością danych przez WWW, które można przetwarzać i analizować w sposób automatyczny. Obecny trend WWW to *Linked Data*, czyli umieszczanie w WWW nie tylko dokumentów, ale i surowych danych.

Przeznaczeniem dokumentów jest ich czytanie, przeznaczeniem danych jest ich analiza. Jak przekonywał Tim Berners-Lee, w lutym w wystąpieniu na konferencji TED (Tim Berners-Lee 2009), linkowane dane to ogromny potencjał, który można będzie uwolnić metodami analitycznymi, ale potrzebny jest jednolity dostęp do danych. Dziś możemy nazwać adresem zaczynającym się od http://... nie tylko dokumenty w WWW, ale ludzi, firmy, produkty, zdarzenia, ... Za każdym razem, tworząc link z jednego obiektu identyfikowanego przez URL do drugiego, tworzony jest związek. Nawet nie jawnie zaznaczając w serwisie społecznościowym swoich znajomych czy ulubione filmy tworzymy związki. Czyli tworzymy *Linked Data*.

Według (Brian Wilson 2008b) do niedawna zaniedbywany atrybut `rel` znacznika `<a>` ostatnio zyskał znaczną popularność, a `tag` i `category` są trzecią i czwartą najczęściej używaną wartością tego atrybutu. To może oznaczać, że idea folksonomii w WWW rozszerza się. Podobnie ma się sprawa z grafem związków z serwisów społecznościowych. Dzięki coraz szerszej implementacji w popularnych serwisach, zaproponowano przez Google API *Open Social*, informacje o związkach między użytkownikami, czyli tzw. „graf", będzie coraz bardziej dostępny dla narzędzi analitycznych. Widząc gwałtownie rosnącą popularność serwisów społecznościowych, czy ogromne nakłady finansowe na udostępnienie danych rządowych w ramach projektu „przejrzystości rządzących", widać, że WWW staje się czymś więcej niż miejscem publikowania dokumentów i wyszukiwania informacji. Staje się globalną bazą pozwalającą niezależnie jednym użytkownikom publikować dane, a drugim je przetwarzać i analizować. Dotyczy to wszystkich dziedzin, w których analiza danych jest niezbędna w celu wydobycia informacji.

Zgodnie z tym trendem wyszukiwanie dokumentów nie będzie już tak istotne, jak analizowane danych. Nie czekając na czyjeś opracowanie danych, każdy użytkownik będzie mógł sam odpowiednim narzędziem przeanalizować surowe dane. Dlatego wspominana kilka razy kwestia zastąpienia w niedługim czasie tradycyjnej wyszukiwarki wyszukiwarką analityczną wydaje się całkiem prawdopodobna. Wszystko to jednak dotyczy technologii, która tak dynamicznie się rozwija, że nikt nie jest w stanie przewidzieć jak będzie wyglądała WWW za kilka lat.

Nie czekając na rozwój WWW dalsze badania nad jej analizą pod kątem spójności i zgodności można rozwijać na wielu płaszczyznach. Tworząc konkretny przypadek zastosowania metod analizy do rerankingu okazało się, że w wielu krytycznych miejscach trzeba było wybrać jedną z wielu możliwych ścieżek, żeby eksperyment stał się praktycznie możliwy. Przede wszystkim sam wybór trzech typów obiektów jako reprezentantów WWW jest już pewnym zawężeniem. Jak wspomniano wyżej obecnie w WWW nie tylko dokumenty, autorów i pojęcia można wyróżnić i znaleźć między

nimi związki. Być może zasadne byłoby wzięcie pod uwagę: zdarzeń (i ogólnie aspektu czasowego), przedmiotów, projektów, firm, zgrupowań i innych rozróżnialnych za pomocą identyfikatorów URI.

Poza wyborem węzłów do grafu modelującego WWW, należy również rozważyć źródła i miary określające związki pomiędzy nimi. Za przykład można tu podać problem wyznaczenia wagi związku między tagami w weryfikowanym modelu blogosfery. Waga krawędzi między węzłami C została na potrzeby eksperymentu przyjęta jako odwrotność odległości między nimi w ontologii WordNet. Odległość, czyli długość ścieżki to tylko jedna z kilkunastu miar zaimplementowanych dla pojęć w perl'owej bibliotece WordNet::Similarity. Wzięcie pod uwagę innych miar może być przedmiotem dalszych badań, ponieważ badanie ich wspływu na jakość związków wykraczało poza niniejszą pracę. Podobne ograniczenie trzeba było przyjąć dla wieloznaczności pojęć – wzięto pod uwagę pierwsze znaczenie, oraz dla założenia którą częścią mowy jest pojęcie – przyjęto, że rzeczownikiem. Te wybory najprawdopodobniej mają wpływ na związki i pośrednio na analizę zgodności. Jednak arbitralny wybór jednego podejścia był kompromisem wymaganym przez możliwość empirycznej weryfikacji metody ogólnej, czyli analizy DAC. Podobne kompromisy musiały być podjęte na różnych etapach aplikowania teorii w praktykę, co zostawia wiele miejsca na przyszłe badania i eksperymenty.

Innym problemem, dotyczącym samego sposobu modelowania WWW, na którym można się w przyszłości skupić, to związki pośrednie. Wyznaczanie związków pośrednich na podstawie związków istniejących może pozwolić na analizę tych aspektów powiązań dwóch obiektów, o których nie mamy informacji wprost. Ten temat został poruszony w Kopel i Daniłowicz 2004b) oraz (Kopel i Zgrzywa 2008).

Dla kontrastu, problemem praktycznym, wynikającym z czasu działania algorytmu grupowania grafu, wymagającym w przyszłości rozwiązania, może być regrupowanie. W aplikacji weryfikującej metodę przyjęto, że dla każdego zapytania graf jest od nowa tworzony i grupowany. Jest to działanie bardzo nieefektywne, zwłaszcza w kontekście zastosowania go w systemie online'owym. W momencie, gdy wielu użytkowników równolegle zadaje zapytania, warto rozważyć buforowanie i ponowne wykorzystanie grafu i grup. Być może celowe okaże się utrzymywanie globalnego grafu DAC i wyników grupowania dla najpopularniejszych zapytań. Biorąc pod uwagę dynamikę WWW, czyli pojawianie się i potrzeba zaindeksowania nowych dokumentów, autorów i pojęć pojawia się pytanie. Czy z każdym nowym dokumentem przeprowadzać nowe grupowanie czy może przychodzące dokumenty klasyfikować do istniejących grup? Jeśli klasyfikacja okaże się bardziej wydajna, to pojawia się kolejne pytanie: Jak długo można klasyfikować nowe dokumenty, jaki przyjąć wyznacznik potrzeby ponownego grupowania?

Problemem jeszcze innej klasy, wymagającym zbadania, może być problem odpowiedzi na nowe potrzeby informacyjne użytkownika. Można rozważyć jaki sposób zastosować analizę spójności i zgodności do pozyskiwania nowych dokumentów o zadanym profilu. Załóżmy, że użytkownik ma swoją bazową kolekcję dokumentów WWW, np. w postaci czytnika RSS. Tradycyjnie użytkownik otrzymuje nowe dokumenty z subskrybowanych źródeł niezależnie od ich wzajemnych związków. Można się zastanowić w jaki sposób efektywnie dostarczać użytkownikowi nowe dokumenty. Nie chodzi jedynie o analizę spójności i zgodności dokumentów, które użytkownik i tak by otrzymał przez wybrane kanały. Za pomocą spójności i zgodności można spróbować rekomendować użytkownikowi nowe dokumenty czy całe nowe kanały, których użytkownik nie zna. Ten aspekt rekomendacji został w pewnym stopniu omówiony w Kopel i Kazienko 2007).

Ostatecznie można zastanowić się nad wdrożeniem możliwości analizy spójności i zgodności do działających dziś aplikacji WWW. Rozważmy następujący scenariusz zaczerpnięty z życia: użytkownik chce kupić w popularnych serwisie aukcyjnym myszkę i klawiaturę. Przy czym mniej istotny jest dla niego konkretny model myszy i klawiatury, a bardziej sam fakt, żeby zapłacić za tylko jedną przesyłkę. Aby oba przedmioty mogły zostać wysłane w jednej przesyłce aukcje muszą być wystawione przez jednego użytkownika. Grupowanie czy nawet proste szukanie aukcji według

sprzedawców nie jest na platformie aukcyjnej możliwe. Przyjmując, że strony aukcji to węzły *D* (dokumenty WWW), a sprzedający to węzły *A* (autorzy dokumentów) – potrzebę informacyjną użytkownika można by rozwiązać prostą analizą spójności w grafie DA. Dodatkowo, wprowadzając do serwisu tagi, czy wydobywając słowa kluczowe można by zamodelować cały graf DAC. Wtedy ewentualna analiza zgodności mogłaby pozwolić na rekomendację innych drobiazgów potrzebnych przy zakupie myszy i klawiatury, jak podkładka, płyn do czyszczenia klawiatury czy koncentrator USB.

Bibliografia

Alexa. 2009. Alexa the Web Information Company. http://alexa.com/.

Allsopp, John . 2007. *Microformats: Empowering Your Markup for Web 2.0*. Friends of ED, Marzec.

Alpert, Jesse, i Nissan Hajaj. 2008. We knew the web was big... *Official Google Blog*. Lipiec 25. http://googleblog.blogspot.com/2008/07/we-knew-web-was-big.html.

Anderson, Shaun. 2008. Google SEO Test - Google Prefers Valid HTML & CSS | Hobo. *Hobo*. Lipiec 25. http://www.hobo-web.co.uk/seo-blog/index.php/official-google-prefers-valid-html-css/.

Axelsson, J., B. Epperson, M. Ishikawa, S. McCarron, A. Navarro, i S. Pemberton. 2006. XHTML 2.0. *W3C Working Draft 26 July 2006, World-Wide Web Consortium*. http://www.w3.org/TR/xhtml2/mod-metaAttributes.html.

Berners-Lee, T., L. Masinter, i M. McCahill. 1994. *Uniform Resource Locators (URL)*. Grudzień. http://tools.ietf.org/html/rfc1738.

Berners-Lee, Tim. 1995. Making a server. http://www.w3.org/Provider/ServerWriter.html.

---. 2007a. Linked Data - Design Issues. Maj 2. http://www.w3.org/DesignIssues/LinkedData.html.

---. 2007b. Giant Global Graph. Weblog. *timbl's blog | Decentralized Information Group (DIG) Breadcrumbs*. Listopad 21. http://dig.csail.mit.edu/breadcrumbs/blog/4.

---. 2009. *Tim Berners-Lee: The next Web of open, linked data*. Luty. http://www.youtube.com/watch?v=OM6XIICm_qo.

Berners-Lee, Tim, Mark Fischetti, i Michael L. Dertouzos. 1999. *Weaving the Web : The Original Design and Ultimate Destiny of the World Wide Web by its Inventor*. Wrzesień 22.

Berners-Lee, Tim, James Hendler, i Ora Lassila. 2001. The semantic Web. *Scientific American* 284, no. 5: 28-37.

Bernstein, Philip A., Vassco Hadzilacos, i Nathan Goodman. 1987. *Concurrency control and recovery in database systems*. Addison-Wesley Longman Publishing Co., Inc. http://portal.acm.org/citation.cfm?id=12518.

Bilgil, Melih. 2009. PICOL - Pictorial Communication Language - Icons & Pictorgrams. http://picol.org/.

Bos, Bert, Tantek Çelik, H\aakon Wium Lie, i Ian Hickson. 2007. *Cascading Style Sheets Level 2 Revision 1 (CSS 2.1) Specification*. Candidate Recommendation. W3C, Lipiec.

Brandes, Gaertler, i Wagner. 2003. Experiments on Graph Clustering Algorithms. *Algorithms - ESA 2003*.

Brian Wilson. 2008a. MAMA: Key findings. *Opera Developer Community*. Październik 15. http://dev.opera.com/articles/view/mama-key-findings/.

---. 2008b. MAMA: Hyperlinks. *Opera Developer Community*. Listopad 21. http://dev.opera.com/articles/view/mama-key-findings/.

Brickley, Dan, i Libby Miller. 2007. *FOAF Vocabulary Specification 0.91*. Listopad 2. http://xmlns.com/foaf/spec/.

Brin, Sergey, i Lawrence Page. 1998. The anatomy of a large-scale hypertextual Web search engine. *Computer Networks and ISDN Systems* 30, no. 1-7: 107-117.

Brown, Allen, i Hugo Haas. 2004. *Web Services Glossary*. W3C Note.

Bush, Vannevar. 1989. As we may think (1945). W *Perspectives on the computer revolution*, 49-61. Ablex Publishing Corp. http://portal.acm.org/citation.cfm?id=98326.98339.

Cao, Pei. 1998. Maintaining Strong Cache Consistency in the World Wide Web. Text. Kwiecień. http://csdl2.computer.org/persagen/DLAbsToc.jsp?resourcePath=/dl/trans/tc/&toc=comp/trans/tc/1998/04/t4toc.xml&DOI=10.1109/12.675713.

Cellary, Wojciech, i Geneviève Jomier. 1990. Consistency of versions in objects-oriented databases. W *Proceedings of the sixteenth international conference on Very large databases*, 432-441. Brisbane, Australia: Morgan Kaufmann Publishers Inc. http://portal.acm.org/citation.cfm?id=94488.

Chamberlin, Don, Mary F. Fern\'andez, J\'er^ome Sim\'eon, Daniela Florescu, Scott Boag, i Jonathan Robie. 2007. *XQuery 1.0: An XML Query Language*. W3C Recommendation.

Chisholm, Wendy, Ian Jacobs, i Gregg Vanderheiden. 1999. *Web Content Accessibility Guidelines 1.0*. W3C Recommendation. W3C, Maj.

Clark, James. 1999. *XSL Transformations (XSLT) Version 1.0*. W3C Recommendation.

Contributors. 2008. Wikipedia, the free encyclopedia. Wikimedia Foundation, Inc. http://en.wikipedia.org/wiki/Main_Page.

Daniłowicz, Czesław, i Marek Kopel. 2003. Analysis method of coherency and topical relevancy for web document collections. W , 83-89. Information Systems Applications and Technology ISAT 2003 Seminar. Proceedings of the 24th international scientific school, [Szklarska Poręba, 25-26 September, 2003]. Wrocław: Oficyna Wydawnicza Politechniki Wrocławskiej.

Daniłowicz, Czesław, i Ngoc Thanh Nguyen. 2000. Consensus-Based Methods for Restoring Consistency of Replicated Data. *Intelligent Information Systems: Proceedings of the Iis' 2000 Symposium, Bystra, Poland, June 12-16, 2000*.

---. 2003. Consensus Methods for Solving Inconsistency of Replicated Data in Distributed Systems. *Distributed and Parallel Databases* 14, no. 1 (Lipiec 1): 53-69. doi:10.1023/A:1022835811280.

Daniłowicz, Czesław, Ngoc Thanh Nguyen, i Łukasz Jankowski. 2002. *Metody wyboru reprezentacji stanu wiedzy agentów w systemach multiagenckich*. Wrocław. Oficyna Wydawnicza Politechniki Wrocławskiej.

Decker, Martin, Guido Moerkotte, Holger Müller, i Joachim Posegga. 1991. Consistency Driven Planning. W *Proceedings of the 5th Portuguese Conference on Artificial Intelligence*, 195-209. Springer-Verlag. http://portal.acm.org/citation.cfm?id=651028.

Deerwester, Scott C., Susan T. Dumais, George W. Furnas, Richard A. Harshman, Thomas K. Landauer, Karen E. Lochbaum, i Lynn A. Streeter. 1989. United States Patent: 4839853 - Computer information retrieval using latent semantic structure. Czerwiec 13. http://patft.uspto.gov/netacgi/nph-Parser?Sect1=PTO1&Sect2=HITOFF&d=PALL&p=1&u=%2Fnetahtml%2FPTO%2Fsrchnum.htm&r=1&f=G&l=50&s1=4839853.PN.&OS=PN/4839853&RS=PN/4839853.

Deo, Narsingh. 1980. *Teoria Grafów I Jej Zastosowania W Technice I Informatyce*. BNI Biblioteka Naukowa Inżyniera. Warszawa: PWN.

Diestel, Reinhard. 2006. *Graph Theory (Graduate Texts in Mathematics)*. Springer, Luty.

http://www.amazon.ca/exec/obidos/redirect?tag=citeulike09-20&path=ASIN/3540261834.

Ding, Li, Tim Finin, i Anupam Joshi. 2005. Analyzing Social Networks on the Semantic Web. *IEEE Intelligent Systems* 9, no. 1 (Styczeń).

Ding, Li, Tim Finin, Yun Peng, Paulo Pinheiro Da Silva, i Deborah L. McGuinness. 2005. Tracking RDF Graph Provenance using RDF Molecules. *2005, Proceedings of the Fourth International Semantic Web Conference* 2: 51-57. doi:10.1.1.74.7085.

Ding, Li, Lina Zhou, Tim Finin, i Anupam Joshi. 2005. How the Semantic Web is Being Used: An Analysis of FOAF Documents. W *Proceedings of the Proceedings of the 38th Annual Hawaii International Conference on System Sciences (HICSS'05) - Track 4 - Volume 04*, 113.3. IEEE Computer Society. http://portal.acm.org/citation.cfm?id=1042928.

Dongen, Stijn van. 2000. *Graph clustering by flow simulation [PhD dissertation] Utrecht (The Netherlands): University of Utrecht. 169 p*.

---. 2009. MCL - an algorithm for clustering graphs. http://www.micans.org/mcl/.

Dubost, Karl, i Mark Skall. 2005. W3C QA - Quality Assurance glossary. Kwiecień 28. http://www.w3.org/QA/glossary.

Duce, David. 2003. *Portable Network Graphics (PNG) Specification (Second Edition)*. W3C Recommendation. W3C, Listopad.

Ebbinghaus, H.-D., J. Flum, i W. Thomas. 1996. *Mathematical Logic*. 2. wyd. Springer, Listopad 15.

ECMA International. 1999. *ECMA-262: ECMAScript Language Specification*. ECMA (European Association for Standardizing Information and Communication Systems), Grudzień. http://www.ecma-international.org/publications/standards/Ecma-327.htm.

Eswaran, K. P., J. N. Gray, R. A. Lorie, i I. L. Traiger. 1976. The notions of consistency and predicate locks in a database system. *Commun. ACM* 19, no. 11: 624-633. doi:10.1145/360363.360369.

Fielding, R, J Gettys, J Mogul, H Frystyk, L Masinter, P Leach, i T Berners-Lee. 1999. *RFC 2616: Hypertext Transfer Protocol - HTTP/1.1*. Czerwiec. http://www.rfc-editor.org/rfc/rfc2616.txt.

Fielding, R. T. 2000. Architectural Styles and the Design of Network-based Software Architectures. UNIVERSITY OF CALIFORNIA.

Garfield, Eugene. 1964. "Science Citation Index"-A New Dimension in Indexing. *Science* 144, no. 3619 (Maj 8): 649-654.

---. 2006. The History and Meaning of the Journal Impact Factor. *JAMA* 295, no. 1 (Styczeń 4): 90-93. doi:10.1001/jama.295.1.90.

Garrett, Jesse James. 2005. Ajax: A New Approach to Web Applications. *Adaptive Path* (Luty 18).

---. 2008a. *Aurora (Part 1)*. Sierpień. http://vimeo.com/1450211?pg=embed&sec=1450211.

---. 2008b. Aurora: Concept Video Part 1. Blog. *Adaptive Path*. Sierpień 4. http://www.adaptivepath.com/blog/2008/08/04/aurora-concept-video-part-1/.

---. 2008c. Aurora Interface Guide. Sierpień 8. http://www.adaptivepath.com/blog/wp-content/uploads/2008/08/adaptive-path_aurora_interface-guide.pdf.

---. 2008d. Aurora design concepts. Sierpień 8. http://www.adaptivepath.com/blog/wp-

content/uploads/2008/08/adaptive-path_aurora_design-concepts.pdf.

Gharachorloo, Kourosh, Daniel Lenoski, James Laudon, Phillip Gibbons, Anoop Gupta, i John Hennessy. 1990. Memory consistency and event ordering in scalable shared-memory multiprocessors. W *Proceedings of the 17th annual international symposium on Computer Architecture*, 15-26. Seattle, Washington, United States: ACM. doi:10.1145/325164.325102. http://portal.acm.org/citation.cfm?id=325164.325102.

Girschweiler, Bruno. 1995. *Hypertext Terms*. W3C Recommendation.

GMPG. 2009. *XFN - XHTML Friends Network*. http://gmpg.org/xfn/.

Golbeck, Parsia, i Hendler. 2003. Trust Networks on the Semantic Web. *Cooperative Information Agents VII*.

Goldfarb, Charles F., Steven R. Newcomb, W. Eliot Kimber, i Peter J. Newcomb. 1997. *Information processing - Hypermedia/Time-based Structuring Language (HyTime) - 2d edition* . http://www1.y12.doe.gov/capabilities/sgml/wg8/document/n1920/.

Google. 2008. Search Engine Optimization (SEO) - Webmaster Help Center. http://www.google.com/support/webmasters/bin/answer.py?answer=35291.

Gordon, Michael, i Praveen Pathak. 1999. Finding information on the World Wide Web: the retrieval effectiveness of search engines. *Inf. Process. Manage.* 35, no. 2: 141-180.

Grudin, J. 1989. The case against user interface consistency. *Communications of the ACM* 32, no. 10: 1164-1173.

Haerder, T., i A. Reuter. 1983. Principles of Transaction-Oriented Database Recovery. *ACM Computing Surveys* 15, no. 4: 287-317.

Handy, J. 1998. *The Cache Memory Book*. Morgan Kaufmann.

Harris, Jonathan, i Sep Kamvar. 2009. I Want You To Want Me. http://iwantyoutowantme.org/.

Hartmann, Alexander K, i Martin Weigt. 2006. Introduction to graphs. *cond-mat/0602129* (Luty 6). http://arxiv.org/abs/cond-mat/0602129.

Hickson, Ian. 2009. Acid Tests - The Web Standards Project. http://www.acidtests.org/.

Hjelm, Johan, Chris Woodrow, Luu Tran, Hidetaka Ohto, Franklin Reynolds, Mark H. Butler, i Graham Klyne. 2004. *Composite Capability/Preference Profiles (CC/PP): Structure and Vocabularies 1.0*. W3C Recommendation.

Hofmann, Thomas. 1999. Probabilistic latent semantic indexing. W *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, 50-57. Berkeley, California, United States: ACM. doi:10.1145/312624.312649. http://portal.acm.org/citation.cfm?id=312624.312649&type=series.

Huynh, David. 2008a. Jon Udell's Interviews with Innovators | David Huynh. IT Conversations. Sierpień. http://itc.conversationsnetwork.org/shows/detail3793.html.

Huynh, David . 2008b. *Freebase Parallax: A new way to browse and explore data on Vimeo*. Screencast. Sierpień. http://vimeo.com/1513562?pg=embed&sec=1513562.

Hyatt, David, i Ian Hickson. 2008. *HTML 5*. W3C Working Draft. W3C.

Ishikawa, Masayasu, Daniel Austin, Shane McCarron, Mark Birbeck, i Subramanian Peruvemba. 2008. *XHTML\texttrademark Modularization 1.1*. W3C Proposed Reccommendation. W3C, Czerwiec.

Ishikawa, Masayasu, i Steven Pemberton. 2002. *HLink: Link recognition for the XHTML Family.*

W3C Working Draft. W3C.

Jacobs, Ian, Jon Gunderson, i Eric Hansen. 2002. *User Agent Accessibility Guidelines 1.0*. W3C Recommendation. W3C.

Jacobs, Ian, i Norman Walsh. 2004. *Architecture of the World Wide Web, Volume One*. W3C Recommendation. W3C.

Jun, FUJISAWA, Jon Ferraiolo, i Dean Jackson. 2003. *Scalable Vector Graphics (SVG) 1.1 Specification*. W3C Recommendation. W3C, Styczeń.

Kannan, Ravi, Santosh Vempala, i Adrian Vetta. 2004. On clusterings: Good, bad and spectral. *J. ACM* 51, no. 3: 497-515. doi:10.1145/990308.990313.

Kay, Michael, Don Chamberlin, Scott Boag, Mary F. Fern\'andez, J\'er^ome Sim\'eon, Anders Berglund, i Jonathan Robie. 2007. *XML Path Language (XPath) 2.0*. W3C Recommendation. W3C.

Kelaidis, Manolis. 2007. O'Reilly Media Tools of Change Conference | Manolis Kelaidis. IT Conversations. Czerwiec 20. http://itc.conversationsnetwork.org/shows/detail3339.html.

Kendall, MG. 1938. A New Measure of Rank Correlation. *Biometrika* 30, no. 1/2 (Czerwiec): 93, 81.

Kent, A., M. M. Berry, F. U. Luehrs Jr. , i J. W. Perry. 1955. Machine literature searching VIII. Operational criteria for designing information retrieval systems. *American documentation* 6, no. 2.

Kermarrec, Yvon, i Alberto Soleto. 1997. Managing document consistency over the Web or managing documents duplication. W . Madrid, Kwiecień. http://citeseer.ist.psu.edu/362506.html.

Kleinberg, Jon M. 1998. Authoritative sources in a hyperlinked environment. *Journal of the ACM* 46, no. 5: 604-632.

Kłopotek, Mieczysław Alojzy. 2001. *Inteligentne Wyszukiwarki Internetowe*. Problemy Współczesnej Nauki. Warszawa: Exit.

Kopel, Marek. 2004. Identyfikacja spamu na podstawie analizy spójności wiadomości. W *Materiały konferencyjne*, 301-311. Multimedialne i sieciowe systemy informacyjne [Szklarska Poręba, 16-17 września 2004]. Wrocław: Oficyna Wydawnicza Politechniki Wrocławskiej.

Kopel, Marek, i Czesław Daniłowicz. 2004a. Measuring the importance of concepts and relations between the concepts in a hypertext collection. W , 72-78. Information Systems Architecture and Technology ISAT 2004. Proceedings of the 25th International Scientific School. Information models, concepts, tools and applications, [Szklarska Poręba, 22-25 September 2004]. Wrocław: Oficyna Wydawnicza Politechniki Wrocławskiej.

---. 2004b. Method of completing the consistency graph of a hyperlinked document collection. W *Intelligent technologies for inconsistent knowledge processing.*, 10:145-161. International Series on Advanced Intelligence. Magill: Advanced Knowledge International.

Kopel, Marek, i Przemysław Kazienko. 2007. Application of Agent-Based Personal Web of Trust to Local Document Ranking. W *Agent and Multi-Agent Systems: Technologies and Applications*, 288-297. http://dx.doi.org/10.1007/978-3-540-72830-6_30.

Kopel, Marek, i Aleksander Zgrzywa. 2008. The Consistency and Conformance of Web Document Collection Based on Heterogeneous DAC Graph. W *New Frontiers in Applied Artificial Intelligence*, 321-330. http://dx.doi.org/10.1007/978-3-540-69052-8_34.

---. 2009. Libraries in the Semantic Web Era. W *Advances in Qualitative and Quantitative Methods in Libraries*. World Scientific Publishing Co. (w druku).

Kulikowski, Juliusz Lech. 1986. *Zarys Teorii Grafów: Zastosowania W Technice*. BNI Biblioteka Naukowa Inżyniera. Warszawa: Państwowe Wydawnictwo Naukowe.

Kumar, V. 1992. Algorithms for Constraint-Satisfaction Problems: A Survey. *AI Magazine* 13, no. 1: 32-44.

Langridge, Stuart, i Ian Hickson. 2002. *Pingback 1.0*. http://www.hixie.ch/specs/pingback/pingback.

Lavoie, B., i H. F. Nielsen. 1999. *Web Characterization Terminology & Definitions Sheet*. W3C. Google Scholar.

Leclerc, Y. G., Q. T. Luong, i P. Fua. 2000. Measuring the Self-Consistency of Stereo Algorithms. W *Proceedings of the 6th European Conference on Computer Vision-Part I*, 282-298. Springer-Verlag London, UK.

Lehmann, E. L., i G. Casella. 1998. *Theory of Point Estimation*. Springer.

Lie, Håkon Wium, i Bert Bos. 2008. *Cascading Style Sheets (CSS1) Level 1 Specification*. W3C Recommendation. W3C, Kwiecień.

Mahajan, R., i B. Shneiderman. 1995. A family of user interface consistency checking tools: design analysis of SHERLOCK. W *Proc. of NASA Twentieth Annual Software Engineering Workshop*, 169-188.

Malhotra, Ashok, Jonathan Marsh, Mary Fern\'andez, Marton Nagy, i Norman Walsh. 2007. *XQuery 1.0 and XPath 2.0 Data Model (XDM)*. W3C Recommendation. W3C.

Manning, Christopher D., Prabhakar Raghavan, i Hinrich Schütze. 2008. *Introduction to Information Retrieval*. Cambridge University Press. http://portal.acm.org/citation.cfm?id=1394399.

Marriott, K., i P. J. Stuckey. 1998. *Programming with Constraints: An Introduction*. MIT Press.

Masinter, L. 1998. The "data" URL scheme. http://portal.acm.org/citation.cfm?id=RFC2397.

Masinter, L., T. Berners-Lee, i R. Fielding. 1998. *Uniform Resource Identifiers (URI): Generic Syntax*. Sierpień. http://tools.ietf.org/html/rfc2396.

Mazur, Hanna, i Zygmunt Mazur. 2004. *Projektowanie relacyjnych baz danych*. Wrocław: Oficyna Wydawnicza Politechniki Wrocławskiej.

Mazur, Zygmunt. 2006. Zarządzanie długimi transakcjami. W *Bazy Danych*, 97-116. Prace Naukowe Instytutu Informatyki Stosowanej Politechniki Wrocławskiej 7. Wrocław: Oficyna Wydawnicza Politechniki Wrocławskiej.

Mazurek, Parkoła, i Werla. 2006. Distributed Digital Libraries Platform in the PIONIER Network. *Research and Advanced Technology for Digital Libraries*.

McCarron, Shane, i Murray Altheim. 2001. *XHTML\texttrademark 1.1 - Module-based XHTML*. W3C Recommendation. W3C.

McCarron, Shane, i Masayasu Ishikawa. 2007. *XHTML\texttrademark 1.1 - Module-based XHTML - Second Edition*. W3C Working Draft. W3C.

McGlashan, Scott, Daniel C. Burnett, Bruce Lucas, Ken Rehor, Brad Porter, Steph Tryphonas, Jim Ferrans, Peter Danielsen, Jerry Carter, i Andrew Hunt. 2004. *Voice Extensible Markup Language (VoiceXML) Version 2.0*. W3C Recommendation. W3C.

Milgram, S. 1967. The small world problem. *Psychology today* 2, no. 1: 60-67.

Miller, George A. 2006. WordNet. Princeton University.

Moerkotte, Guido, i Peter C. Lockemann. 1991. Reactive consistency control in deductive databases. *ACM Trans. Database Syst.* 16, no. 4: 670-702. doi:10.1145/115302.115298.

Moon, J., i L. Moser. 1965. On cliques in graphs. *Israel Journal of Mathematics* 3, no. 1 (Marzec 1): 23-28. doi:10.1007/BF02760024.

Mosberger, David. 1993. Memory consistency models. *SIGOPS Oper. Syst. Rev.* 27, no. 1: 18-26. doi:10.1145/160551.160553.

Mullender, Sjoerd, Thierry Michel, Antti Koivisto, Jack Jansen, Dick Bulterman, Nabil Laya\"ıda, Daniel Zucker, i Guido Grassel. 2005. *Synchronized Multimedia Integration Language (SMIL 2.1)*. W3C Recommendation. W3C, Grudzień.

Nelson, Theodore. 1982. *Literary Machines*. {Mindful Pr}, Lipiec 1. http://www.amazon.ca/exec/obidos/redirect?tag=citeulike09-20&path=ASIN/0893470627.

Nguyen, Ngoc Thanh. 2002. Consensus system for solving conflicts in distributed systems. *Information Sciences* 147, no. 1-4: 91-122.

Orchard, David, Eve Maler, i Steven DeRose. 2001. *XML Linking Language (XLink) Version 1.0*. W3C Recommendation. W3C.

O'Reilly, Tim. 2005. *What Is Web 2.0? Design Patterns and Business Models for the Next Generation of Software*. OReilly Media, Inc., Wrzesień.

Page, Lawrence, Sergey Brin, R. Motwani, i T. Winograd. 1998. *The pagerank citation ranking: Bringing order to the web*. Technical report, Stanford Digital Library Technologies Project.

Pańczyk, Michał. 2008. Wykorzystanie złożoności Kołmogorowa do wyznaczania związków kontekstowych poprzez analizę efektów pracy wyszukiwarek internetowych. Praca magisterska, Uniwersytet Marii Curie-Skłodowskiej W Lublinie Wydział Matematyki Fizyki I Informatyki.

Pedersen, T., S. Patwardhan, i J. Michelizzi. 2004. Wordnet:: similarity-measuring the relatedness of concepts. W *Proceedings of the National Conference on Artificial Intelligence*, 1024-1025. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999.

Pell, Barney. 2008. The Semantic Web Gang talk to Powerset CTO, Barney Pell. Maj. http://semanticgang.talis.com/2008/05/16/may-2008-the-semantic-web-gang-talk-to-powerset-cto-barney-pell/.

Pemberton, Steven. 2002. *XHTML\texttrademark 1.0 The Extensible HyperText Markup Language (Second Edition)*. W3C Recommendation. W3C.

Piasecki, Maciej, Stanisław Szpakowicz, i Bartosz Broda. 2009. A *Wordnet from the Ground Up*. Oficyna Wydawnicza Politechniki Wrocławskiej. http://www.plwordnet.pwr.wroc.pl/main/content/files/publications/A_Wordnet_from_the_Ground_Up.pdf.

Pipino, Leo L., Yang W. Lee, i Richard Y. Wang. 2002. Data quality assessment. *Commun. ACM* 45, no. 4: 211-218. doi:10.1145/505248.506010.

PSNC. 2006. Dokumentacja Wielkopolskiej Biblioteki Cyfrowej: Najczęściej zadawane pytania. PSNC, Poznan Supercomputer and Networking Center. http://www.wbc.poznan.pl/dlibra/text?id=faq.

Pujol, Josep M., Ramon Sangüesa, i Jordi Delgado. 2002. Extracting reputation in multi agent

systems by means of social network topology. W *Proceedings of the first international joint conference on Autonomous agents and multiagent systems: part 1*, 467-474. Bologna, Italy: ACM.

PWN. 2008. Słownik języka polskiego. Wydawnictwo Naukowe PWN SA.

Raggett, D., A. Le Hors, i I. Jacobs. 1999. HTML 4.01 Specification. *W3C Recommendation REC-html401-19991224, World Wide Web Consortium (W3C), Dec*. http://www.w3.org/TR/html401/struct/links.html#h-12.3.

Raynal, M., i A. Schiper. 1996. A Suite of Formal Definitions for Consistency Criteria in Distributed Shared Memories. *Proceedings Int Conf on Parallel and Distributed Computing (PDCS'96)*: 125–130.

Reed, D., i G. Strongin. 2004. *The Dataweb: An Introduction to XDI, A White Paper for the OASIS XDI Technical Committee-v2*. April.

Rosenthal, Lynne, Dominique Haza\"el-Massieux, Karl Dubost, i Lofton Henderson. 2005. *QA Framework: Specification Guidelines*. W3C Recommendation. W3C.

Rówińska, Magdalena. 2008. Ponad trzy miliony Polaków czyta blogi - Biuro Prasowe - Portal Gazeta.pl. Wrzesień 2. http://media.netpr.pl/PressOffice/PressRelease.105842.po?rss=true.

Salton, G., i M. J. McGill. 1983. Introduction to Modern Information Retrieval. *McGrawHill Book Co., New York*.

Santos, C. A. S., P. N. M. Sampaio, i J. P. Courtiat. 1999. Revisiting the concept of hypermedia document consistency. W *Proceedings of the seventh ACM international conference on Multimedia (Part 2)*, 183-186. Orlando, Florida, United States: ACM. doi:10.1145/319878.319927. http://portal.acm.org/citation.cfm?id=319878.319927.

Saracevic, Tefko. 2007. Relevance: A review of the literature and a framework for thinking on the notion in information science. Part II: nature and manifestations of relevance. *J. Am. Soc. Inf. Sci. Technol.* 58, no. 13: 1915-1933.

Six Apart. 2002. *TrackBack Technical Specification*. http://www.sixapart.com/pronet/docs/trackback_spec.

Soergel, D. 1994. Indexing and Retrieval Performance: The Logical Evidence. *Journal of the American Society for Information Science* 45, no. 8: 589-99.

Spearman, C. 1904. The proof and measurement of association between two things. *The American journal of psychology* 15: 72-101.

Steinke, Robert C., i Gary J. Nutt. 2004. A unified theory of shared memory consistency. *J. ACM* 51, no. 5: 800-849. doi:10.1145/1017460.1017464.

Sun Microsystems, Inc. 2003. Java(TM) 2 SDK Documentation. Sun Microsystems, Inc. http://java.sun.com/j2se/1.4.2/docs/index.html.

Swanson, D. R. 1986. Subjective versus objective relevance in bibliographic retrieval systems. *The Library Quarterly*: 389-398.

Teniente, E., i A. Olivé. 1995. Updating knowledge bases while maintaining their consistency. *The VLDB Journal The International Journal on Very Large Data Bases* 4, no. 2: 193-241.

Thereaux, Olivier. 2008. W3C Glossary and Dictionary. W3C - World Wide Web Consortium - http://www.w3.org. http://www.w3.org/2003/glossary/.

Treviranus, Jutta, Charles McCathieNevile, Jan Richards, i Ian Jacobs. 2000. *Authoring Tool Accessibility Guidelines 1.0*. W3C Recommendation.

U-M Gateway . 2008. U-M Internet Publishing Policies, Guidelines, & Instructions. The Regents of the University of Michigan. http://www.umich.edu/policy_guidelines6.php.

Voß, J. 2007. Tagging, Folksonomy & Co - Renaissance of Manual Indexing? *cs/0701072v2* (Styczeń 10).

WAI. 2009. Web Accessibility Initiative (WAI) - home page. http://www.w3.org/WAI/.

Watts, Duncan J., i Steven H. Strogatz. 1998. Collective dynamics of /`small-world/' networks. *Nature* 393, no. 6684 (Czerwiec 4): 440-442. doi:10.1038/30918.

Wesch, Michael . 2007. *The Machine is Us/ing Us (Final Version)*. Marzec 8. http://www.youtube.com/watch?v=NLlGopyXT_g.

Yourdon, E., i L. L. Constantine. 1979. *Structured Design: Fundamentals of a Discipline of Computer Program and Systems Design*. Prentice-Hall, Inc. Upper Saddle River, NJ, USA.

Dodatek A

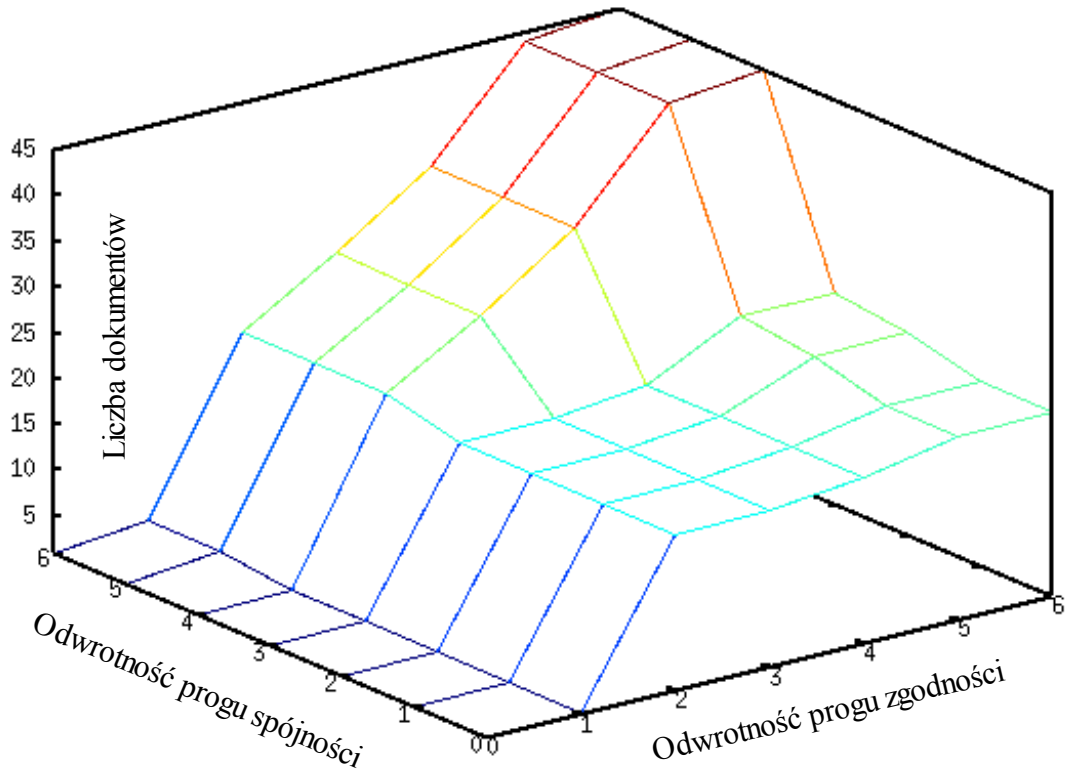
Indeks definicji

Dodatek B

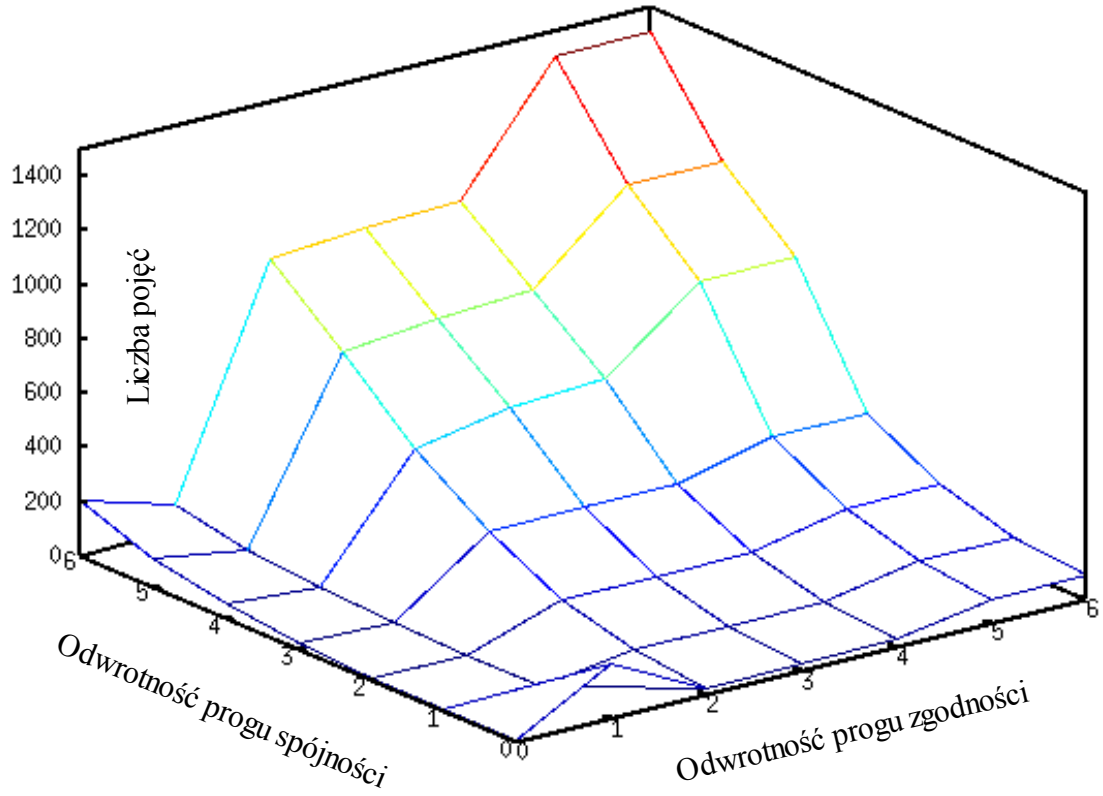
Indeks rysunków

Dodatek C

Wykresy zależności wyników od progów spójności i zgodności



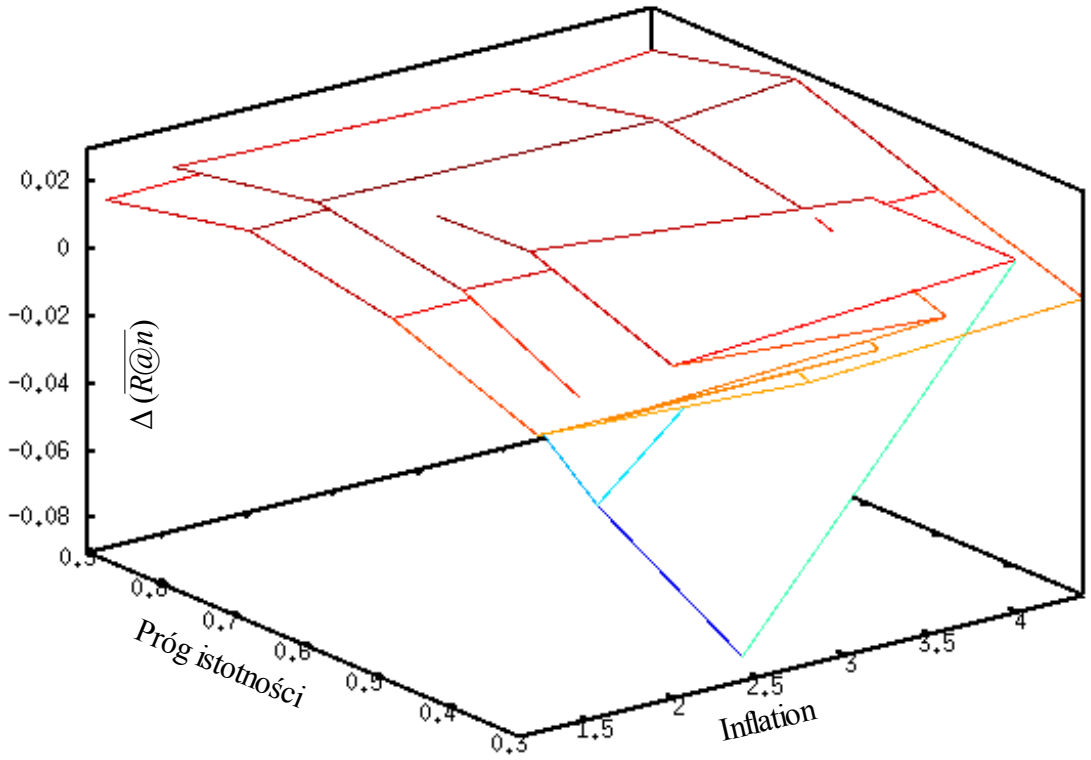
Rysunek 1. Wykres zależności liczby dokumentów w przykładowej kolekcji wynikowej od progów spójności i zgodności



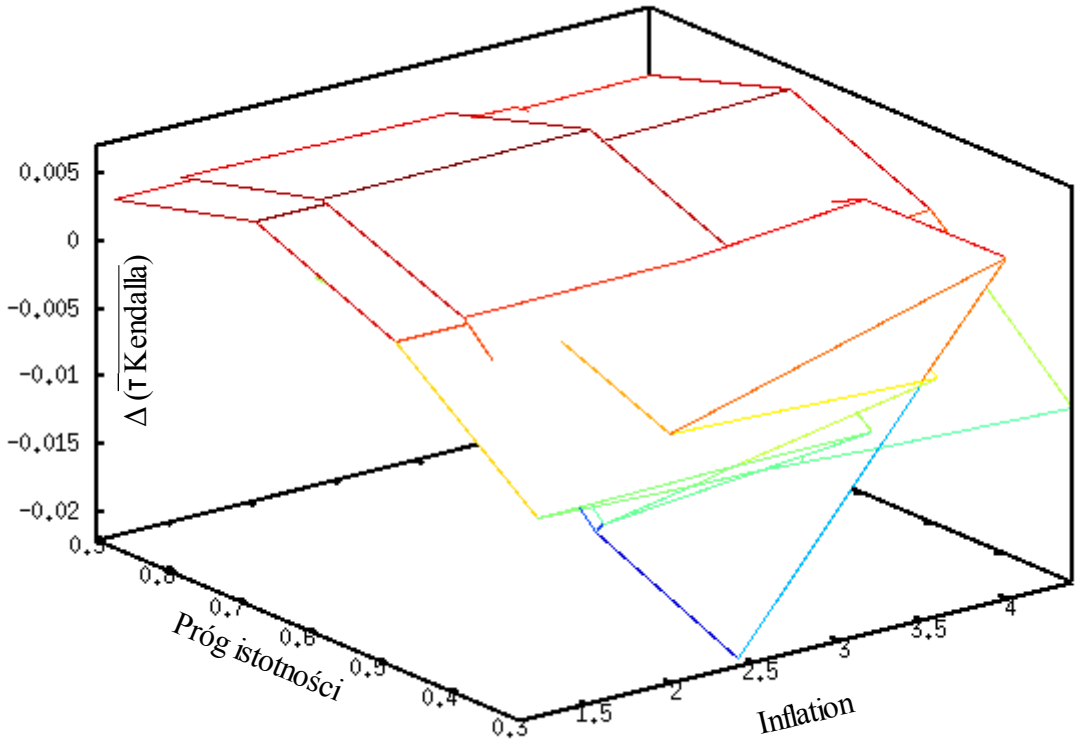
Rysunek 2. Wykres zależności liczby unikatowych pojęć przypisanych do dokumentów w przykładowej kolekcji wynikowej od progów spójności i zgodności

Dodatek D

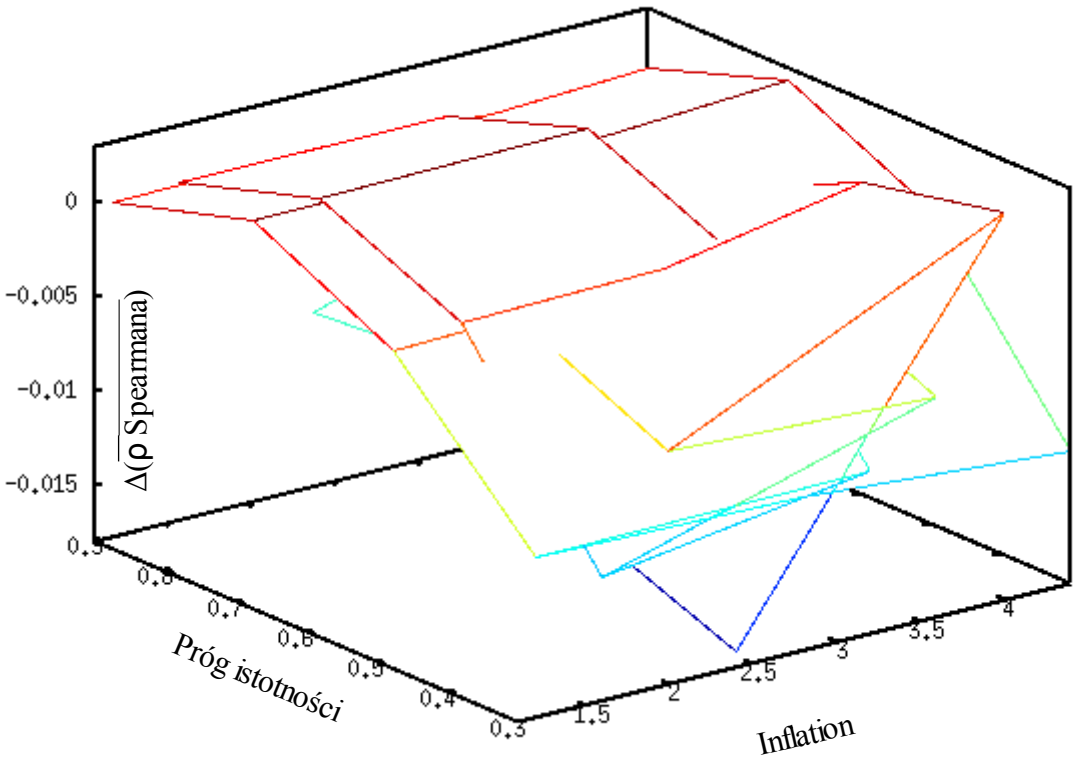
Wykresy zależności średnich miar *P@n* i *R@n* oraz współczynników Kendalla i Spearmana od parametrów algorytmu rerankingu: progu istotności i *inflation*



Rysunek 1: Wykres zależności różnicy średnich kompletności porządkowych R@n od przyjętych parametrów grupowania DAC: progu istotności i inflation (na podstawie tabeli 6.4 s. 111)



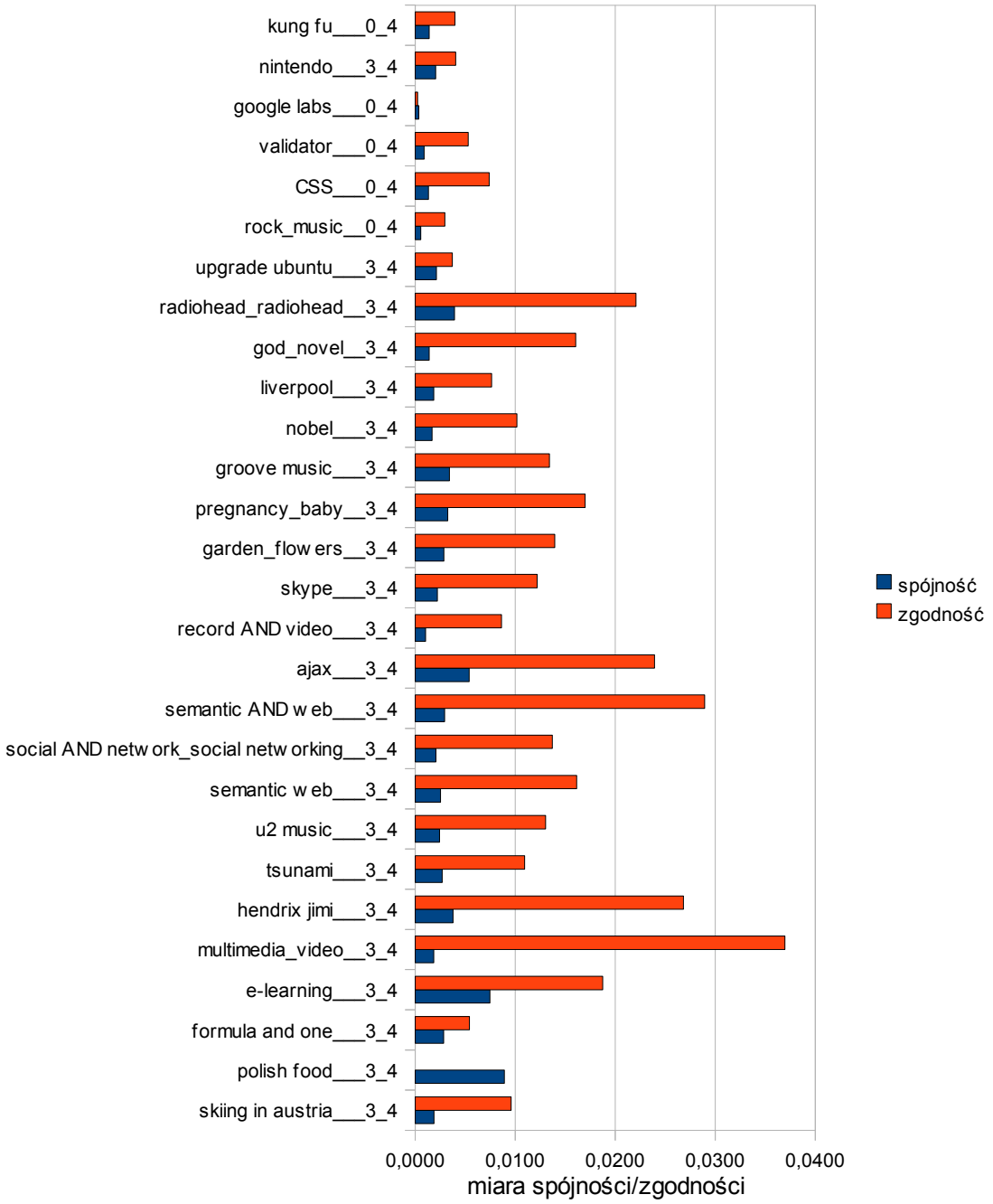
Rysunek 2: Wykres zależności różnicy średnich współczynników korelacji Kendalla od przyjętych parametrów grupowania DAC: progu istotności i inflation (na podstawie tabeli 6.4 s. 111)



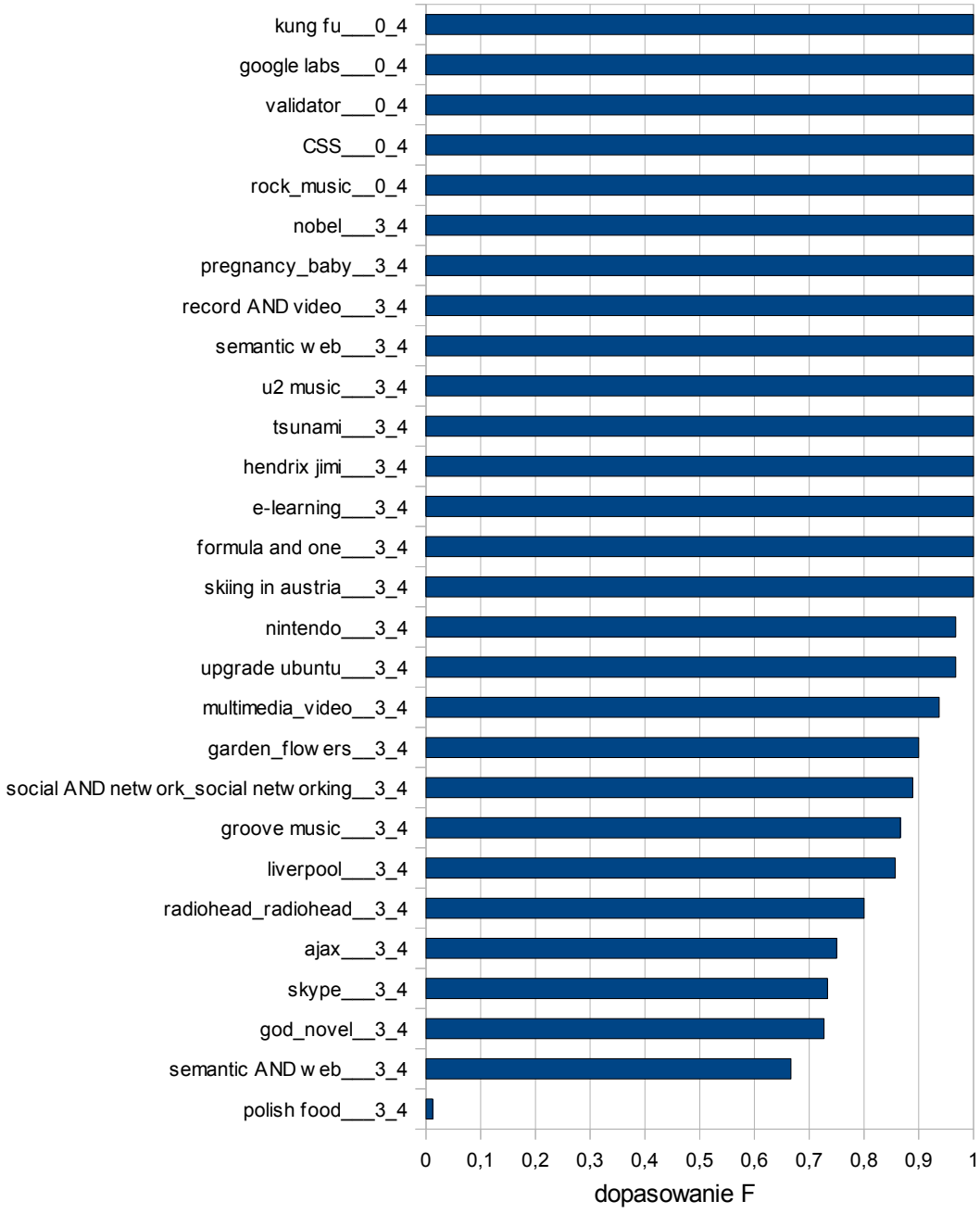
Rysunek 3: Wykres zależności różnicy średnich współczynników korelacji Spearmana od przyjętych parametrów grupowania DAC: progu istotności i inflation (na podstawie tabeli 6.4 s. 111)

Dodatek E

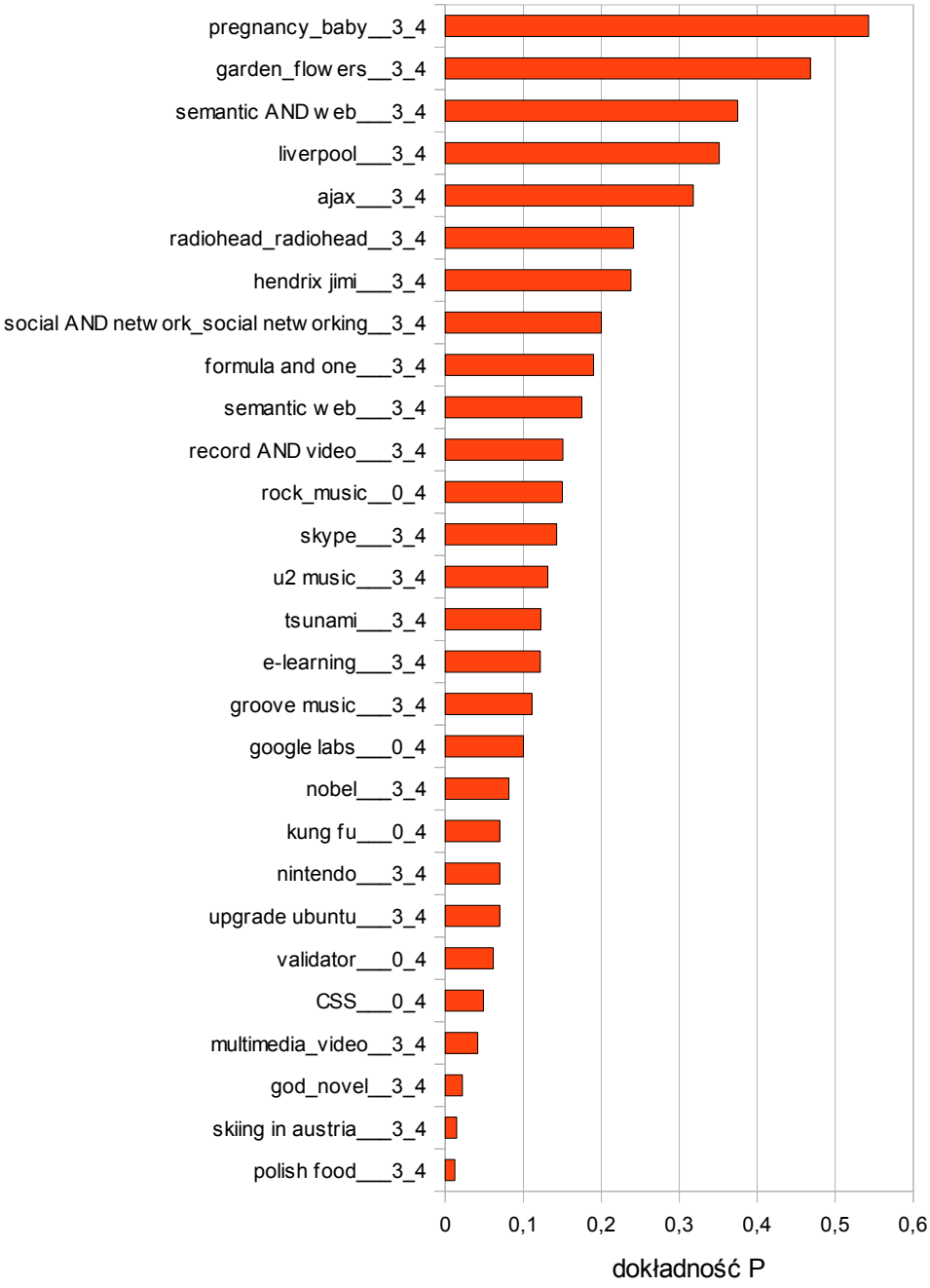
Dodatkowe wykresy danych zebranych podczas testów weryfikacji empirycznej



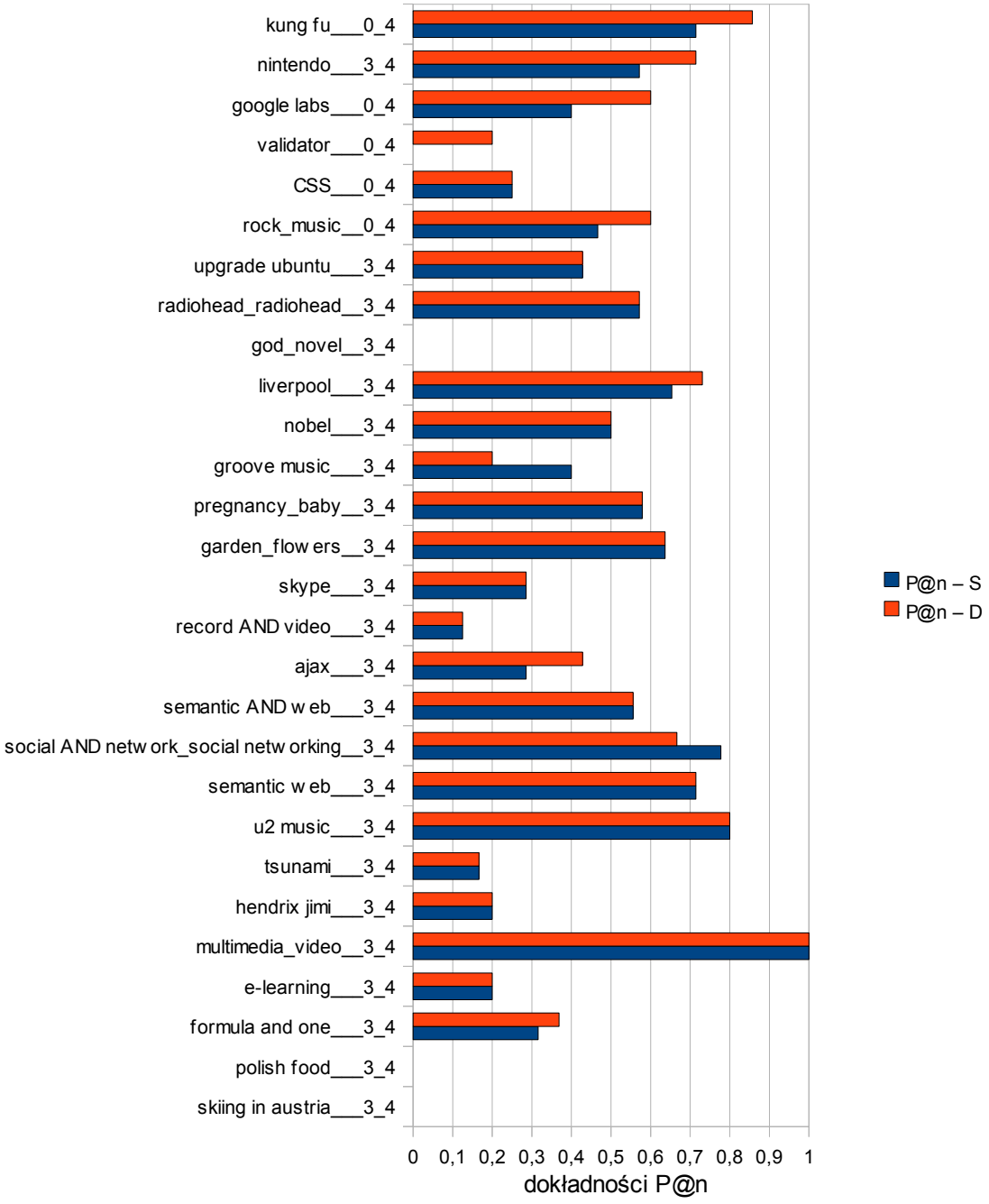
Rysunek 1. Wykres porównujący wielkości miar spójności i zgodności z tabeli 6.2 s. 103



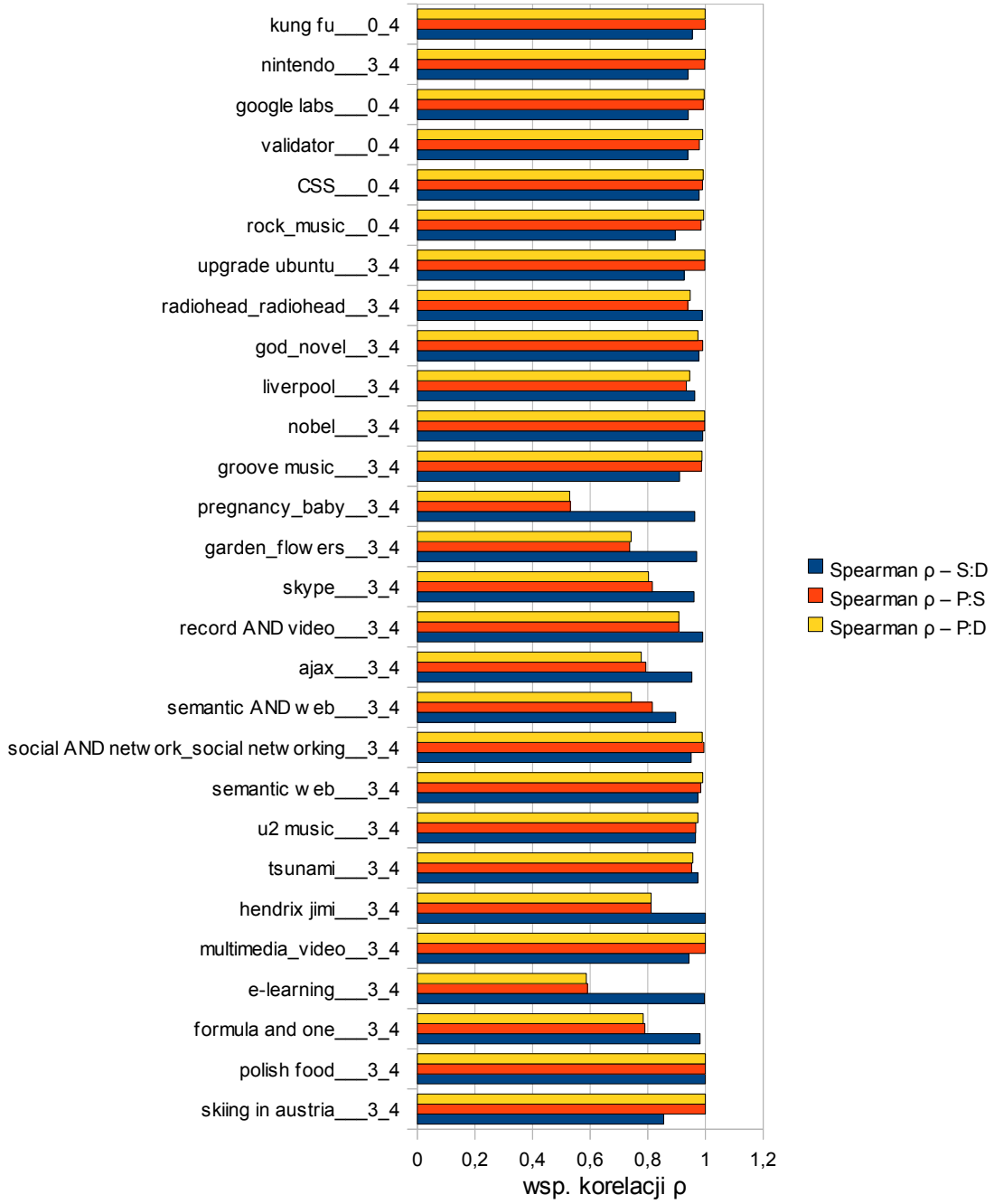
Rysunek 2: Wykres posortowanych malejąco wartości dopasowania F z tabeli 6.2 s. 103



Rysunek 3: Wykres posortowanych malejąco wartości dokładności P z tabeli 6.2 s. 103



Rysunek 4. Wykres porównujący dokładność P@n dla rankingów Solr i DAC z tabeli 6.2 s. 103



Rysunek 5. Wykres porównujący współczynniki korelacji Spearmana w tabeli 6.3 s. 107