Katedra Automatyki Akademii Górniczo - Hutniczej

Praca Dyplomowa

Budowa słownika frekwencyjnego na bazie polskojęzycznych stron internetowych

Bartosz Zarzyczny
pod kierunkiem: dr Adriana Horzyka

Kraków, wrzesień 2004

Spis treści:

1. Wstęp

Coraz większa ilość przenośnych urządzeń i automatów komunikujących się z operatorem powoduje konieczność opracowania ograniczonych słowników umożliwiających porozumienie się z użytkownikiem. Słowniki te muszą być ograniczone pod względem ilości słów. Powodem tego jest to, że komunikacja jest dla nich dodatkową funkcją, a nie podstawowym procesem. Bardzo dużą popularnością cieszą się obecnie słowniki w telefonach komórkowych wspomagające szybkie pisanie krótkich wiadomości tekstowych (sms) – niestety zawarte w nich słowa nie zawierają najpopularniejszych fraz z języka potocznego, który dynamicznie się zmienia – powoduje to konieczność dopisywania nowych słów (o ile telefon udostępnia taką funkcję).

Celem pracy jest stworzenie słownika frekwencyjnego (słownika przedstawiającego ilość wystąpień danego wyrazu) na podstawie polskojęzycznych stron internetowych. W tym celu został wykonany program analizujący polskie strony internetowe, a wyniki tych analiz są zapisywane w bazie danych. Jako platformę służącą do stworzenia pająka (programu analizującego, *z ang. crawler, spider*) wybrano PHP (*Hypertext Preprocessor*) – język służący do tworzenia stron internetowych oraz bazę danych MySQL. Zarówno PHP jak i MySQL są projektami OpenSource (o ogólno dostępnym kodzie źródłowym) umożliwiającymi uruchomienie projektu na najbardziej popularnych platformach systemowych (np. Linux, Windows) i natychmiastowej prezentacji wyników w sieci Internet.

Efektem pracy jest aplikacja, która jednokrotnie odwiedziła kilkanaście tysięcy stron internetowych i dokonała ich analizy w celu stworzenia słownika frekwencyjnego. Wynikiem działania aplikacji będzie baza danych zawierająca wszystkie polskie słowa (także te, które w świetle zasad ortografii zawierają błędy, lecz mogą być wykorzystywane w potocznym języku) z odwiedzonych stron internetowych. Ubocznym efektem będzie baza adresów odwiedzonych stron internetowych, które zostały poddane analizie.

Praca została podzielona na pięć rozdziałów:

Rozdział 1 - wstęp zawiera krótkie przedstawienie zakresu pracy.

Rozdział 2 - sposoby pozyskiwania informacji w internecie - prezentuje tematykę wyszukiwania informacji w sieci i sposoby jej pozyskiwania.

Rozdział 3 - budowa własnej wyszukiwarki – opis, w jaki sposób tworzy się wyszukiwarkę internetową, jakie elementy powinna zawierać.

Rozdział 4 - realizacja - projekt wyszukiwarki.

Rozdział 5 - zakończenie - podsumowanie pracy, wnioski.

2. Sposoby pozyskiwania informacji w Internecie.

2.1. Podstawowe informacje.

Dzięki Internetowi świat staje się globalną wioską, w której wszystko jest w zasięgu ręki. Niestety ta wioska jest bardzo zaśmiecona przez różnego rodzaju informacje. W tym momencie mamy (wg. danych otrzymanych z Google.com) ponad cztery miliardy stron internetowych. W takim kotle ciężko jest znaleźć jakiekolwiek informacje. Stanowi to całkiem realny problem, przed którym staje każdy użytkownik Internetu.

Rozwiązaniem tego problemu są różnego rodzaju metody i narzędzia umożliwiające oraz ułatwiające przeszukiwanie zasobów internetowych. Możemy natrafić na setki różnego rodzaju narzędzi, które możemy podzielić na podstawowe grupy [1]:

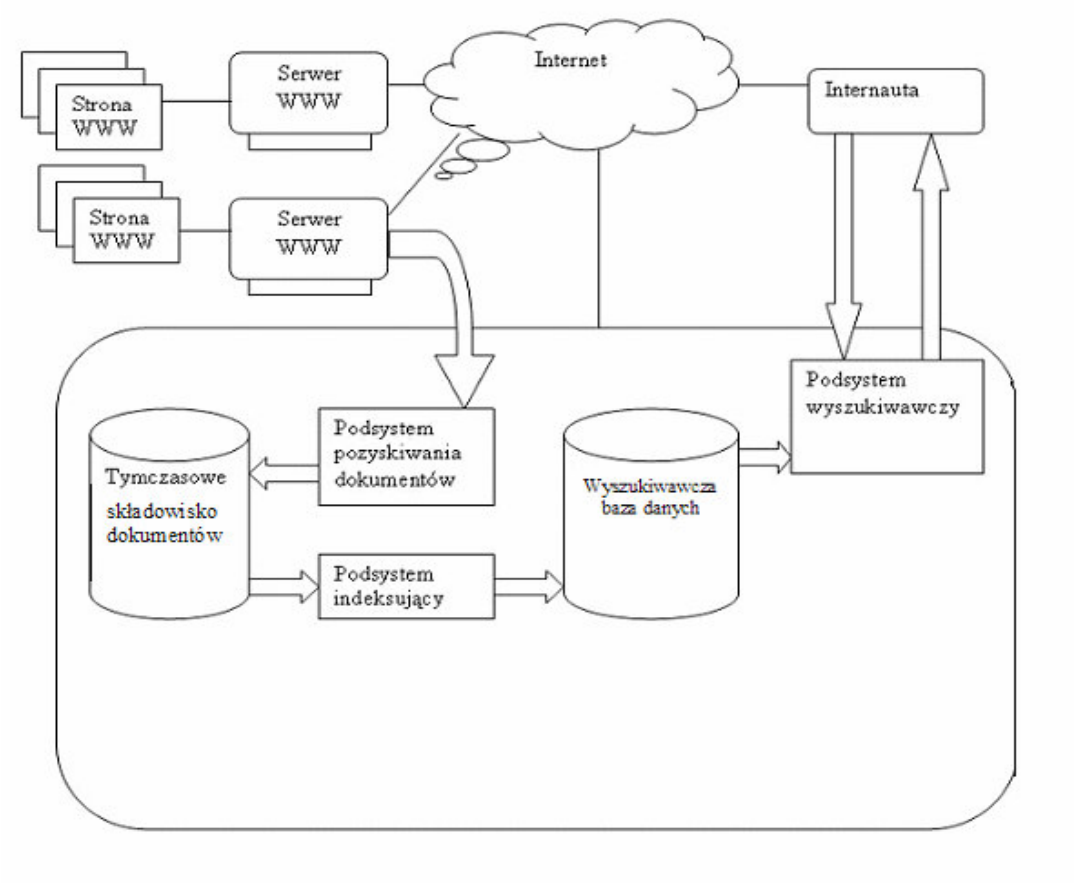
- wyszukiwarki
- metawyszukiwarki
- portale, wortale
- kolekcje odnośników do innych stron internetowych
- katalogi stron
- osobiste narzędzia wyszukiwawcze
- „oświecone zgadywanie”

- rozwiązania mieszane (połączenie dwóch lub więcej wyżej wymienionych, np. najpierw poszukiwanie w katalogu, potem zapytania do wyszukiwarek)

Taki podział może wydawać się oczywisty, lecz zazwyczaj jest on ignorowany przez większość internautów, a stanowi jedną z podstawowych informacji, gdy chcemy świadomie i efektywnie wyszukiwać informacje.

Systemy wyszukiwawcze składają się z:

- Podsystemu pozyskiwania dokumentów
- Podsystemu indeksującego
- Podsystemu wyszukiwawczego
- Wyszukiwawczej bazy danych
- Tymczasowego składowisko dokumentów



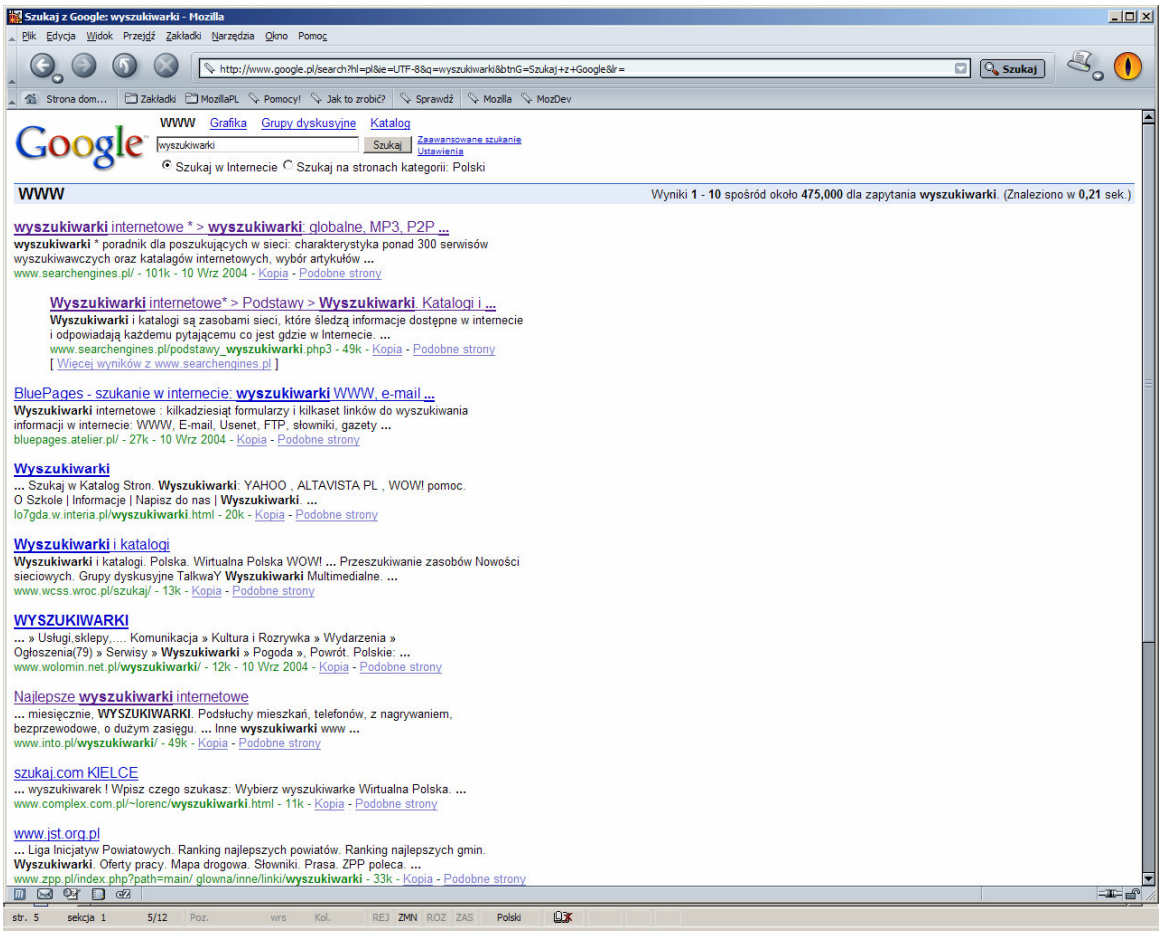
Rysunek 2.1: Budowa systemu wyszukiwawczego [1]

2.2. Wyszukiwarki.

W obecnej chwili najlepsze możliwości wyszukiwania i to ze względu na zakres, relatywność i szybkość wyszukiwania mają wyszukiwarki internetowe. Komunikacja z nimi odbywa się za pomocą słów kluczowych. Wyszukiwarka przeszukuje podane słowa lub słowo w bazie a następnie wyświetla wynik w postaci listy adresów url (ang. *Uniform Resource Locator* - zunifikowany format odnośników do zasobów), zazwyczaj z krótkim opisem pobranym ze strony lub wpisanym przez redaktorów. Zapytania do bazy danych mogą składać się z sekwencji słów kluczowych, wyrażeń logicznych lub z zapytań w języku naturalnym.

Wyszukiwarki obecnie mają znaczną przewagę nad katalogami. Największe wyszukiwarki ogólnego przeznaczenia posiadają ponad miliard linków (np. http://www.google.pl ponad cztery miliardy), zaś najmniej znaczące wyszukiwarki - co najmniej 50 milionów, w porównaniu z największymi katalogami jest to przynajmniej kilkadziesiąt razy więcej. Wyszukiwarki są także szybsze od katalogów, w których użytkownik musi wędrować poprzez kilka podkatalogów nim znajdzie interesujące go dokumenty. Niestety wyszukiwarki swoją szybkość zyskują kosztem relatywności dokumentów odnoszących się do wyszukiwanego hasła - tutaj ujawnia się jedyna i podstawowa przewaga katalogów nad wyszukiwarkami. Wyszukiwane dokumenty w katalogach są recenzowane i opisywane ludzką ręką. Jednakże i tutaj wyszukiwarki zaczynają konkurować: firmy utrzymujące katalogi nie są w stanie recenzować masy nowych dokumentów ani weryfikować ich aktualności wystarczająco często. Przez to jakość wyników wyszukiwania w katalogach pogarsza się. Wyszukiwarki za to polepszają jakość swych odpowiedzi dzięki zastosowaniu inteligentnych algorytmów wyszukiwania, wykorzystywaniu coraz większych oraz coraz aktualniejszych baz danych.

Do najbardziej znanych wyszukiwarek należą: Google (http://www.google.pl), AltaVista (http://www.altavista.com) i Infoseek (http://www.go.com).



Rysunek 2.2. Przykład wyszukiwarki [5]

2.3. Metawyszukiwarki.

Nawet najlepsza wyszukiwarka nie pokrywa więcej niż 30 - 40% stron WWW w sieci [1]. Dzisiejszy Internet szacuje się na 24 miliardy stron w części dostępnei i na ponad 14 bilionów w części ukrytej (tzw. Ukryty Internet). Bazy danych poszczególnych wyszukiwarek pokrywają się tylko częściowo, a algorytmy wyszukiwarek są tak różne, że wykonanie tego samego zapytania daje odmienne rezultaty. Baza danych wyszukiwarki może być nie tylko zbyt mała, ale i proces uaktualniania może odbywać się zbyt rzadko - AltaVista odświeża bazę

co 10 dni, podczas gdy Lycros jest uaktualniany co kilka godzin. Kolejną wadą może być też zbyt wolno działający program do pobierania danych - Excite potrzebuje około 28 dni na skompletowanie danych. Widać więc, że jeżeli będziemy korzystać z kilku różnych wyszukiwarek możemy dostać diametralnie różne wyniki. Dlatego dobrym pomysłem wydawało by się stworzenie nowych aplikacji WWW - multiwyszukiwarek (metawyszukiwarek), które wysyłają to samo zapytanie do kilku wyszukiwarek i przedstawiają wyniki na jednej stronie. Z punktu widzenia internauty (użytkownika) nie widać różnicy w działaniu, jednak multiwyszukiwarki nie posiadają własnej bazy danych, lecz korzystają z usług kilku innych zwykłych wyszukiwarek. Najczęściej metawyszukiwarki łączą wyniki wyszukiwania, dodając własne oceny i grupując dokumenty.

Rozróżniamy trzy typy multiwyszukiwarek:

- serwis typu “lista”
- serwis poszukujący równolegle
- serwis poszukujący pojedynczo.

Serwis typu “lista” to strona WWW ze zgromadzonymi na niej odnośnikami do wybranych wyszukiwarek. Takie rozwiązani jest wygodne, gdyż łącząc się z jednym adresem możemy przeszukać kilka serwisów. Inną zaletą jest fakt, że twórcy takich serwisów potrafią zgromadzić na jednej “liście” kilkaset adresów różnych wyszukiwarek i katalogów. Serwisy tego typu nie są uważane za prawdziwe metawyszukiwarki. W rzeczywistości nie oferują zaawansowanych metod wyszukiwania, nawet wyniki są przedstawiane w postaci właściwej dla wykorzystywanej aktualnie wyszukiwarki.

Serwisy przeszukujące pojedynczo to multiwyszukiwarki oferujące zazwyczaj jedno pole do wpisania słów kluczowych - tak jak w zwykłych wyszukiwarkach. Użytkownik ma możliwość wybrania wyszukiwarki względnie katalogu, w których zostaną wykonane zapytania. Wyboru

dokonuje się poprzez zaznaczenie odpowiednich pól lub wybieranie z listy poszczególnych wyszukiwarek. Proces wyszukiwania odbywa się po kolei, tzn. w określonym czasie multiwyszukiwarka łączy się i pobiera dane tylko z jednej wyszukiwarki lub katalogu. Uzyskane w ten sposób wyniki są odpowiednio porządkowane i wyświetlane przy zachowaniu podziału na poszczególne wyszukiwarki. Przykładowo jeżeli korzystamy z Google, InfoSeeka i AltaVisty to w oknie przeglądarki zobaczymy pierw odnośniki z Google, później z InfoSeeka, a na końcu AltaVista. Wyniki będą odpowiednio uporządkowane i przedstawione na jednej stronie.

Największą wadą powyższych typów metawyszukiwarek jest to, że jeśli trafimy na wolny lub obciążony serwer, będziemy musieli czekać, dopóki ów serwer nie prześle swoich odnośników do multiwyszukiwarki. Po otrzymaniu wyników z serwisu nastąpi dopiero połączenie z kolejnym.

Ostatnim typem multiwyszukiwarek są serwisy przeszukujące równolegle - są to metawyszukiwarki w pełnym znaczeniu tego słowa. Wyszukiwanie w nich jest identyczne jak w przypadku klasycznych wyszukiwarek (jedno pole do wpisania słów kluczowych), lecz w efekcie otrzymujemy dużą liczbę odnośników pochodzących z kilku lub nawet kilkunastu serwerów. Wyniki wyszukiwania przychodzą na ekran bardzo szybko i nie ma konieczności czekania, aż załadują się wszystkie wyniki, możemy od razu przystąpić do ich przeglądania. W tym czasie nowe wyniki będą stopniowo doczytywane. Dzieje się tak, ponieważ multiwyszukiwarki tego typu łączą się z kilkoma serwerami równocześnie i na bieżąco pobierają dane.

Najlepiej właśnie jest korzystać z multiwyszukiwarek równoległych - zapewniają one najwyższy komfort użytkowania i największą szybkość.

Zaletą multiwyszukiwarak jest przede wszystkim duża szybkość (wyniki otrzymujemy prawie natychmiast) przy możliwości otrzymania reprezentatywnych wyników (dzięki dostępowi do większej bazy

danych). Wadą jest to, że można zadawać tylko proste pytania z uwagi na różnorodność dostępnych opcji, syntaktyki i semantyki bardziej zaawansowanych zapytań w różnych wyszukiwarkach. Najpopularniejsze multiwyszukiwarki to: Debriefing (http://www.debriefing.com), Dogpile (http://www.dogpile.com) i polskie: EMULTI (http://www.emulti.pl) oraz Ithaki (http://www.ithaki.net).

Rysunek 2.3. Przykład multiwyszukiwarki [6]

2.4. Portale, wortale i strony startowe.

Są pomyślane jako punkty wyjściowe dla internautów. Firmy posiadające portale wiedzą, że dla uzyskania sukcesu komercyjnego muszą przyciągać ogromną liczbę odwiedzających. Podstawą ich sukcesu są informacje na nich zawarte: skróty wiadomości, pogoda,

horoskopy, bardzo często darmowe konta pocztowe, forum dyskusyjne, chaty, blogi, darmowe miejsce na publikowanie stron WWW.

Właściwie wszystkie portale zawierają katalogi stron (Onet, Interia), linki do oprogramowania, wyszukiwarki internetowe. Wyszukiwarki internetowe są najczęściej obciążone komercyjnie – wyniki wyszukiwania zależą bardziej od sponsora niż od zawartości. Pewnego rodzaju wyjątkiem może być tu http://www.onet.pl, który wprowadził boksy reklamowe w celu uwolnienia swojej wyszukiwarki od obciążeń komercyjnych.

Coraz częściej pojawiają się specjalistyczne portale internetowe zajmujące się konkretnymi dziedzinami: architekturą (http://www.architekci.pl), rolnictwem (pierwszy portal rolniczy: http://www.ppr.pl), ekonomią (http://nbportal.pl). Dają one możliwość znalezienia od razu informacji interesujących internautę. Wiadomości w nich zawarte są związane z tematyką portalu i nie wykraczają poza tematykę portalu.

Rysunek 2.4. Przykład wyszukiwarki [7]

2.5. Kolekcje linków.

Kolekcje linków nie wymagają zbyt dużego zaangażowania czasu oraz wysiłku – wystarczy opracować odpowiednie skrypty. Administrator strony podaje adres URL swojej strony, jego zgłoszenie jest automatycznie przetwarzane i dopisywane na końcu listy. Nie ma przeprowadzanej kategoryzacji, grupowania, opisywania czy recenzowania – akceptowana jest każda strona. Kolekcje były popularne w latach 1997 – 1998. Były wykorzystywane przez internautów do odwiedzania nowych stron pojawiających się w sieci, ponieważ umieszczenie strony w wyszukiwarce trwało co najmniej tydzień a w katalogu jeszcze dłużej. Obecnie popularność stron zawierających kolekcję linków jest bliska zeru. Głównym powodem jest pojawienie się oprogramowania automatycznie dodającego adresy stron do wielu wyszukiwarek jednocześnie. Z uwagi na łatwość zgłaszania z użyciem tego oprogramowania w kolekcjach linków pojawia się tysiące stron dziennie – nikt nie jest w stanie ich przeczytać.

2.6. Katalogi stron.

Katalog jest drzewiastą strukturą, w którą się zagłębiamy w czasie wyszukiwania. Strony są przypisane ze względu na swoją zawartość do poszczególnych kategorii, które dzielą się na kategorie podrzędne. W kategoriach elementarnych internauta znajdzie bezpośrednie hiperłącza do interesujących go stron, należących do danej kategorii.

Katalogowaniem witryn internetowych zajmują się redaktorzy, którzy przeglądają poszczególne strony i decydują o wpisaniu ich lub nie do bazy danych. Po zakwalifikowaniu ustalają w jakiej kategorii

tematycznej należy daną stronę umieścić, zazwyczaj też umieszczają krótki opis zawartości.

Redagowaniem katalogów zajmują się ludzie – pracownicy firmy prowadzącej katalog bądź osoby z zewnątrz (np. ochotnicy) , albo też udostępniane są specjalne formularze do dodawania strony. W takim wypadku to administrator strony WWW kataloguje swoje strony.

W ten sposób powstaje spis stron WWW, który ma postać katalogu o drzewiastej strukturze, podzielonego na kategorie główne, które dzielą się na kategorie podrzędne i tak dalej. To wszystko jest przedstawione w wygodnej do przeszukiwania formie hipertekstowej, gdzie na końcu struktury znajdują się odsyłacze do konkretnych stron WWW. Najczęściej taki katalog jest przeszukiwany poprzez stopniowe zagłębianie się strukturę katalogów, aż do momentu uzyskania zadowalających efektów.

Zaletą katalogów jest to, że ich zawartość jak i struktura jest przygotowana przez ludzi – dlatego jest bardziej zrozumiała, a odpowiedzi na zapytania są bardziej relatywne i łatwiej jest znaleźć odpowiedzi na ogólne zapytania.

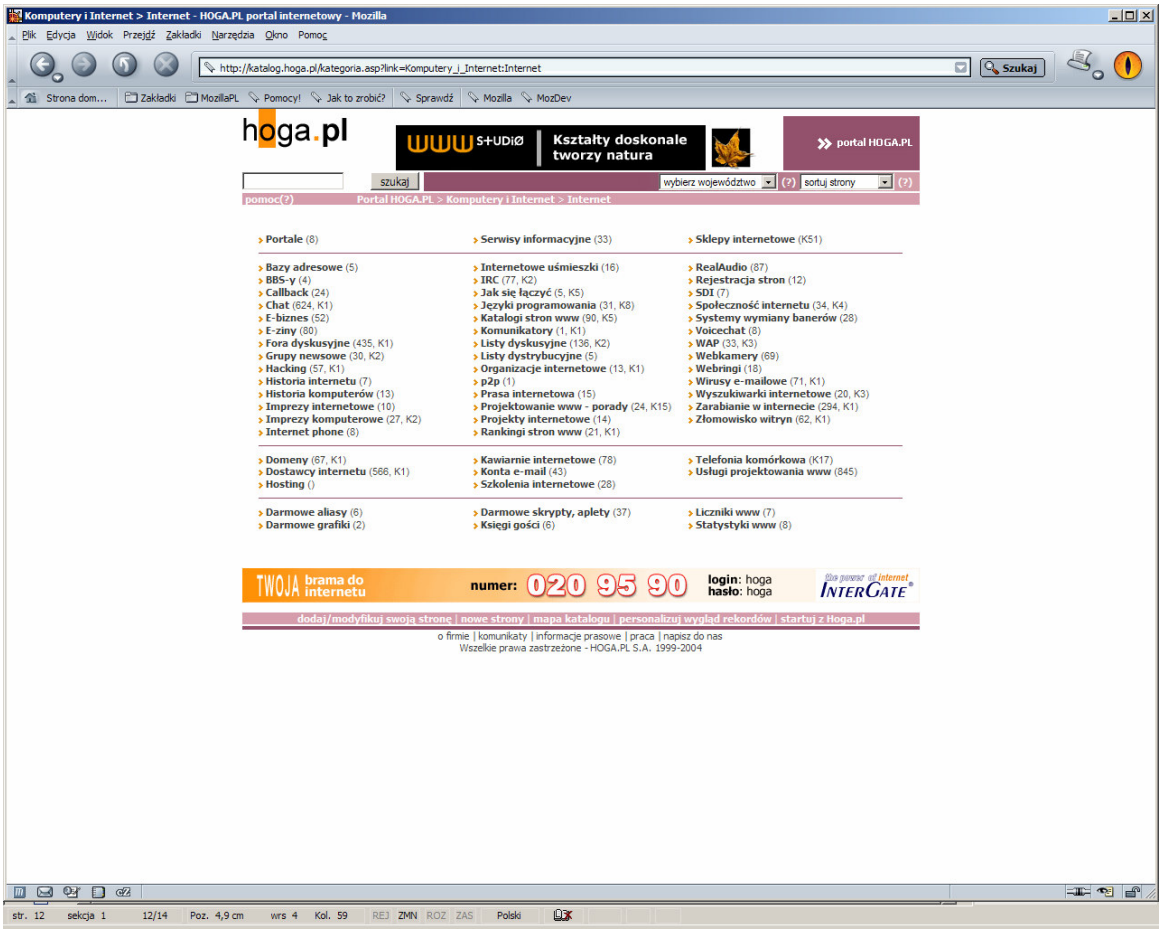
Z drugiej strony może być to też wadą katalogów. Nie zawsze można polegać na opiniach (wynikach) redakcji, gdyż strony są przydzielane do poszczególnych kategorii na podstawie subiektywnych kryteriów oceniających. Mogą znaleźć się tacy użytkownicy, którzy mogą mieć inne spojrzenie na dane zagadnienie i szukać go w innej kategorii, niż ta, do której zostało zakwalifikowane przez redaktora.

Największym problemem katalogów jest ich aktualność. Zmiana, usunięcie strony zarejestrowanej w katalogu nie ponosi za sobą automatycznej aktualizacji, aż do następnych odwiedzin przez redaktora lub zmiany danych w formularzu przez administratora danej strony WWW. Przy bardzo szybkiej zmienności zasobów informacyjnych staje się to bardzo dużym problemem i zmusza redaktorów do wypracowania

metod weryfikacji informacji zawartych w bazie. Jest to głównym powodem katalogowania niewielkiej liczby stron.

Kolejną wadą jest nienadążanie za rozwojem Internetu. Szacuje się, że tylko 0.25% wszystkich stron WWW zostało w ten sposób skatalogowanych.

Przykładami katalogów mogą być: Yahoo!, Dmoz, czy polska Hoga.

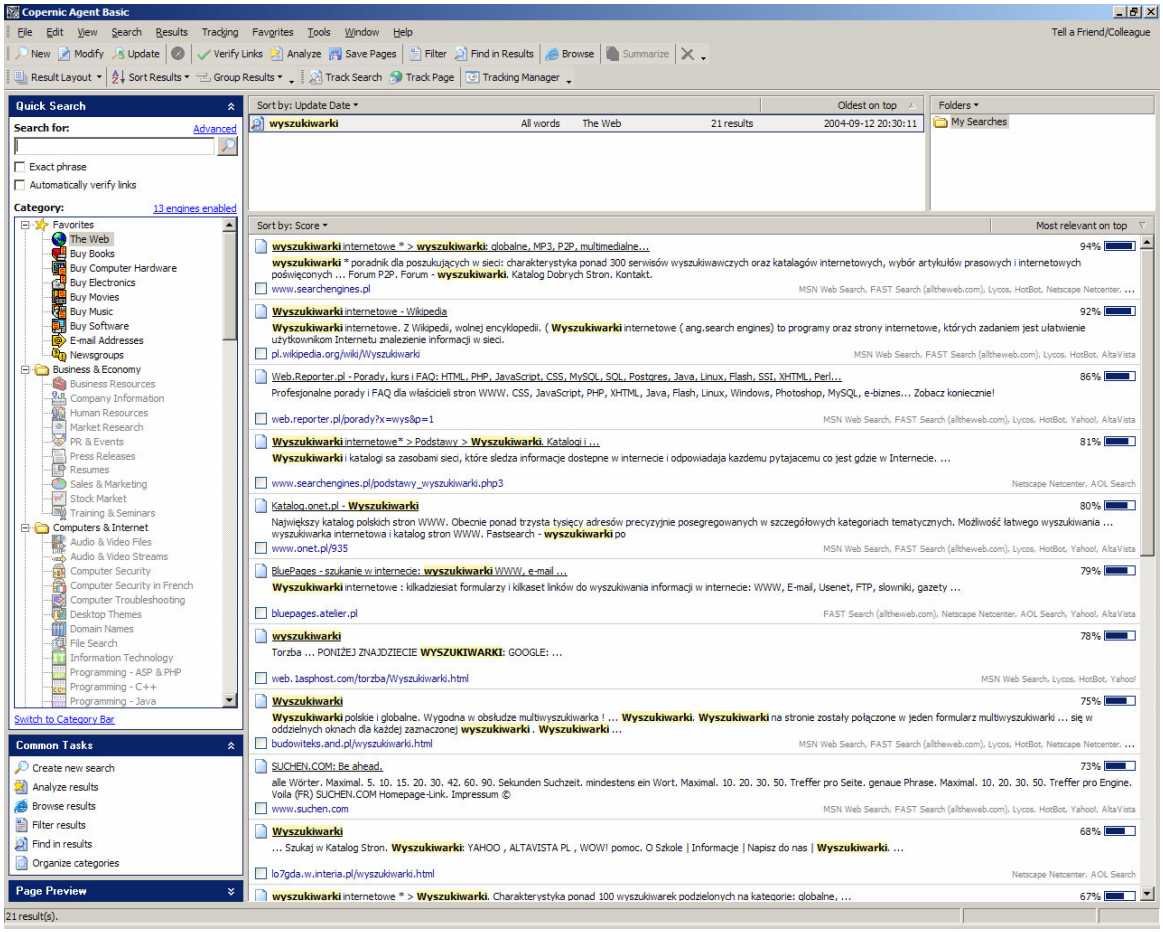


Rysunek 2.6. Przykład wyszukiwarki [8]

2.7. Osobiste narzędzia wyszukiwawcze.

Osobiste narzędzia multiwyszukiwawcze to atrakcyjne rozwiązanie dla naukowców. Zwykle nie są oni zainteresowani znalezieniem odpowiedzi na jedno specyficzne pytanie, lecz dogłębnym zbadaniem określonego obszaru. W takich wypadkach dobrze jest stworzyć i rozwijać swe własne

przedmiotowe bazy danych. Rozwiązania takie jak narzędzia osobiste oferują takie możliwości. Dają one możliwość eliminacji duplikatów dokumentów, ich automatycznej kontroli pod kątem poprawności adresu dokumentu oraz umożliwiają usuwanie dokumentów nierelewantny z baz danych. Czołowi producenci w tej dziedzinie (Copernic, Inforian Quest) włączyli setki wyszukiwarek do swego oprogramowania grupując je według profilu i dając możliwość naukowcom tworzenia własnych grup wyszukiwarek (przez dołączanie / usuwanie z listy). Nowa generacja osobistych narzędzi multiwyszukiwawczych poszerza zakres usług. Zbierana jest lista linków, każdy link jest sprawdzany, dokumenty są ściągane i analizowane. Jeśli dokument jest nierelewantny, jest usuwany, a jeśli jest przydatny, jest dodawany do bazy dokumentów. Po zakończeniu przetwarzania on - line zaczyna się przetwarzanie off - line: ustalanie stopnia relewancji czy też poszukiwanie odpowiedzi na pytanie. Wytwórcy twierdzą, że stosują metody sztucznej inteligencji podczas procesu analizy i rankingowania, a oprogramowanie ma zdolność samouczenia się. Osobiste narzędza multiwyszukiwawcze bez wątpienia stosują najbardziej zaawansowane technologie, ponieważ nie ma nacisku na czas przetwarzania, mogą więc być wyposażone w bardziej zaawansowane metody analizy i pracować z pełnymi dokumentami, a nie ich reprezentacją.



Rysunek 2.7. Okno w Copernicusie

2.8. „Oświecone zgadywanie”.

„Oświecone zgadywanie” jest jednym z najprostszych sposobów na znalezienie informacji w Internecie. Jego zastosowanie jest niestety ograniczone – ze względu na ograniczoną liczbę domen nie zawsze możemy znaleźć interesujące nas informacje. Polega ono na odwiedzaniu adresów składających się z poszukiwanych wyrazów. Jest za to najszybszym sposobem znalezienia firm lub popularnych produktów – np. firma Carrefour: http://www.carrefour.pl, Nokia: http://www.nokia.pl,

3. Budowa wyszukiwarki internetowej.

3.1. Wyszukiwarki.

W świecie wyszukiwarek WWW ogólnego przeznaczenia, typu Google, Yahoo itd. o ogromnych zasobach obserwuje się ostrą walkę konkurencyjną, należy oczekiwać, że być może osiągniemy stan monopolu. Jednak nawet małe wyszukiwarki, wyspecjalizowane w określonym zakresie, mają szansę znaleźć zainteresowanie wśród specjalistów z danej dziedziny. Wielką ich przewagą jest niezaśmiecanie odpowiedzi na specjalistyczną kwerendę dokumentami mającymi przypadkowy związek ze słowami, które zostały zawarte w zapytaniu. Dlatego wiele uniwersytetów prowadzi prace nad własnymi wyszukiwarkami na potrzeby własnego środowiska, a firmy spoza branży internetowej są zainteresowane zakupem takiej wyszukiwarki. Takie firmy i instytucje zazwyczaj posiadają małe zasoby komputerowe i dlatego zakup dużych gotowych systemów jest dla nich nie do zaakceptowania.

Generalnie coraz więcej organizacji tworzy wyszukiwarki dla własnych stron internetowych. Oczywiste jest, że taki lokalne wyszukiwarki są finansowane w ograniczony sposób a ich moc obliczeniowa jest stosunkowo mała.

3.2. Zadanie pająka.

Zadaniem pająka jest ściągnięcie z Internetu jak największej ilości danych na dysk twardy komputera (serwera) celem dokonania analizy zawartości plików. Dane są analizowane pod względem zawartej treści oraz linków, które pająk wykorzystuje w dalszych poszukiwaniach. Linki niepożądane (np. adresy url do plików graficznych, skryptów, itp.) powinny być usuwane przez program.

3.2.1. Budowa pająka.

Pająk można zbudować w oparciu o gotowe elementy. Na przykład Acme.Spider (http://www.acme.com/java/software/Acme.Spider.html) jest pająkiem udostępnianym w postaci biblioteki Javy, którą można dostosować do własnych potrzeb.

Lista przykładowych robotów dostępnych na WWW:

- Checkbot (Perl) http://degraaff.org/checkbot/
- DWCP (Java) http://www.dridus.com/~rmm/dwcp.php3
- Fluid Dynamics Software Corporation (Perl) http://www.xav.com/scripts/search/
- JoBo (Java) http://www.matuschek.net/software/jobo/

Lista pająków, które mogą być przydatne do budowy własnej wyszukiwarki jest dostępna pod adresem: http://www.robotstxt.org/wc/active/html/index.html

3.2.2.Punkty startowe.

Strony bogate w linki z danej dziedziny - tzw. hubs, są dobrym punktem startowym do przeszukiwania Internetu. Powinno dobierać się hubsy niepodobne do siebie. Przykładowymi dobrymi stronami startowymi w polskim Internecie są strony: http://www.onet.pl, http://www.interia.pl, http://www.prv.pl (bardzo duża ilość prywatnych i niekomercyjnych stron), http://www.friko.pl (podobnie jak serwis PRV z tym, ze udostępnia także przestrzeń dyskową).

3.2.3.Specjalizacja pająka.

Pająki zwykle zawierają miejsca, gdzie programiści mogą dołączyć kod umożliwiający odfiltrowanie własnych treści. Przykładowo można dodać odpowiednie filtry zapobiegające odwiedzanie stron CGI, plików

z grafiką, plików exe, zip, itp. Zwykle zdefiniowany jest cache zapobiegający kręceniu się pająka w kółko.

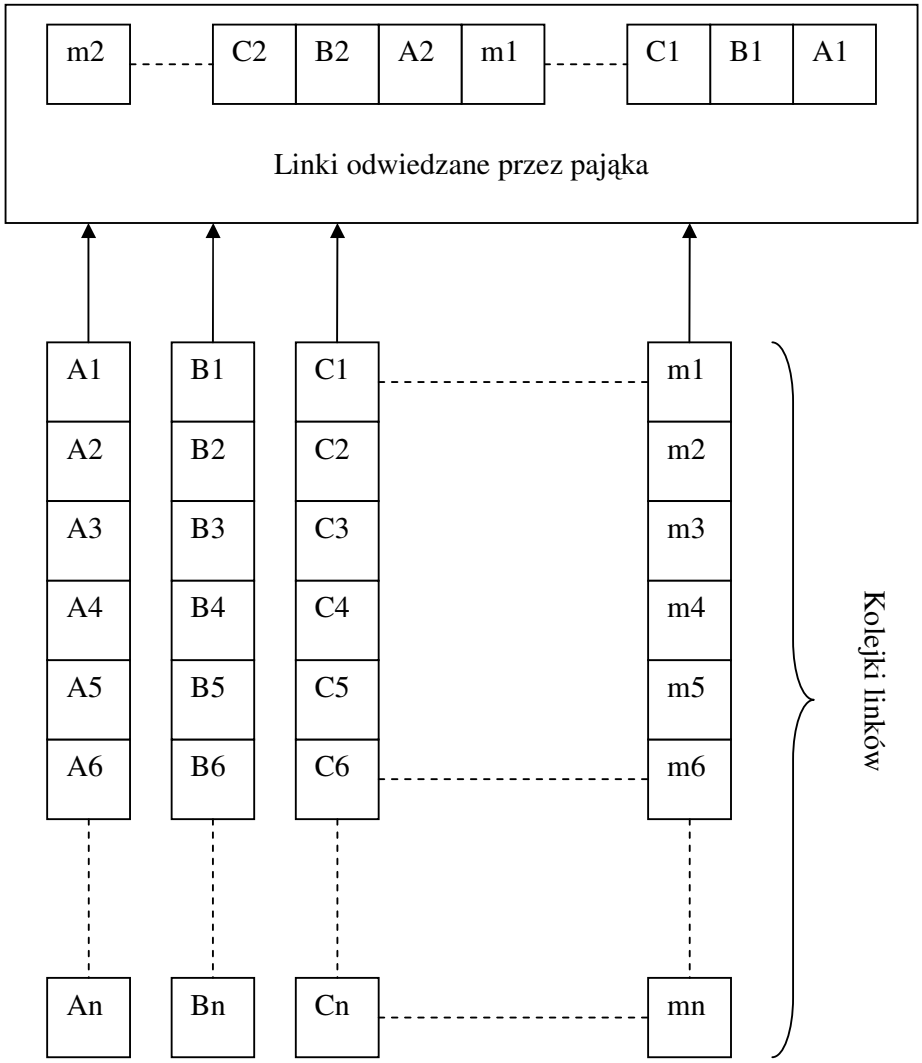
Pająk tworzy bazę adresów URL, które zamierza odwiedzić, poprzez ściągnięcie strony na dysk twardy serwera, wyodrębnieniu z niej linków i dodaniu ich do kolejki. Zajętość kolejki rośnie zwykle wykładniczo z głębokością przeszukiwania.

3.2.4. Częstotliwość odświeżania.

Częstotliwość odświeżania jest ograniczona szybkością pająka. Są pająki przetwarzające nawet ponad 25 stron na sekundę, ale zwykle prędkość przetwarzania nie jest wyższa niż 10 stron na minutę. W celu przetworzeniu około 100 000 stron komputer potrzebowałby przynajmniej 7 dni. Ponieważ komputer nie może wyłącznie pracować jako pająk, strony są odświeżane z reguły raz na 2 - 3 tygodnie. W wypadku najbardziej popularnej wyszukiwarki Google.com strony są odświeżane co 2 - 3 dni – zostało to sprawdzone na podstawi zapisów w logach kilku serwerów.

3.2.5. Bardzo zaawansowane pająki.

Zaawansowane pająki powinny być w stanie zarządzać nie tylko pamięcią, ale też procesorem oraz ruchem w sieci. Nie powinien zbyt często robić dostępu do tego samego serwera i przez to go blokować. Niektóre pająki przyjmują strategię kolejkowania – jeżeli liczba linków jest duża tworzą kilkaset kolejek, przy czym robot odwiedza po jednym linku z każdej kolejki. W ten sposób jeden serwer jest odwiedzany raz na kilkaset razy.



Rys. 3.1. Strategia kolejkowania linków.

Celem pracy pająka jest ściągnięcie z Internetu jak największej ilości dokumentów na własny serwer i przeanalizowanie ich pod względem treści, którą zawiera. Dokumenty mogą być analizowane i ocenione dopiero po ściągnięciu na serwer.

Strategie pająka:

- Błądzenie po stronach internetowych: na wejściu dostaje listę adresów, następnie przeszukuje je metodą kolejki priorytetowej.
- Poszukiwanie w innych bazach danych (*multi search*) np. google, yahoo. W tym przypadku trudność sprawia odpowiednie sformułowanie zapytań do baz danych. Konieczne jest stworzenie

„pijawek” (ang. *leech*) – programów specjalizujących się w pozyskiwaniu informacji z konkretnych serwisów. Gdy jedno zapytanie jest skierowane do kilku serwisów konieczne jest skorzystanie z komponentów tłumaczących określone zapytanie na kwerendy dla danych baz.

- Mieszana – najpierw poprzez przeszukiwanie innych baz danych (*multi serach*) ustali listę adresów, następnie wyszukuje metodą kolejki priorytetowej.

Głównym zadaniem “inteligentnego pająk” jest wstępne przewidywanie czy dokument może być interesujący przed jego ściągnięciem. Wstępna ocena zawartości strony może być dokonana na podstawie:

- krótkich informacji o danym dokumencie - jeśli pająk ma dostęp do dokumentu przez bazę danych (np. Google), to również posiada kilka zdań na temat tego dokumentu.
- opis linku do danego dokumentu w innym dokumencie w bazie pająka - dokumenty, po ściągnięciu, są analizowane i ekstrahowane są informacje o linkach do innych dokumentów. Na ogół linki są stowarzyszone z krótkimi opisami. Czasem można dużo się dowiedzieć o stronie na podstawie tych opisów.
- informacje wyciągane z adresu URL - nazwa serwera, nazwa pliku itp.

Inteligentny pająk może stosować jeden lub kilka z poniższych środków:

Właściwą ocenę przydatności dokumentu do głównego problemu systemu dokonuje wspomagający moduł oceniający, który wyekstrahuje ze ściąganego dokumentu potrzebne informacje. Praca inteligentnego pająka musi być zależna od modułu, który potrafi weryfikować, czy dany dokument jest interesujący czy nie.

Każdy inteligentny pająk musi posiadać pewną bazę wiedzy, która będzie swego rodzaju heurystyką dla procesu poszukiwania połączeń internetowych. Baza wiedzy powinna reprezentować wiedzę lingwistyczną kojarzącą napotkane w tekście pojęcia ze słowami kluczowymi, które nas interesują.

Jeżeli stworzymy platformę wielopająkową, to możemy założyć, że pająk pracujący w środowisku MAS (*Multi Agent Systems*), gdzie każdy z nich ma tylko lokalną bazę wiedzy. Można pokusić się o strategie ewolucyjne dla pająków, które wspólnie wyszukują optymalną strategię.

Głównym problemem dla pająków poszukujących po linkach jest posortowanie aktualnej listy adresów URL w taki sposób, aby na początku listy znajdowały się dokumenty, które z dużym prawdopodobieństwem zawierają poszukiwaną informację.

Baza wiedzy pająka winna zawierać listę słów i fraz kluczowych z wagami oraz reguły rozpoznawania interesujących stron. Adresy URL winny być posortowane względem sumy wag słów kluczowych w ich opisach.

Bazę wiedzy można wygenerować metodami systemów uczących się (systemów odkryć) klasyfikacji, jeśli dostępna jest (przygotowana przez człowieka) baza danych zawierająca oprócz treści dokumentów także decyzje, czy dokumenty jest relatywny czy nie, a także decyzję, czy dokument może zawierać linki do stron, które zawierają wyszukiwaną informację.

Pająk zadający pytania do innych wyszukiwarek może uzyskać listę adresów URL z innych wyszukiwarek poprzez zadawanie do nich odpowiednich pytań. Najważniejszym zadaniem dla tego typu pająków jest sformułowanie zapytań do innych znanych wyszukiwarek w celu uzyskania informacji. Informacja pająka polega na umiejętności

dobierania zapytań do odpowiednich wyszukiwarek w celu uzyskania najistotniejszych informacji.

3.3. Indekser dokumentów.

Indeksowanie to proces tworzenia ‘indeksu’, czyli specjalizowanej bazy danych zawierającej skomplikowaną wersję dokumentów ściągniętych przez pająka. Indeks winien być zoptymalizowany w celu szybkiego wyszukiwania listy dokumentów zawierających określone słowa bądź frazy (‘termy’).

Proces indeksowania składa się zasadniczo z następujących etapów:

- Identyfikacji słów, fraz, terminów występujących w dokumentach
- Usuwania słów popularnych
- Ekstrakcji tematów słów przy użyciu algorytmów szukającego tematu
- Zastąpienie tematów przez numeryczne identyfikatory wyrazów, słów (termów) indeksujących - w celu wydajniejszego przetwarzania
- Zliczanie wystąpień tematów (obliczanie tzw. *tf - term frequency*)
- Opcjonalne tworzenie fraz dla termów o wysokiej częstotliwości
- Obliczanie wag dla wszystkich prostych termów, fraz i klas tezaurusa - w oparciu o stosowany później model wyszukiwania
- Przypisanie każdemu dokumentowi przynależnych prostych termów, fraz i klas tezaurusowych z odpowiednimi wagami.

Budując indekser trzeba postarać się o parser stron HTML, jeśli chcemy indeksować inne dokumenty niż HTML, potrzebujemy także stosownych parserów zawartości lub konwerterów na pliki HTML.

Niezbędnym elementem indeksera jest analizator, który będzie sterował podziałem dokumentów na temy (słowa, czy też frazy).

Analizator może przykładowo tekst ‘The ill dohs’ rozbić na jednostki ‘the’, ‘ill’, ‘dogs’ (małe litery), podczas gdy inny może zamienić ten tekst na ‘ill’, ‘dog’ (małe litery, zamiana na liczbę pojedynczą i pominięcie pospolitego wyrazu ‘the’).

Słowa pospolite pomija się na bazie tzw. stop - listy. Dla języka angielskiego taka stop - lista zwykle obejmuje słowa: ‘a’, ‘and’, ‘are’, ‘at’ itp. Słowa pospolite mogą się różnić dla specjalizowanych zbiorów dokumentów. Np. w tekstach prawnych takimi słowami byłby ‘article’, ‘paragraph’ itd.

Często analizator może dodać i / lub zmienić słowa na ich tłumaczenie na określony język. Może dodać i / lub zamienić słowa na słowo pokrewne (notebook - laptop) synonimy (laptop - laptopik), pojęcia ogólne (notebook - komputer przenośny) i / lub pojęć zawężających (notebook - Centrino, mobile itd.) Trzeba tu oczywiście użyć odpowiedniego słownika, tezaurusa.

Wreszcie na podstawie częstego bliskiego współwystępowania i odpowiedniego słownika fraz analizator może zmienić grupę słów na term, np. mysz komputerowa itp.

Analizator może szerzej patrzeć na analizowany dokument i wiązać z wygenerowanymi termami dodatkowe atrybuty, takie jak częstość występowania w tekście, czy też to być może nazwa własna, czy term wystąpił w ważnym miejscu (tytuł dokumentu, nagłówek rozdziału, opis pliku) czy też w miejscu deprecjonowanym (np. jako notatka zapisana małymi literami itp.), czy wreszcie wagę termu, mierzącą jego ważność na tle występowania w całej populacji dokumentów.

Stąd blisko jest do generowania listy słów kluczowych dla dokumentu, a także automatycznego generowania streszczeń oraz automatycznego przydzielania do kategorii czy też grupowania.

Analizator musi być użyty zarówno przy indeksowaniu, jak i przy

wyszukiwaniu. Podczas indeksowania będzie przetwarzać dokumenty, podczas wyszukiwania: treść zapytania internauty.

Należy zwrócić uwagę na to, że proces indeksowania może kolidować z równoległym procesem wyszukiwania. Aby proces wyszukiwania działał poprawnie nawet w czasie indeksowania, proces indeksowania nie powinien zmieniać na bieżąco zawartości indeksu wykorzystywanego w trakcie wyszukiwania powinno się jednorazowo zastępować stary indeks indeksem nowym.

3.4. Zapytania.

Wyszukiwanie to operacja polegająca na identyfikacji podzbioru zbioru dokumentów, zawierających pożądaną treść lub mających pożądane cechy.

Proces odpowiadania na zapytania składa się zwykle z etapów:

- Pobranie od internauty słów z kwerendy
- Wyszukiwanie tematów tym samym algorytmem co na etapie indeksowania
- Zastąpienie tematów numerami termów indeksujących
- Obliczenie wag dla wszystkich termów z zapytania
- Tworzenie wektora zapytania (lub innej formy reprezentacji kwerendy, zależnie od modelu wyszukiwania
- Obliczenie stopnia podobieństwa między kwerendą a opisami dokumentów
- Zwrócenie listy dokumentów z podaniem rankingu

Na wejściu operacji wyszukiwania podaje się kwerendę (zapytanie), która specyfikuje kryteria wyboru dokumentów, a jej wynikiem jest lista dokumentów (trafień), która spełnia kryteria.

Lista trafień jest zwykle uporządkowana według pewnego

kryterium relewancji (rankingu) i może zawierać tylko podzbiór zbioru tych dokumentów, które faktycznie spełniają kryteria (najczęściej są to dokumenty o najwyższym rankingu).

Operacje wyszukiwarek przebiegają zwykle nie na faktycznych dokumentach, lecz na 'indeksie', który zawiera wcześniej przygotowane informacje o dokumencie.

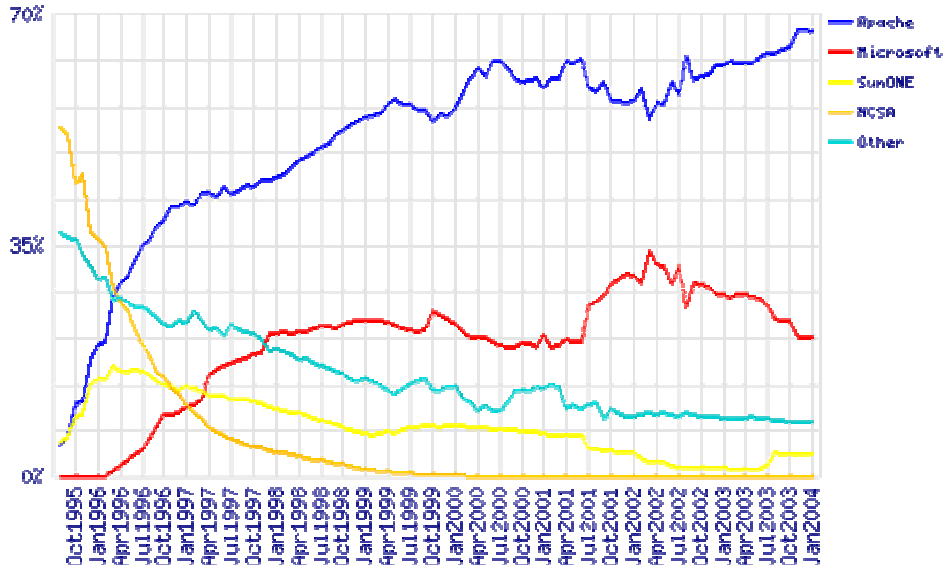
4. Realizacja.

4.1. Podstawowe informacje.

W części praktycznej tej pracy została zrealizowana aplikacja pełniąca funkcję pająka internetowego (jednej z głównych części wyszukiwarki). Będzie to aplikacja udostępniająca interfejs WWW. Będzie zrealizowana w sposób umożliwiający łatwą rozbudowę oraz dodawanie nowych funkcjonalności (np. części dodającej opis strony przy adresie url – czyli stworzenia pełnej wyszukiwarki). Dostęp do systemu będzie realizowany protokołem HTTP.

4.2. Założenia.

Poniżej zostały przedstawione techniczne aspekty realizacji Pająka Internetowego. Jako podstawowa platforma implementacji systemu posłużył serwer WWW Apacze 2 (http://www.apache.org). Apache jest otwartym serwerem HTTP dostępnym dla wielu systemów operacyjnych (m. in. UNIX, GNU/Linux, BSD, MS Windows). Po angielsku słowo Apache wymawia się *epaczi*, co brzmi tak samo jak *a patchy (server)*, co było określeniem tego serwera we wczesnym stadium jego rozwoju w 1995 roku, kiedy był on głównie zbiorem poprawek (*patch*) nałożonych na serwer HTTP o nazwie NCSA [13]. Oprogramowanie to daje zaawansowane możliwości konfiguracyjne i ma małe wymagania systemowe. Nie bez znaczenia jest również fakt, że Apacze jest udostępniany na warunkach licencji GPL (ang. *General Public License*) przez co jest łatwo dostępny i szeroko stosowany.



Rys. 4.1. Popularność serwerów www [15]

Jako podsystem bazy danych zostanie wykorzystana baza MySQL (http://www.mysql.com). Jest to oprogramowanie udostępnione na licencji GPL. Baza danych MySQL charakteryzuje się dużą wydajnością oraz niezawodnością. Zakres implementacji języka SQL oraz mechanizmów bazodanowych jest wystarczający dla zbudowania w oparciu o tą bazę pająka internetowego.

Projekt został wykonany w PHP (ang. *Hypertext Preprocessor*) - skryptowym języku programowania, wykonywanym po stronie serwera, służący przede wszystkim do tworzenia dynamicznych stron WWW, z możliwością zagnieżdżania w HTML-u. Udostępniany na zasadach licencji open-source. Jego składnia bazuje na językach C, Java i Perl. PHP może być używany nie tylko do tworzenia stron WWW. Jego modułowa budowa pozwala także na programowanie aplikacji z interfejsem graficznym (rozszerzenie PHP-GTK), a także na wykonywanie z linii poleceń (podobnie jak Perl i Python).

PHP pozwala także na interakcję z wieloma systemami relacyjnych baz danych (np. MySQL, Oracle, PostgreSQL) oraz na korzystanie z alternatywnych sposobów przechowywania danych - plików

tekstowych i XML-a. Może być uruchamiany na większości systemów operacyjnych (uwzględniając najpopularniejsze) oraz serwerów sieciowych.

Implementacja PHP w środowisku Linux wraz z serwerem Apache i silnikiem baz danych MySQL stanowi popularną platformę serwerową, tzw. LAMP.

4.3. Projekt.

W niniejszym rozdziale zostaną przedstawione techniczne aspekty realizacji pająka internetowego.

Aplikacja będzie realizować następujące cele:

- Pająk Internetowy:
  - pobieranie strony na dysk twardy serwera
  - wyszukiwanie w treści dokumentu kolejnych adresów www
  - analiza pod względem treści
- Przedstawienie rankingu słów
- Lista adresów odwiedzonych
- Lista adresów do odwiedzenia.

Podstawą działania pająka internetowego jest biblioteka cURL (http://www.curl.haxx.se).

Typowe zastosowania cURL to:

- Pobieranie stron z serwera stosującego HTTPS (fopen() nie może być używany w wypadku protokołu HTTPS)
- Łączenie się ze skryptem spodziewającym się danych przesłanych metodą POST
- Tworzenie skryptu wysyłającego próbne dane do własnych skryptów i sprawdzanie wyników.

```
<?php
$file=fopen("plik_do_zapisu.txt","w");
$c=curl_init();
curl_setopt($c,CURLOPT_URL,"https://serwerhttps.pl");
curl_setopt($c,CURLOPT_FILE,$file);
curl_exec($c);
curl_close($c);
fclose($file);
?>
```

gdzie:
„curl_init()” – inicjacja funkcji cURL
„curl_setopt” – ustawienie parametrów opcji
„CURLOPT_URL” – deklaracja adresu z którym nasz skrypt się łączy.
„CURLOPT_FILE” – wskazanie pliku do zapisu
„curl_exec” – wykonanie sesji cURL
„curl_close” – zamknięcie sesji cURL

Dane będą zapisywane do bazy MySQL. Za pomocą PHP tworzymy tabele:

```
<?php
$baza=baza;
$host=localhost;
$user=root;
$login=brak;
mysql_connect($host, $user);
mysql_create_db("$baza");
```

```
mysql_select_db("$baza");
$zapytanie="CREATE TABLE adresy (id int(11) auto_increment
PRIMARY KEY, adesy char(250),
odwiedzony tinyint(1))";
$wykonaj=mysql_query("$zapytanie");

$zapytanie2="CREATE TABLE slowa (idslowa int(11)
auto_increment PRIMARY KEY, slowo char(250),
ile int(11))";
$wykonaj2=mysql_query("$zapytanie2");
?>
```

gdzie:

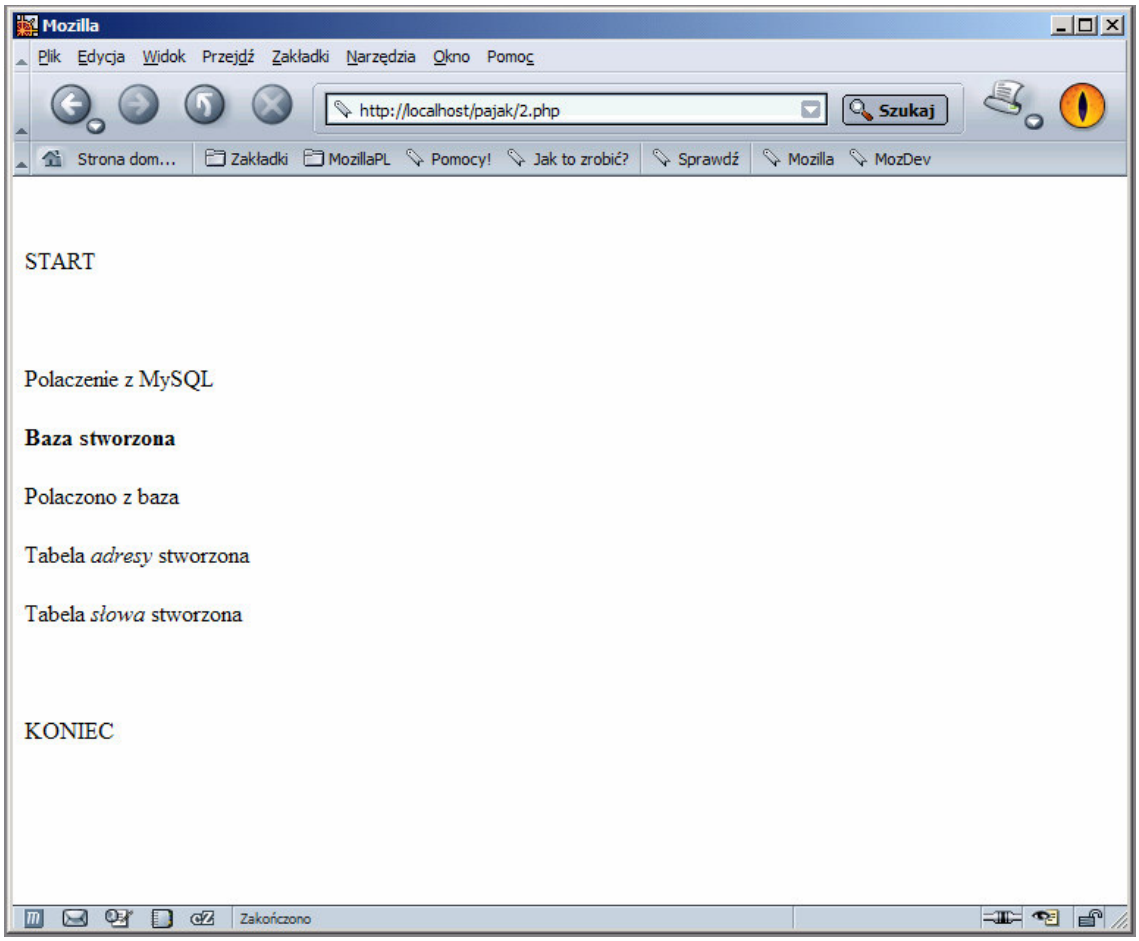
„mysql_connect()” – połączenie z MySQL’em (podanie hosta i użytkownika)

„mysql_create_db()” – stworzenie bazy danych

„mysql_select_db()” – wybór bazy danych

„CREATE TABLE *nazwa*” – stworzenie tabeli o nazwie *nazwa*.

„mysql_query ()” – wykonanie zapytania



Rys. 4.2. Wykonanie skryptu.

W skład bazy wchodzą dwie tabele – *adresy* i *słowa*.

Tabela adresy składa się z następujących pól :

- id – unikalny numer adresu (pola w tabeli)
- adres – adres url
- odwiedzony – pole typu 0 (adres nieodwiedzony) 1 (adres odwiedzony)

←T→	id	adres	odwiedzony
	1	http://www.agh.edu.pl	1
	2	http://www.onet.pl	1
	3	http://www.prv.pl	1
	4	http://www.friko.pl	1
	5	http://www.wp.pl	1
	7	http://www.agh.edu.pl/komunikaty/ranking/wprost.ht...	1
	8	http://www.sejm.gov.pl/komisje/edu/edu.htm	1
	9	http://www.miasteczko.agh.edu.pl/02_wakac.htm	1
	10	http://www.zarz.agh.edu.pl/polska/30lat/30_lat.asp	1
	11	http://www.cyf-kr.edu.pl/iccs2004/	1
	12	http://www.agh.edu.pl/agh/foto.html	1
	13	http://www.agh.edu.pl/aktualnosci.html	1
	14	http://www.agh.edu.pl/studenci/idkns/podyplomowe20...	1
	15	http://www.agh.edu.pl/agh/szkoly.html	1
	16	http://www.agh.edu.pl/agh	1
	17	http://www.agh.edu.pl/rekrutacja	1

Rys. 4.3. Wygląd tabeli adresy w phpMyAdmin

Tabela słowa składa się z następujących pól:

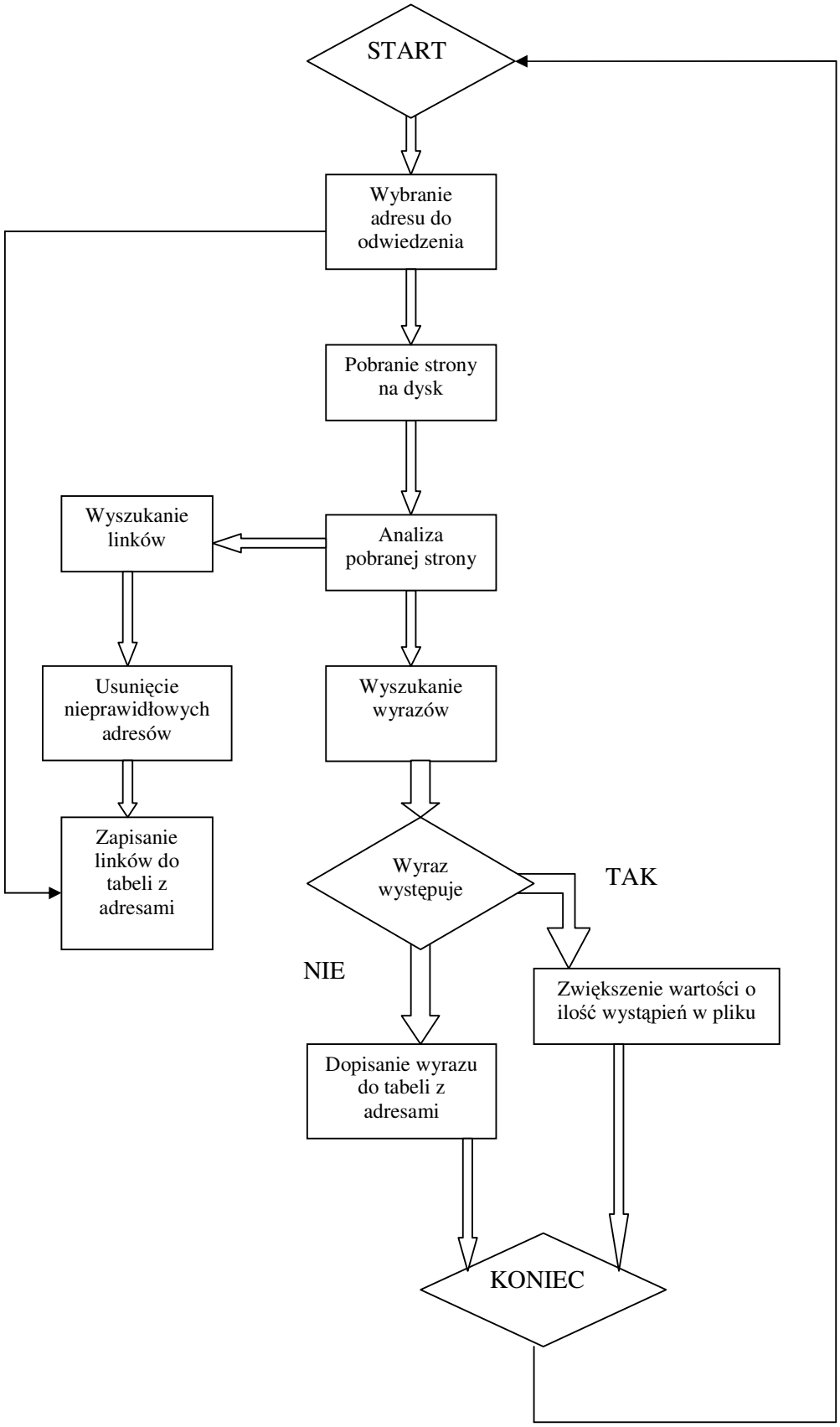
- idslowa – unikalny numer słowa (pola w tabeli)
- slowo – słowo, wyraz.
- ile – ilość wystąpień danego słowa, wyrazu

4.4. Schemat działania pająka internetowego.

Schemat działania pająka internetowego:

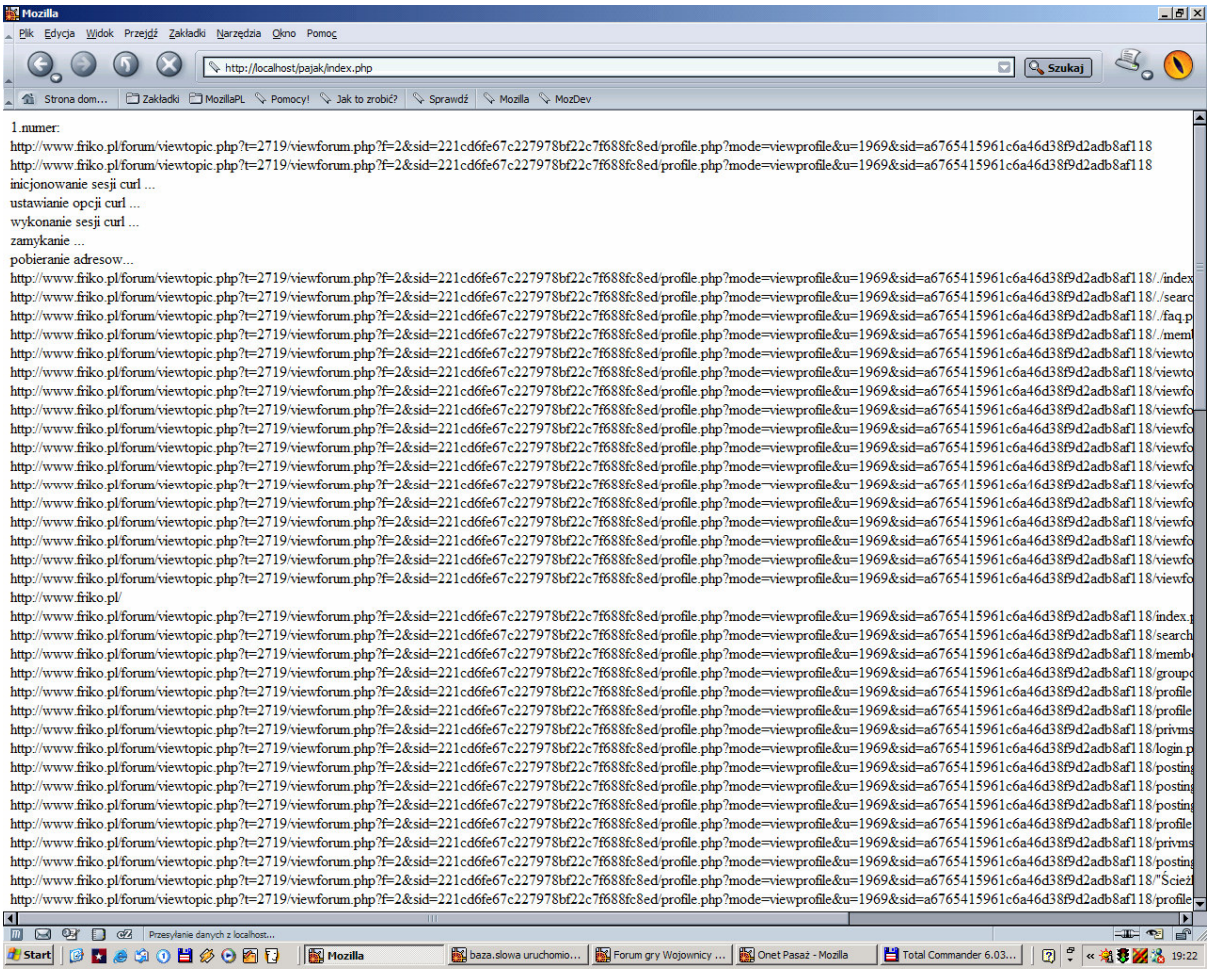
- Pobranie adresu do odwiedzenia z bazy danych (Tabela *adresy*)
- Oznaczenie pobranego adresu jako *odwiedzony*
- Pobranie strony na dysk twardy serwera
- Analiza pobranej strony
    - Wyszukanie linków – sprawdzenie czy są unikalne
        - Usunięcie nieprawidłowych linków
        - Zapisanie prawidłowych linków do Tabeli *adresy*
    - Wyszukanie wyrazów (słów)
        - Sprawdzenie czy dany wyraz występuje w Tabeli *slowa*
            - Jeżeli występuje: zwiększamy jego wartość *ile*
            - Jeżeli nie występuje: dodanie go do Tabeli *slowa* i ustawienie wartości *ile*.
- Zakończenie działania Pająka Internetowego i powrót na start.

Schemat blokowy działania został przedstawiony na rysunku 4.4.



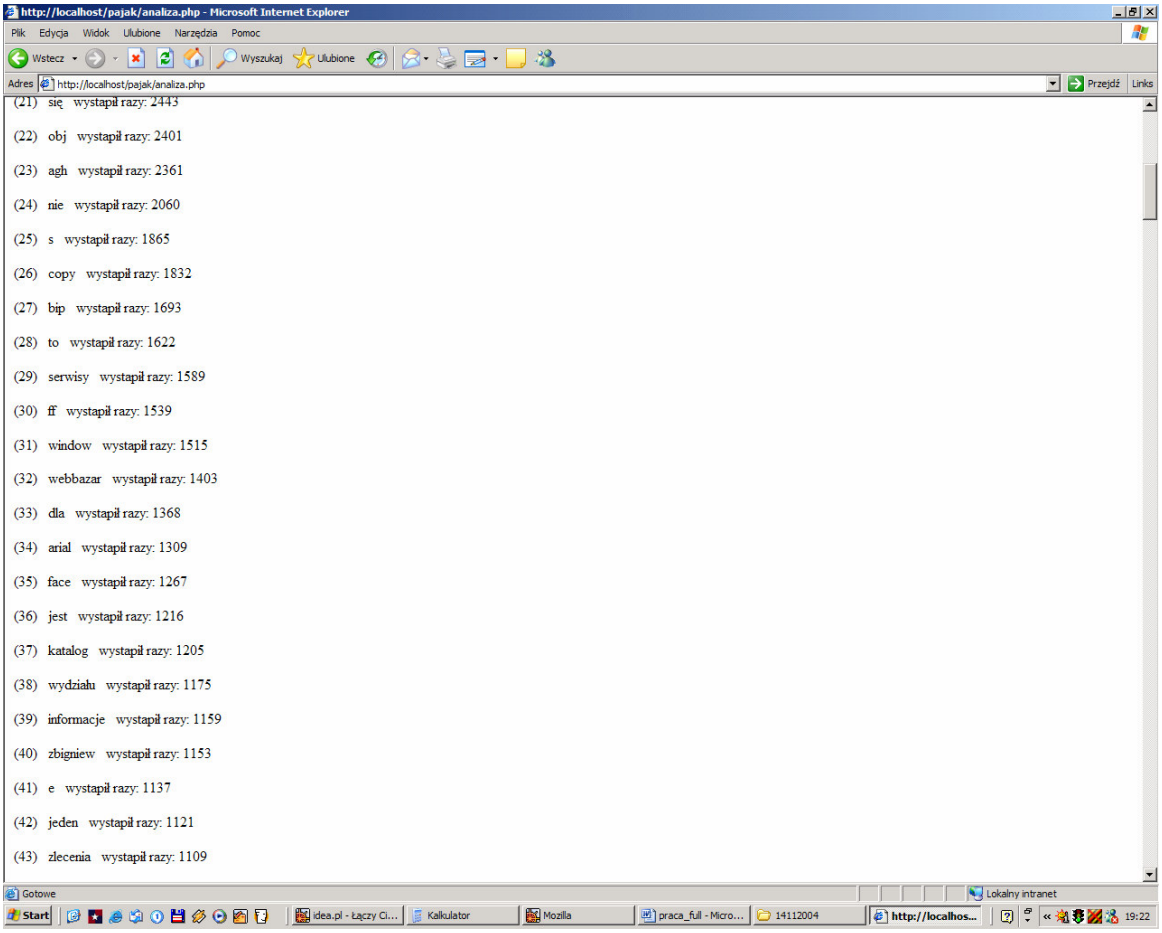
Rys. 4.4. Schemat pracy aplikacji (pająka internetowego).

Widok działającego pająka internetowego został przedstawiony na rysunku 4.5.



Rys. 4.5. Widok działającego pająka internetowego.

Analiza danych jest przedstawiona w oknie przeglądarki i umożliwia przejrzenie trzystu pierwszych wyników. Mamy też możliwość sprawdzenia wszystkich adresów które zostały odwiedzone i które pozostają do odwiedzenia.



Rys. 4.6. Analiza danych – przegląd trzystu najwyższych wyników.

Oto sześćdziesiąt najczęściej występujących słów lub skrótów na polskich stronach internetowych (Lp. wyraz/słowo, ilość wystąpień) :

1.	pl	12357	31.	kategorie	1073
2.	i	11102	32.	oferty	1065
3.	onet	9273	33.	strony	1063
4.	w	8438	34.	enternet	1055
5.	http	7584	35.	roku	1040
6.	html	4444	36.	akademia	1031
7.	z	3846	37.	stron	1007
8.	o	3763	38.	ok	983
9.	na	3599	39.	darmowe	924
10.	a	3425	40.	konta	912
11.	www	2928	41.	komisja	891
12.	do	2714	42.	górniczo	881
13.	się	2443	43.	hutnicza	876
14.	obj	2401	44.	wszystkie	857
15.	agh	2361	45.	uczelni	849
16.	nie	2060	46.	poczta	846
17.	bip	1693	47.	pomocy	828
18.	to	1622	48.	senatu	802
19.	serwisy	1589	49.	rada	789
20.	webbazar	1403	50.	czat	788
21.	dla	1368	51.	świadczeń	779
22.	jest	1216	52.	akademickim	768
23.	katalog	1205	53.	przyznawania	768
24.	wydziału	1175	54.	wypłacania	767
25.	informacje	1159	55.	władz	766
26.	zbigniew	1153	56.	działalności	758
27.	jeden	1121	57.	od	757
28.	zlecenia	1109	58.	inżynierii	752
29.	forum	1105	59.	pomoc	725
30.	krakowie	1095	60.	tylko	719

Wyniki są uzależnione od punktów startowych pająka. W tym wypadku zostały wybrane następujące strony:

- http://www.onet.pl
- http://www.friko.pl
- http://www.prv.pl
- http://www.agh.edu.pl

Dlatego w rankingu pojawiły się słowa: bip (biuletyn informacji publicznej) – informacje ze strony AGH, webbazar – nazwa formy płatności PRV, itp.

4.5. Porównanie wyników.

Przedstawienie i porównanie otrzymanego słownika frekwencyjnego, słownika na podstawie stron Onetu oraz dwóch innych słowników ("Słownik fleksyjny języka polskiego"[15] oraz słownika http://fanthom.math.put.poznan.pl/~janny/index.php?id=studia&sid=studia_kck_slownik [16]).

Tabela 4.1. Porównanie wyników.

Słownik otrzymany		Słownik Onet		Słownik jeden [15]		Słownik dwa [16]	
Słowa	Ilość	Słowa	Ilość	Słowa	Ilość	Słowa	Ilość
pl	12357	pl	5442	w	703536	w	949142
i	11102	onet	5376	i	442062	nie	846607
onet	9273	html	3012	z	375809	i	812235
w	8438	i	1256	na	361857	się	765405
http	7584	w	911	się	333707	na	643045
html	4444	a	693	do	251165	z	592439
z	3846	do	523	nie	222939	to	575158
o	3763	s	505	że	165345	do	435740
na	3599	katalog	502	o	141932	że	351718
a	3425	www	468	to	128234	jest	287145
www	2928	na	462	jest	124290	a	276448
do	2714	czat	381	a	91950	o	256947
się	2443	z	342	od	77189	co	253935
obj	2401	polityka	329	po	71798	jak	199070
agh	2361	c	327	przez	71498	tak	195382
nie	2060	się	310	za	70461	za	160754
bip	1693	kartki	288	dla	60445	ale	158218
to	1622	poczta	283	pap	56117	po	150494
serwisy	1589	d	276	jak	55030	od	126651
webbazar	1403	b	272	roku	51790	mnie	117843
dla	1368	style	269	dziennik	49770	już	112081
jest	1216	wszystkie	252	tym	49540	tym	111891
katalog	1205	góry	247	oraz	48602	dla	108401
wydziału	1175	nie	232	co	44669	czy	103312

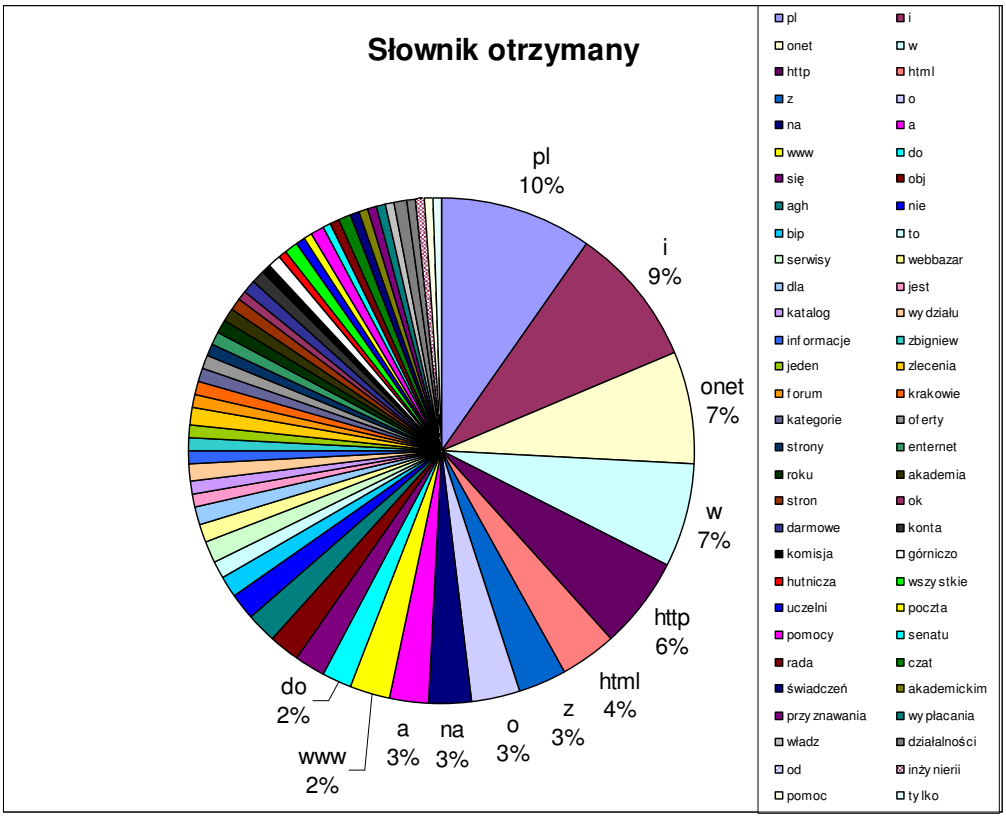
informacje	1159	internet	216	ma	44474	mi	103117
zbigniew	1153	sport	211	są	43977	tylko	100821
jeden	1121	sa	208	już	42126	tego	96412
zlecenia	1109	o	206	ale	41123	go	92684
forum	1105	dotycząca	202	tylko	39419	przez	90597
krakowie	1095	zobacz	200	ze	34122	ma	87836
kategorie	1073	serwisy	198	też	33753	ja	82453
oferty	1065	film	189	tego	33687	może	81181
strony	1063	dla	184	który	32675	tu	75594
enternet	1055	praca	180	jednak	32565	są	72756
roku	1040	kat	174	jego	32218	ci	70698
akademia	1031	artykul	173	przy	31993	jego	70584
stron	1007	internecie	172	proc	30904	ze	69036
ok	983	info	172	lat	30850	pan	65834
darmowe	924	kategoria	172	może	30530	będzie	65591
konta	912	strony	172	które	30382	ten	64776
komisja	891	php	170	pod	29813	ich	64424
górniczo	881	jest	167	ich	29720	jeszcze	64352
hutnicza	876	wiadomosci	166	czy	29045	on	63736
wszystkie	857	narty	155	także	28330	ty	63435
uczelni	849	wiadomości	155	można	28195	był	62216
poczta	846	dzial	154	tak	27761	było	60148
pomocy	828	nieruchomości	153	ul	27159	tam	57002
senatu	802	muzyka	152	przed	25170	być	56678
rada	789	to	152	było	24764	bo	56282
czat	788	pascal	152	lub	24488	sobie	55673
świadczeń	779	banner	151	być	24414	teraz	54586
akademickim	768	co	150	by	23881	jej	54508
przyznawania	768	kropka	150	tej	23780	też	53575
wypłacania	767	pogoda	146	jeszcze	23721	jestem	53403
władz	766	motoryzacyjne	146	min	23214	by	53025
działalności	758	sympatia	144	był	22340	więc	52208
od	757	polskim	138	powiedział	21612	mam	51262
inżynierii	752	stron	136	jej	20818	bardzo	50604
pomoc	725	republika	133	ten	18907	mu	50336
tylko	719	tv	130	bardzo	18691	wszystko	50072

Porównując cztery słowniki można zauważyć, że słownik otrzymany, posiada słowa bardzo często występujące na stronach internetowych (pl, www, html) nie używane często w języku

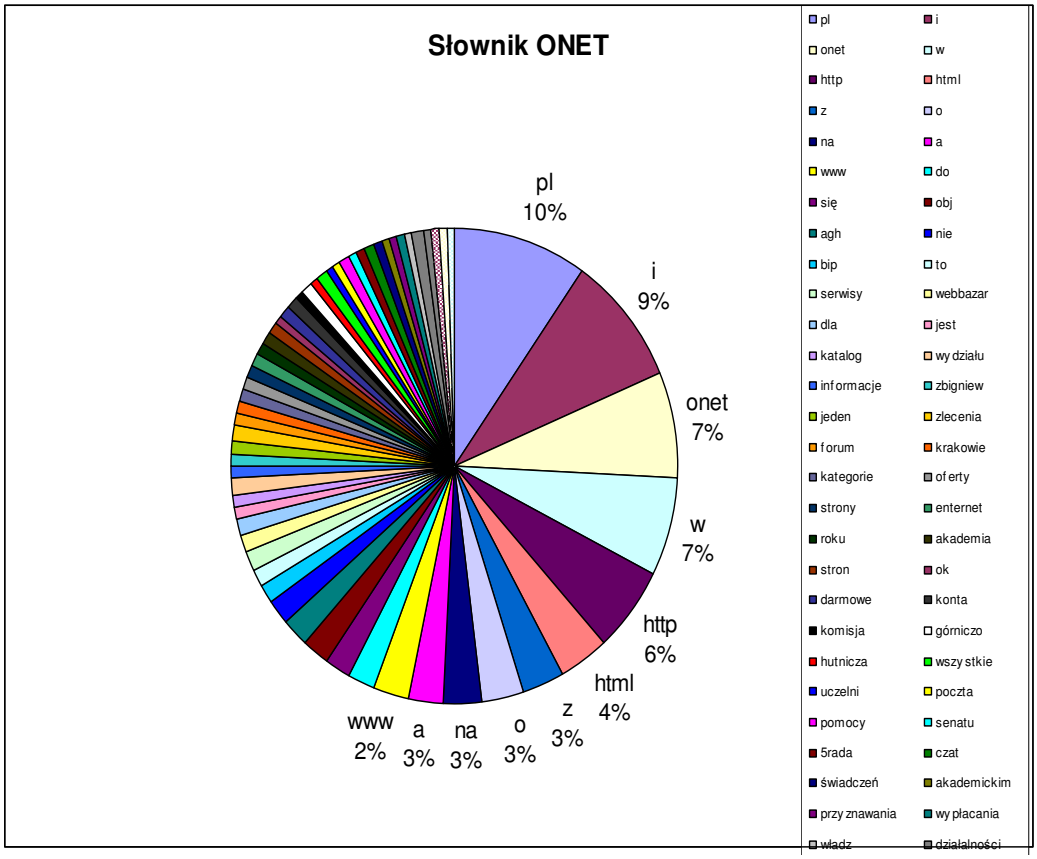
potocznym. Widać jednak, iż wszystkie słowniki (po odrzuceniu specyficznych słów występujących w Internecie) posiadają podobne najpopularniejsze wyrazy (i, w, z, o, na, a, do). Słownik otrzymany na podstawie słów z analizy portalu ONET jest obciążony komercyjnie, występują w nim słowa specyficzne dla niego: Onet, czat, baner itp. Wynika z tego, że tak naprawdę słowa otrzymane zależą od punktów startowych, a same strony mają wyrazy związane z charakterem strony.

Słownik internetowy nie został oparty na żadnym słowniku otrzymanym w inny sposób – wykonany słownik jest samouczący (napotkane wyrazy, nie występujące w bazie danych są automatycznie dodawane). Powodem takiego wykonania jest to, iż w Internecie używa się specyficznego języka (częste skróty (np. NTG – Nie Ta Grupa), słowa nie używane w języku potocznym (np. czat, mail) oraz bardzo często nie stosuje się polskich znaków diakrytycznych (np. robić – robic, miałby – mialby).

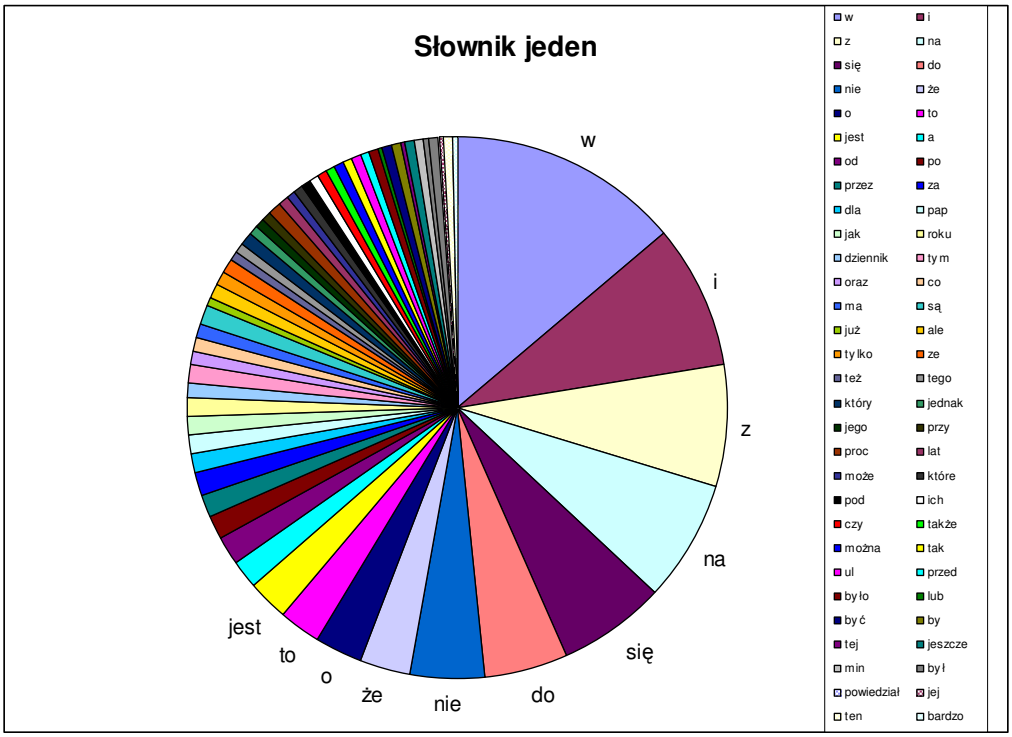
Graficzne przedstawienie otrzymanych wyników (sześćdziesiąt najpopularniejszych słów):



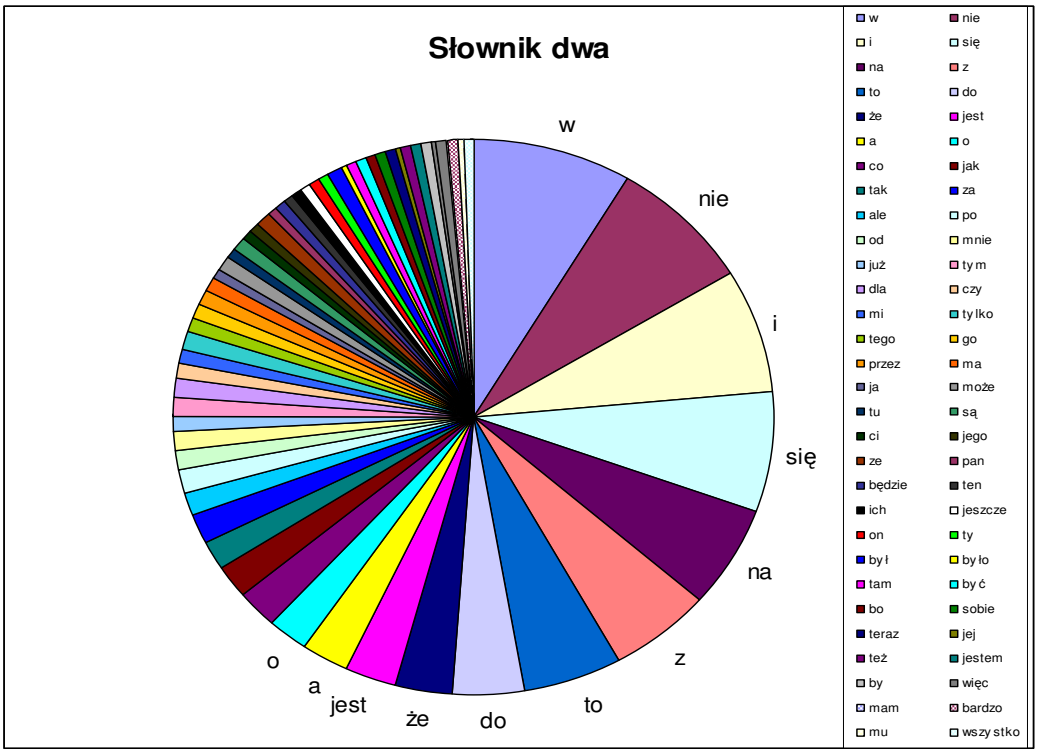
Rys. 4.7. Słownik otrzymany.



Rys. 4.9. Słownik jeden.



Rys. 4.9. Słownik jeden.



Rys. 4.10. Słownik dwa.

Dodatkowo otrzymane dane można przefiltrować za pomocą słownika semantycznego. *Zakładając, że słownik semantyczny, to program przechowujący opisy znaczeń wyrazów, gdzie każde znaczenie jest reprezentowane jako zbiór relacji, zachodzących pomiędzy wyrazem definiowanym, z zbiorem wyrazów definiujących* [17], przykładowo: Internet – Internecie – Internetem. Badania nad słownikiem semantycznym prowadzone są w Grupie Lingwistyki Komputerowej [18], tworzonej przez pracowników Katedry Informatyki AGH, Katedry Lingwistyki Komputerowej UJ i Katedry Automatyki AGH.

5. Zakończenie.

Na podstawie polskojęzycznych stron internetowych skonstruowano słownik frekwencyjny dla języka polskiego został on zrealizowany w dwóch etapach. W pierwszym – teoretycznym przedstawiono sposoby wyszukiwania i pozyskiwania informacji w Internecie poprzez kolejne fazy rozwoju, które doprowadziły do budowy istniejących obecnie zaawansowanych systemów wyszukiwawczych. Przedstawiono sposoby pozyskania informacji z sieci i w jaki sposób stworzyć własne narzędzia wyszukiwawcze.

W części praktycznej przedstawiono projekt Pająka Internetowego. Wykonanie aplikacji zostało oparte o oprogramowanie dostępne na licencji GNU GPL (ang. *General Public License*).

Pająk Internetowy może być wykorzystany w praktyce. Aplikacja została stworzona w taki sposób, aby umożliwić przyszły rozwój o dodatkowe elementy.

Na podstawie otrzymanych wyników widać, że Internet posiada swoje specyficzne słownictwo (czat, www, katalog), otrzymane wyniki są także obciążone komercyjnie (słowa związane z daną stroną internetową, np. webbazar, Onet) lub związane ze specyfiką danego portalu lub strony (senat, akademia). W otrzymanych wynikach widać także część wspólną z pozostałymi słownikami (i, w, na, do, się).

Badanie stron Internetowych posiada pewne ograniczenia związane z przepustowością łącza i dostępnością do serwisów. Duży problem stanowią też w nieprawidłowy sposób wykonane strony – powoduje to konieczność wykonania dodatkowych mechanizmów „czyszczących” wyniki.

Dodatkowe informacje:

1. W pracy zostało wykorzystane następujące oprogramowanie:
    - Serwer www *Apache* http://www.apache.org
    - Baza danych MySQL http://www.mysql.com
    - Edytor HTML *Zajączek* http://amigo.pop.pl
    - MySQL – Front http://www.mysql-front.com/
    - PHP http://php.net
    - cURL http://curl. haxx.se
    - "Słownik fleksyjny języka polskiego", Wiesław Lubaszewski, Henryk Wróbel, Marek Gajęcki, Barbara Moskal, Alicja Orzechowska, Paweł Pietras, Piotr Pisarek, Teresa Rokicka, SFP, Wydawnictwo Pfawnicze LexisNexis, 2001, ISBN 83-7334-055-6, Grupa Lingwistyki Komputerowej, http://winnie.ics.agh.edu.pl, Katedra Informatyki, Katedra Lingwistyki Komputerowej UJ.
    - Słownik, http://fanthom.math.put.poznan.pl/~janny/index.php?id=studia&sid=studia_kck_slownik

2. Praca została napisana w dwóch edytorach: OpenOffice i MS Office.
   Praca została napisana w dwóch edytorach ponieważ była pisana w dwóch różnych lokalizacjach i na oprogramowaniu dostępnym w nich.

3. Legalność.
   Do wykonania pracy dyplomowej została użyte legalne oprogramowanie (z wykupioną licencją, freeware lub open source).

Bibliografia:

[1] Kłopotek Mieczysław Alojzy, *Inteligentne wyszukiwarki internetowe*, Warszawa 2001, Akademicka Oficyna Wydawnicza EXIT

[2] http://www.searchengines.com

[3] http://www.web.reporter.pl, wyszukiwarki

[4] http://www.searchengines.pl

[5] http://www.google.pl

[6] http://www.emulti.pl

[7] http://www.onet.pl

[8] http://www.hoga.pl

[9] Luke Helling, Laura Thomson, *PHP i MySQL, Tworzenie stron WWW, Vademecum profesjonalisty*, Wydanie 2, Wydawnictwo HELION, 2003

[10] Danny Goldman, *JavaScript, Księga eksperta*, Wydanie 3, Wydawnictwo HELION, 2000

[11] Ben Forta, *Poznaj SQL, Krótkie lekcje, właściwe rozwiązania*, Wydanie 1, Intersoftland

[12] http://pl.wikipedia.org

[13] http://pl.wikipedia.org Apacze (serwer)

[14] http://pl.wikipedia.org OpenOffice

[14] http://news.netcraft.com/archives/2004/01/01/january_2004_web_ server_survey.html

[15] Wiesław Lubaszewski, Henryk Wróbel, Marek Gajęcki, Barbara Moskal, Alicja Orzechowska, Paweł Pietras, Piotr Pisarek, Teresa Rokicka, "Słownik fleksyjny języka polskiego", SFP, Wydawnictwo Pfawnicze LexisNexis, 2001, ISBN 83-7334-055-6, Grupa Lingwistyki Komputerowej, http://winnie.ics.agh.edu.pl, Katedra Informatyki, Katedra Lingwistyki Komputerowej UJ.

[16] http://fanthom.math.put.poznan.pl/~janny/index.php? id=studia&sid=studia_kck_slownik

[17] Wiesław Lubaszewski, *Referat przedstawiony na posiedzeniu plenarnym Komitetu Językoznawstwa PAN,* Warszawa, 31 maja 2004
[18] http://winnie.ics.agh.edu.pl/