UNIVERSITAS GALATIENSIS

ROMÂNIA
Universitatea „Dunărea de Jos" din Galaţi

MINISTERUL EDUCAŢIEI NAŢIONALE

C 10653/28.10.2014

C ă t r e

_____

_____

Universitatea „Dunărea de Jos " din Galaţi vă face cunoscut că, în data de **24.11.2014,** ora **10.00,** în **sala Consiliului de administraţie,** va avea loc susţinerea publică a tezei de doctorat intitulată: **"CONTRIBUŢII LA APLICAREA TEHNICILOR DE DATA MINING ÎN INVENTICA ASISTATĂ DE CALCULATOR"**, elaborată de doamna/domnul **VLASE MIHAI**, în vederea conferirii titlului ştiinţific de doctor în domeniul de doctorat **Ştiinţa calculatoarelor.**

Comisia de doctorat are următoarea componenţă :

1. Preşedinte	**Conf.univ.dr.ing. Marian GĂICEANU** Universitatea „Dunărea de Jos" din Galaţi
2. Conducător de doctorat	**Prof.univ.dr.ing. Luminiţa DUMITRIU** Universitatea „Dunărea de Jos" din Galaţi
3. Referent oficial	**Prof.univ.dr.ing. Ştefan TRĂUŞAN-MATU** Universitatea POLITEHNICA din Bucureşti
4. Referent oficial	**Prof.univ.dr.ing. Costin BĂDICĂ** Universitatea din Craiova
5. Referent oficial	**Conf.univ.dr.ing. Cornelia TUDORIE** Universitatea „Dunărea de Jos" din Galaţi

Cu această ocazie vă transmitem rezumatul tezei de doctorat, şi vă invităm să participaţi la susţinerea publică. În cazul în care doriţi să faceţi eventuale aprecieri sau observaţii asupra conţinutului lucrării, vă rugăm să le transmiteţi în scris pe adresa universităţii, str. Domnească nr. 47, 800008 Galaţi, Fax 0236 / 461353, e-mail rectorat@ugal.ro.

RECTORAT

Rector,

Prof. univ. dr. ing. Iulian Gabriel BÎRSAN

Universitatea „Dunărea de Jos” din Galaţi
Școala doctorală de Inginerie

REZUMAT TEZĂ DE DOCTORAT

Contribuții la aplicarea tehnicilor de data mining în inventica asistată de calculator

Doctorand
Mihai Vlase

Conducător științific,	Prof. univ. dr. ing. Luminița Dumitriu
Referenți științifici	Prof. univ. dr. ing. Ștefan Trăușan-Matu Prof. univ. dr. ing. Costin Bădică Conf. univ. dr. ing. Cornelia Tudorie

Seria I2: Calculatoare și tehnologia informației Nr 3
GALAŢI
2014

Seriile tezelor de doctorat sustinute public în UDJG începând cu 1 octombrie 2013 sunt:

Domeniul **ȘTIINȚE INGINEREȘTI**

- Seria I 1**: Biotehnologii**
- Seria I 2: **Calculatoare și tehnologia informației**
- Seria I 3. **Inginerie electrică**
- Seria I 4: **Inginerie industrială**
- Seria I 5: **Ingineria materialelor**
- Seria I 6**: Inginerie mecanică**
- Seria I 7: **Ingineria produselor alimentare**
- Seria I 8: **Ingineria sistemelor**

Domeniul **ȘTIINȚE ECONOMICE**

- Seria E 1: **Economie**
- Seria E 2: **Management**

Domeniul **ȘTIINȚE UMANISTE**

- Seria U 1: **Filologie- Engleză**
- Seria U 2: **Filologie- Română**
- Seria U 3**: Istorie**

Cuvânt înainte

La încheierea procesului de elaborare a lucrării de doctorat, gândurile și recunoștința mea se îndreaptă spre toți aceia care m-au sprijinit și au fost alături de mine la realizarea și finalizarea tezei de doctorat.

Aduc mulțumiri postmortem domnului Profesor dr. ing. Severin Bumbaru, pentru competența cu care a coordonat activitatea desfășurată pe parcursul elaborării tezei precum și întregul suportul oferit pentru depășirea obstacolelor întâlnite de-a lungul anilor.

Mulțumesc doamnei Profesoare dr. ing. Luminița Dumitriu, care prin profesionalismul său, prin tactul pedagogic, răbdarea și înţelegerea manifestă a avut o contribuție definitorie în finalizarea ultimei părți și implicit a întregii tezei de doctorat.

Mulțumesc membrilor Comisiei pentru evaluarea și susținerea tezei de doctorat, pentru onoarea ce mi-o fac prin analiza lucrării precum și pentru sprijinul acordat în finalizarea tezei doctorale.

Doresc să le mulțumesc de asemenea colegilor din Departamentul de Calculatoare și Tehnologia Informației din cadrul Facultății de Automatică, Calculatoare, Inginerie Electrică și Electronică pentru punctele de vedere exprimate, sprijinul moral şi susţinerea permanentă acordată.

Mulțumesc pe această cale Domnului dr. ing. Radu Negulescu pentru ajutorul în inițierea, alegerea temei prezentei teze de doctorat și întregul suport tehnic acordat. Colaborarea cu domnia sa mi-a o ferit o solidă bază de cercetare în domeniul ales.

Nu în ultimul rând, mulțumesc familiei pentru susținerea continuă acordată pe parcursul realizării acestei lucrări. Le sunt recunoscător părinților, pentru sprijinul moral, soției pentru înțelegere, fiicei mele pentru toleranță și nu în ultimul rând surorii mele care m-a încurajat mereu să merg mai departe inspirându-mi tenacitate.

Galați, ing. Mihai Vlase
2014

Cuprins

Contents

Introducere

Cercetarea sintetizată în prezenta teză de doctorat a plecat de la un set de nevoi practice întâlnite în procesul de brevetare a invențiilor. În acest proces de brevetare sunt urmați o serie de pași, printre care unul dintre cei mai importanți este căutarea de informații și referințe existente până în prezent, relevante pentru invenția aplicantului.

La ora actuală există numeroase motoare de căutare în bazele de date de invenții sau în literatura științifică de specialitate, unele aparținând chiar oficiilor de invenții și mărci, altele aparținând unor companii private. Puține sunt însă cele care oferă posibilitatea ordonării rezultatelor căutării după relevanță, folosind unul din algoritmii specifici din data mining existenți.

Ca aplicant de brevet, am fost pus în situația de a face astfel de căutări și am fost nevoit să caut prin zeci de baze de date și sute de rezultate care în marea majoritate s-au dovedit irelevante, și am simțit din propria experiență nevoia existenței unui instrument mai performant care să mă asiste în etapa de căutare [1], [2], [3], [4].

Astfel, inventatorii sunt puși în situația de a căuta informații utile prin liste de sute de rezultate ordonate simplu doar după unul din câmpurile existente în bazele de date, cel mai adesea doar după data brevetării. Există deci, o nevoie reală în găsirea și listarea rezultatelor căutării de brevete relevante și o soluție la această nevoie propune prezenta lucrare.

O altă problemă cu care se confruntă cei implicați în procesul de brevetare este alegerea corectă a clasei din care va face parte un nou brevet. La ora actuală există câteva tipuri de clasificări folosite pentru brevete, însă datorită faptului că prin definiție o invenție aduce ceva nou, nemaiîntâlnit înainte, întotdeauna vor apărea noi clase și chiar industrii inexistente până la acea dată, care sunt câteodată „forțat” clasificate într-una din clasele existente. Aceste clasificări standard sunt actualizate la diferite perioade de timp, însă este destul de greu de spus când a luat naștere o noua clasă sau o nouă industrie. O grupare independentă de clasificarea existentă ar putea spune dacă și-a făcut sau nu apariția o nouă clasă. Prezenta teză își propune să ajute la rezolvarea acestei probleme prin folosirea unor tehnici de data mining, cum ar fi clusterizarea și prin îmbunătățirea și adaptarea acestora la cazul particular al brevetelor de invenție.

Privire de ansamblu asupra tezei

În Capitolul 1 al tezei sunt prezentate tipurile de motoare de căutare disponibile și sistemele acestora de calcul al relevanței. Studiul din capitolul 1 a fost făcut în încercarea de a împrumuta un model de la un sistem la altul, de la un tip de aplicație la altul, în măsura în care acest lucru este posibil, cu scopul obținerii de rezultate mai bune.

În secțiunea 1.3 au fost sintetizate metodele folosite pentru calculul relevanței paginilor web și articolelor științifice, care ar putea fi cel mai bine aplicate în cazul brevetelor de invenție.

În capitolul 1.4 prezentăm modelele matematice din data mining ce stau la baza grupării și ordonării documentelor după relevanță. Sunt discutate aspectele teoretice ale PageRank-ului și calculul acestuia, precum și aspecte teoretice ale clusterizării, capitolul sfârșindu-se cu algoritmul bisecting k-means, cunoscut ca fiind unul dintre cei mai eficienți algoritmi folosiți în clusterizarea textului.

Capitolul 2 reprezintă contribuțiile teoretice aduse de teză cu privire la calculul unui rang pentru fiecare brevet de invenție în parte, în vederea folosirii ulterioare a acestui rang într-un motor de căutare la ordonarea brevetelor după relevanță. În acest scop propunem un calcul al rangului specializat, care ține cont de proprietățile specifice ale brevetelor. Astfel în secțiunea 2.2.1 propunem calculul rangului ținând cont de relevanța aplicantului, iar în secțiunea 2.2.2 propunem calculul rangului ținând cont de anul apariției brevetelor.

În capitolul 3 prezentăm contribuțiile teoretice la clusterizarea brevetelor de invenție, unde este propus un nou model de reprezentare al documentelor de tip text, un model mai apropiat de caracteristicile brevetelor, unde se ține cont și de un set de parametri suplimentari, metadata brevetelor, cu exemplificare pe metadata aplicanților. Clusterele obținute astfel pot fi folosite la rafinarea rezultatelor unei căutări sau la determinarea unor clase noi de brevete, putându-se astfel identifica noi domenii sau industrii.

În capitolul 4 prezentăm o serie de contribuții aplicative, prin care au fost implementate și testate toate aspectele teoretice descrise în capitolele anterioare. Capitolul începe cu prezentarea unor aplicații auxiliare ce permit prelucrarea brevetelor. Dintre acestea amintim:
- aplicația **Patent Data Parser**, din subcapitolul 4.2, cu ajutorul căruia pregătim baza de date și importam brevetele din forma lor inițială în această bază de date;
- aplicația de calcul al rangului, din subcapitolul 4.3, prin intermediul căreia sunt generate rangurile pentru fiecare brevet;
- aplicația de clusterizare a brevetelor **Patent Clustering**, din subcapitolul 4.4, cu ajutorul căreia se pot grupa în clustere distincte brevetele cu un conținut asemănător.

În finalul capitolului 4, mai exact în subcapitolul 4.5, am propus un motor de căutare al brevetelor, care folosește rangurile și clusterizările calculate cu aplicațiile auxiliare menționate anterior. În cadrul acestui motor de căutare, au fost integrate mai multe funcții de căutare, rezultatele obținute în urma unei căutări putând fi ordonate de către utilizatori după diferitele tipuri de ranguri calculate cu algoritmii propuși în teză.

Rezultatele experimentale ale tezei au fost sintetizate în cadrul capitolului 5. Capitolul debutează cu analiza îmbunătățirii performanțelor de calcul în clusterizare detaliată în subcapitolul 5.1 și se continuă cu prezentarea avantajelor folosirii modelelor bazate pe relevanță propuse în teză, exprimate prin rezultatele experimentale ale calculului rangului, subcapitolul 5.2, și cu rezultatele experimentale a căutării în clustere, respectiv subcapitolul 5.3.

În ultimul capitol prezentăm concluziile finale, enumerăm contribuțiile și menționăm direcțiile viitoare de cercetare.

Introduction

The research summarized in this thesis started from a set of practical needs encountered in the process of inventions patenting. In this patenting process a series of steps are followed, including one of the most important, which is the search for information and references available to date, relevant to the applicant's invention.

Currently there are many search engines in inventions or scientific literature databases, some of them even belonging to patent offices, other belonging to private companies. Few are those that offer the possibility of ordering the search results by relevance, using one of the existing specific data mining algorithms.

Myself, as a patent applicant, I was put in a position to make such searches and have been forced to look through dozens of databases and hundreds of results where the vast majority proved irrelevant and I felt from my own experience the need for a more efficient tool to assist me in the search stage [1], [2], [3], [4].

Therefore, the inventors are put in a position to search for useful information through lists of hundreds of results simply ordered only by one of the existing fields in the database, usually by application date. Hence, there is a real need to find and list the relevant patent search results and a solution to this need is proposed in the present paper.

Another problem faced by those involved in patenting is the right selection of the class in which the new patent will be classified. Currently there are several types of classifications used for patents, but because by definition an invention brings something new, never seen before, there will always be a chance that new classes or even new industries to appears and which sometimes these new classes or industries are "forced" classified into one of the existing classes. The standard classifications are updated periodically, but it is quite difficult to say when a new class or a new industry emerged. An independent grouping of existing classification could say if a new class appeared. The present thesis aims to help in solving this problem by using data mining techniques such as clustering and by improving and adapting them to the particular case of patents.

Thesis Overview

In Chapter 1 of the present thesis are introduced the available types of search engines and their specific systems of relevance. The study from Chapter 1 was made as an attempt to borrow a model from one system to another, from one application to another, as far as possible, in order to obtain better results.

In section 1.3 were synthesized the methods used to calculate the relevance of web pages and scientific articles, methods that could successfully be applied to patents.

In chapter 1.4 we presented mathematical models from data mining used for clustering and ordering documents by relevance. We discussed theoretical aspects of PageRank and its calculation, and then we mention theoretical aspects of clustering, the chapter being ended with bisecting k-means algorithm, known as one of the most effective clustering algorithms used in text.

Chapter 2 contains the theoretical contributions to a computed rank for each patent from database, in order to use these ranks in a search engine for patent ordering. To this purpose we proposed a special rank calculation, which takes into account the specific properties of patents. Thus, in Section 2.2.1, we suggested an approach where we took into account the relevance ranking applicant, and in Section 2.2.2, we proposed a rank computing taking into account the released year of the patents.

In Chapter 3 we presented theoretical contributions to patent clustering, where we proposed a new model of text documents representation, a model closer to the patents characteristics, which takes into account a set of parameters, patents metadata, with appliance on applicants metadata. Clusters thus obtained can be used to refine a search results or to determine new classes of patents and could thus identify new areas or industries.

In Chapter 4 we presented a number of applied contributions, where have been implemented and tested all the theory described in the previous chapters. This chapter begins with a presentation of auxiliary applications used in patents processing. Among these are:
- **Patent Data Parser** application, in section 4.2, which can be used to clean the database and to import patents from their original form into the database;
- the rank computing application, in section 4.3, which can be used to generate ranks for each patent;
- **Patent Clustering** application, in section 4.4, which can be used to group similar patents into distinct clusters.

At the end of Chapter 4, specifically in section 4.5, we proposed a patent search engine, which uses the ranks and the clustering computed with auxiliary applications mentioned above. In this search engine, several functions have been integrated, which permits the search results to be sorted by different types of ranks computed with algorithms proposed in this thesis.

The experimental results of the present thesis were summarized in chapter 5. The chapter begins with the improve analysis of computing performances in clustering, detailed in section 5.1 and continues with the the advantages of the proposed relevance based models, expressed by the experimental results of the rank computing, section 5.2, and the experimental results of the search in clusters, section 5.3.

In the last chapter we presented the final conclusions, we listed the contributions and we mentioned future research directions.

Lista figurilor

Lista Tabelelor

1 Studiul actual al inventicii asistate de calculator

1.1 Introducere

Unul dintre cei mai importanți pași în realizarea și brevetarea unei invenții este așa numita căutare „prior art”. „Prior art” este un termen folosit în legislația brevetelor și reprezintă toate informaţiile care au fost publicate în orice formă despre o invenţie înainte de o anumită dată. „Prior art” include orice brevet relevant pentru invenţie, orice articol publicat despre invenţia ce se dorește a fi brevetată sau orice demonstraţie publică.

Există mai multe tipuri de căutări ce se efectuează în timpul procesului de brevetare. Un asemenea tip de căutare este căutarea preliminară, ce se execută în cazul în care invenţia există doar la nivel de idee, iar inventatorii vor să verifice dacă o invenție similară există sau nu deja. Acest tip de căutare este necesară pentru ca inventatorii să nu fie puși în situația de a începe dezvoltarea metodei sau a obiectului ce se dorește a fi inventat și ulterior, în momentul în care deja sunt interesați de scrierea brevetului, să constate că invenția lui sau părți importante din aceasta au fost deja revendicate în alt brevet.

Un alt tip de căutare se efectuează în timpul şi după ce brevetul a fost scris. Această căutare este făcută atât de către inventatori, înainte de aplicarea brevetului pentru acceptare, cât și de către agenţii de brevetare care verifică brevetele aplicate. Legislația brevetelor prevede ca „la preluarea unei cereri de examinare sau a unui brevet în cadrul procedurii de reexaminare, examinatorul va face un studiu aprofundat şi o investigaţie „prior art” referitoare la obiectul invenției revendicate.” [5] În urma acestei căutări brevetele sunt sau nu validate. Aceasta înseamnă că operația de căutare „prior art” ar putea descalifica cererea pentru un brevet, în cazul în care se constată că invenția a mai fost realizată de altcineva în trecut. Drept urmare, inventatorii dau o deosebită importanță acestei etape din procesul de brevetare.

Acest al doilea tip de căutare este executat în vederea găsirii tuturor brevetelor similare cu invenția care de dorește a fi brevetată şi pentru a se asigura că revendicările invenției propuse nu au fost utilizate anterior într-un alt brevet. Dacă această căutare nu se face corect şi nu i se dă importanţa cuvenită, există pericolul ca ulterior acceptării unui brevet, o altă companie concurentă să deschidă un proces de încălcarea a dreptului de folosire a brevetului aplicat şi ar putea pretinde că una sau mai multe revendicări ale propriului brevet i-au fost încălcate. Atunci când un astfel de proces privind încălcarea dreptului de proprietate este deschis, un pas important în soluționarea acestuia este de a executa un al treilea tip de căutare, în scopul găsirii de brevete „prior art” care au fost omise şi nu au fost citate anterior în brevetul aflat in litigiu [6].

În toate aceste tipuri de căutări, problema cu care se confruntă utilizatorii este identificarea informațiilor relevante din multitudinea de brevete și articole existente. În procesul de căutare „prior art” apar două situații importante: în primul rând acoperirea a câtor mai multe brevete posibile incluse în bazele de date de brevete populare, cum ar fi USPTO, WIPO, Esp@ceNet, și terminând cu bazele de date locale cum ar fi DEPATISnet, pentru brevetele din Germania, Japan Patent Office pentru Japonia, OSIM etc. Cea de-a doua problema este selectarea brevetelor relevante din multitudinea de brevete extrase din aceste baze de date.

La ora actuală căutările „prior art” sunt realizate, în cea mai mare parte, cu ajutorul aplicațiilor puse la dispoziție de către marile oficii de brevetare. Aceste aplicații permit însă tipuri de căutări destul de limitate, în urma cărora se obține un număr mare de rezultate care pot fi cel mult ordonate după câmpurile sau metadatele disponibile în brevete. Mai mult, pe baza rezultatelor obținute cu ajutorul acestor aplicații nu se pot realiza analize complexe de căutare sau de urmărire a tendințelor tehnologice la nivel de industrie sau la nivel global.

În continuare este prezentată structura brevetelor, pentru a putea înțelege mai bine relația de similitudine care există între acestea și alte tipuri de documente științifice ce vor fi analizate ulterior.

1.1.1 Analiza structurii brevetelor de invenție

Un brevet este un document care descrie o invenție ce poate fi fabricată, utilizată sau vândută cu acordul titularului de brevet. O invenție este o soluţie la o problemă tehnică specifică. Un brevet conţine în mod normal, cel puţin o revendicare, textul integral al descrierii invenției, precum şi informaţii bibliografice, cum ar fi numele aplicantului [7]***.

Brevetele au o structură fixă riguroasă, cu un conţinut standardizat cum ar fi numărul brevetului, aplicantul, inventatorii, clasificarea domeniul tehnologiei, descriere, revendicări, etc [8]***. Cele mai multe dintre aceste informaţii pot fi găsite în prima pagină a unui brevet şi poartă denumirea de **metadate** [9]. Toate aceste informații speciale ce aparțin brevetelor reprezintă o sursă valoroasă de cunoştinţe [10].

Această structură riguroasă a brevetelor de invenție diferă însă în funcție de zona în care sunt aplicate, iar la scrierea lor pot fi folosite diferite standarde și notații, însa câteva elemente de identificare esențiale se regăsesc la toate. Dintre cele mai importante secțiuni ce sunt conținute într-un brevet putem aminti:

- Titlul
- Numele inventatorilor
- Numele firmei beneficiare (aplicantul / solicitantul)
- Data aplicării brevetului
- Clasa tehnologică din care face parte
- Abstractul sau sumarul
- Descrierea detaliată
- Referințe citate – lista de referințe către alte brevete, articole sau documente din aceeași categorie tehnologică relevante pentru brevet
- Revendicări – fragmente precise de propoziţie care delimitează natura exactă a invenţiei

La fel ca și în cazul articolelor științifice sau al paginilor Web, se remarcă secțiunea de referințe citate. Pe baza acestor referințe se pot calcula diferite metrici sau ranguri care cuantifică importanța brevetelor.

Ca și structură, brevetele de invenție sunt mai apropiate de articolele științifice, însă ca și metode de căutare aplicate și ca relevanță sau specific al căutărilor ce se efectuează în bazele de date de brevete, acestea sunt mai apropiate de metodele folosite pentru paginile web. Spre

deosebire de brevetele de invenție, articolele științifice au particularitatea de a fi grupate în jurnale sau conferințe, iar impactul articolelor este calculat în funcție de impactul jurnalelor. În plus, datorită acestei abordări există câteva dezavantaje ale acestui calcul, dezavantaje ce vor fi prezentate ulterior. În continuare vom aborda problema calculului relevanței brevetelor din perspectiva structurii de legături a paginilor Web.

Modelarea legăturilor Web printr-un graf poate fi extrapolată și în cazul brevetelor de invenție, prin intermediul citărilor acestora. Astfel linkurile din paginile Web sunt înlocuite de către referințele din brevete. Avem atât referințe din brevete către alte brevete, similar linkurilor dintr-o pagină Web, cât si referințe ale altor brevete către un anumit brevet, similar linkurilor ce vin de la alte pagini către o anumită pagină Web.

Pentru a arăta mai bine aceste legături, în continuare am luat un exemplu concret. Pentru exemplificare am considerat brevetul cu numărul **EP0750986**, brevet ce aparține bazei de date European Patent Office (EPO). Brevetul **EP0750986** citează brevetele EP0064939, EP0453790, EP0559556, EP0648599, DE1957270, DE3638322, DE4205746, DE605994, DE8224870, FR397430, FR2561217, FR11969, GB1065028, GB1099069, GB2128953, US4618138. Acelaşi brevet EP0750986 este citat de trei alte brevete EP0851811, EP0960018, EP0977662. De asemenea dacă luam unul dintre brevetele citate de către EP0750986, de exemplu **EP0453790**, acesta citează la rândul său alte şase brevete (EP0134526, DE1611379, DE3343811, DE3838078, GB2218953, US4819928) şi este citat de încă unul (EP0888992).

Aceste citări între brevete formează o structură de tip graf orientat aciclic al citărilor și este descris în Figura 1.

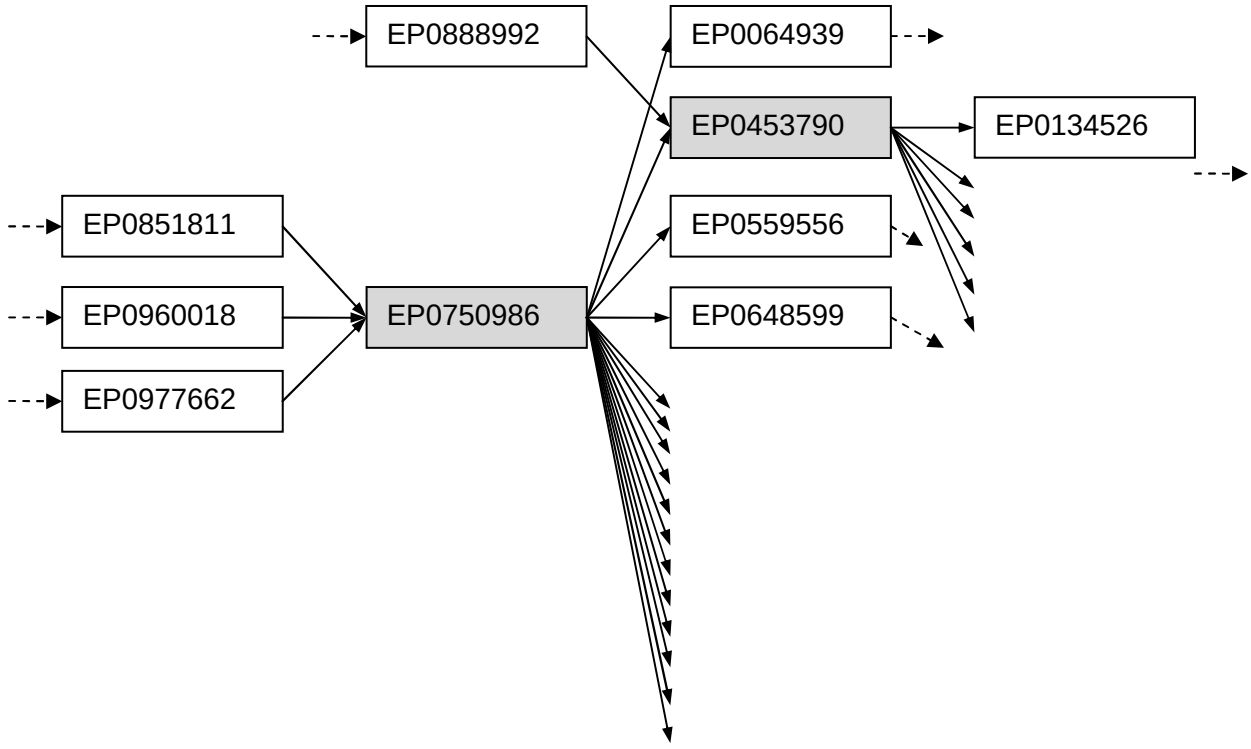

Figura 1 Exemplu de graf al citărilor

1.2 Căutare în documentație științifică

1.2.1 Motoare de căutare

În această secțiune sunt prezentate motoarele de căutare pe Internet. Prezentăm tipurile și caracteristicile, și exemplificăm unele dintre cele mai cunoscute și populare motoare de căutare apărute de-a lungul timpului.

În prezent, în societatea noastră modernă, informatizată, o persoană este expusă într-o singură zi la mai multe informaţii decât era o persoană care trăia acum un secol într-un an de zile. Aceste informații includ: reclame, titluri de ziare, site-uri, mesaje text, semne de trafic, slogane de pe tricouri, afișe etc. Nu este deloc surprinzător că atenția noastră a devenit din ce în ce mai selectivă și ne simțim câteodată copleșiți de volumul mare de informație disponibilă.

În luna martie 2006 în lume, s-au făcut 6.4 miliarde de căutări, în 2008 s-au făcut peste 10 miliarde de căutări pe lună, an de an numărul căutărilor crescând constant, ajungându-se ca în decembrie 2012 numai pe motorul de căutare Google să fie înregistrate 114,7 miliarde de căutări [11]***. Motoarele de căutare acţionează în esenţă ca filtre pentru bogăţia de informaţii disponibile pe Internet și permit utilizatorilor să găsească informaţiile rapid şi uşor. Ele afișează doar paginile care sunt de interes autentic și scutesc utilizatorii de numeroasele pagini web irelevante. Scopul motoarelor de căutare este de a oferi utilizatorilor rezultate ale căutării care să conducă la informaţii relevante. Pentru a furniza rezultate relevante motoarele de căutare utilizează algoritmi complecși pentru a evalua site-urile web și pentru a crea un clasament pentru cele mai căutate cuvinte. Acești algoritmi în marea lor majoritate nu sunt publici, sunt foarte bine protejați și actualizați frecvent. Google de exemplu urmărește peste 200 de parametri diferiți la evaluarea site-urilor web [12]***.

1.2.1.1 Definiție

Ca definiție generală, un motor de căutare (search engine) este un program ce indexează documente sau fișiere sau date dintr-o bază de date sau dintr-o rețea de calculatoare (în special din Internet) [13]***.

Un motor de căutare mai poate fi definit și ca un program ce caută în documente după anumite cuvinte cheie specificate și returnează o listă a documentelor în care au fost găsite cuvintele cheie.

Deși un motor de căutare reprezintă de fapt o clasă generală de programe, termenul este adesea utilizat pentru a descrie sisteme specifice cum ar fi Google, Alta Vista sau Excite, care permit utilizatorilor să caute informații pe Internet.

Cele mai populare motore de căutare funcționează de obicei prin executarea unui program specializat, numit „web crawler”, care are rolul de a colecta cât mai multe documente posibil. Un alt program, numit „indexer”, citește aceste documente şi creează un index bazat pe cuvintele conţinute în fiecare document. Fiecare motor de căutare foloseşte un algoritm propriu de a crea indecși, astfel încât, în mod ideal, doar rezultate semnificative să fie returnate la fiecare interogare [14]***.

Dacă ar fi să definim într-o singură propoziție ce este un motor de căutare am putea spune că un acesta este o resursă care oferă posibilitatea de a căuta informații pe Internet [15].

Pentru documentele World Wide Web cea mai potrivită definiție ar fi că un motor de căutare este un program care accesează Internetul în mod automat și frecvent și care stochează titlul, cuvinte cheie și parțial chiar conținutul paginilor web într-o bază de date. În momentul în care un utilizator apelează la un motor de căutare pentru a găsi o anumită frază sau cuvânt, motorul de căutare se va uita în această bază de date și în funcție de anumite criterii de prioritate va crea și afișa o listă de rezultate (in lb. engleză *hit list*) [16]***.

Motoarele de căutare pot fi clasificate în 3 mari categorii [17]:

1. Motoare de căutare bazate pe „web crawlers”
2. Directoarele web (Web portals)
3. Meta-motoarele de căutare (Meta-Search engines)

1.2.1.2 Motoare de căutare bazate pe „web crawlers”

Un „web crawler” este un program specializat, care parcurge automat World Wide Web într-o manieră metodică și sistematizată [18]***.

Web crawlers sunt cunoscute și sub numele de „web robots”, „web spiders”, worms, walkers, wanderers, ants, „automatic indexers” sau bots, și sunt aproape la fel de vechi ca însăşi web-ul [19]. Primul crawler a fost scris în primăvara lui 1993 de către Matthew Gray și a fost numit „The Wanderer” [20]. În același an a apărut și ALIWEB un crawler creat de către Martin Koster [21].

Principala funcție a unui crawler este de a căuta pe Internet pagini pentru a le indexa. În general se începe cu un set de adrese de web predefinite şi apoi se continuă cu descărcarea lor.

La ora actuală există pe Internet câteva sute de motoare de căutare comerciale bazate pe crawler-e web. Dintre cele mai popular putem le putem aminti pe următoarele: Google, Bing, AltaVista

1.2.1.3 Directoarele web

Directoarele web organizează siturile Web după subiect și de obicei sunt actualizate de către utilizatori și nu de către un program. Când este făcută o căutare, utilizatorul vede siturile organizate într-o serie de categorii și meniuri. Directoarele web sunt de obicei mult mai mici ca dimensiune și structură decât bazele de date ale motoarelor de căutare, datorită faptului ca siturile sunt verificate de către persoane și nu de către un crawler sau program automat [29].

Unele dintre cele mai populare directoare web sunt: Yahoo, Open Directory Project

1.2.1.4 Meta-motoarele de căutare

Meta-motoarele de căutare (sau meta-crawler-ele) sunt site-uri care preiau interogări (sau cuvinte cheie și chiar interogări în limbaj natural) și le trimit la un număr mare de motoare de

căutare. Rezultatele sunt apoi returnate la utilizator. Meta-motoarele de căutare folosesc trei metode de a căuta pe web:

1. Lista directă a motoarelor de căutare - trimite interogarea utilizatorului direct la o listă de motoare de căutare şi adună rezultatele lor pentru acea interogare, ca şi când utilizatorul interoghează direct în fiecare din motoarele de căutare din listă.
2. Căutare secvenţială - utilizator poate selecta unele motoare de căutare dintr-o listă şi apoi poate trimite interogarea utilizatorului la acele motoare de căutare selectate
3. Căutare concurențială - folosește o metodă similară cu cea de căutare secvențială, dar nu mai aşteaptă să primească rezultatele de la toate motoarele de căutare. Pe măsură ce primește rezultate de la primul motor de căutare, le și afişează, iar rezultatele noi primite vor fi treptat adăugate la lista finală de rezultate

Dintre meta-motoarele de căutare disponibile în acest moment pe piață amintim: Ask, Yippy, WebCrawler.

1.2.1.5 Concluzii

După cum am văzut în acest capitol, utilizatorii au de ales dintr-o gamă largă de motoare de căutare de diferite tipuri. Unele folosesc crawler-e web pentru obținerea paginilor și documentelor web, unele sunt directoare web actualizate de către editori, iar altele sunt o combinație a celor două. Multe din motoarele de căutare de-a lungul timpului au atins apogeul după care au intrat în declin, fiind cumpărate și anexate la alte motoare de căutare, rezistând doar cele care au știut să se adapteze noilor tehnologii sau care au folosit propriile tehnologii inovatoare pentru a filtra si ordona miile de rezultate după o cât mai bună relevanță.

În continuare vom discuta despre o altă parte importantă a Internetului deținătoare de informație și anume bazele de date științifice.

1.2.2 Baze de date și motoare de căutare în literatura științifică

În această secțiune vom discuta despre bazele de date științifice și de cele mai populare motoare de căutare în literatura științifică. Acestea includ articole din reviste, articole din conferințe, cărți și fragmente de cărți, prezentări de noi tehnologii, inovații, invenții, brevete de invenție și tot ce poate fi clasificat ca informație științifică. Vom discuta de asemenea despre modalitățile de căutare în aceste baze de date și despre cum pot ajuta ele utilizatorii în căutarea de referințe relevante pentru domeniul lor de cercetare.

Dintre numeroasele motoare de căutare disponibile pentru reviste și conferințe le amintim pe următoarele: CiteSeer, Google Scholar, ACM, IEEE, ISI Web of Knowledge, DBLP, SpringerLink.

1.2.2.1 Motoare de căutare în baze de date de brevete

La ora actuală există foarte multe baze de date de brevete, unele locale, unde fiecare țară gestionează propriile brevete naționale, alte internaționale care au acoperire pe principalele piețe mondiale.

Dintre cele mai importante baze de date naționale se numără cele ale USA, Germaniei, Japoniei, iar dintre cele internaționale se număra European Patent Office (EPO) pentru Europa și World Intellectual Patent Organization (WIPO) cu acoperire la nivel mondial.

În continuare sunt menționate câteva motoare de căutare populare care își iau rezultatele din baze de date de genul celor mai sus menționate: USPTO, EPO, Google Patent Search, Free Patents Online, Delphion, MicroPatent, LexPat, QPAT, PatentMax, PatBase.

1.2.2.2 Concluzii

Din această secțiune se pot desprinde câteva idei importante referitoare la literatura științifică disponibilă în bazele de date existente. Una dintre acestea este că la ora actuală este foarte greu să se reunească toate lucrările de științifice într-un singur loc centralizat, într-o singura bază de date unitară, în care să se poată face o căutare. Cele mai apropiate de acest țel sunt serviciile Google Scholar și Google Patents. Însă datorită diversității formatelor articolelor, revistelor sau a lucrărilor conferințelor, datele bibliografice nu sunt ușor de extras, apărând adesea erori. Probleme apar și la numele autorilor din bibliografii, care în unele articole sunt scrise cu nume și prenume, iar în altele cu prenume și nume, sistemele de analiză și prelucrare identificând astfel doi autori diferiți chiar dacă este vorba despre aceeași persoană. Astfel numărul de citări a unui articol este indexat în bazele de date cu un număr mai mic decât cel real. Mai mult, pot apărea de asemenea situații când există doi autori diferiți cu același nume, iar sistemele de prelucrare identifică un singur autor. Identificarea și unificarea tuturor citărilor unui articol sau autor este o adevărată provocare pentru companiile ce indexează astfel de informație.

În cazul bazelor de date a brevetelor de invenții situația este mai dificilă. Există foarte multe formate în care sunt scrise brevetele, fiecare cu propriile secțiuni sau clasificări. În plus nu toate brevetele sunt scrise în engleză. Fiecare țară are propriile brevete de invenție scrise în limba specifică, acestea având acoperire doar în țările respective. Drept urmare motoarele de căutare existente se limitează la cele mai mari baze de date de brevete cu aceleași format, sau măcar aceeași limbă.

1.3 Relevanța rezultatelor căutării în documentația științifică

Bazele de date tradiționale indexează colecții mari de informații sub formă de înregistrări structurate şi oferă metode pentru interogarea bazei de date pentru a obţine toate înregistrările de care utilizatorii au nevoie. Nevoia extragerii de informații netriviale şi posibil utile, necunoscute anterior, a motivat introducerea unei noi familii de instrumente pentru accesarea informaţiilor din bazele de date, cunoscute sub numele de *descoperirea de cunoştinţe în baze de date* (Knowledge Discovery in Databases (KDD)), sau *data mining*. Ca parte importantă a acestui domeniu este aplicarea de tehnici de analiză statistică în descoperirea automată a modelelor în baze de date, precum şi dezvoltarea mediilor pentru explorarea datelor oferite utilizatorilor. Deşi obiectivul de lucru KDD este de a oferi acces la informaţii şi modele în colecţii de informaţii on-line, cele mai multe eforturi s-au concentrat pe descoperirea de cunoştinţe în baze de date structurate, în ciuda cantității mari de informaţii on-line, care apare doar sub formă de colecţii de text nestructurate.

O abordare a problemei *descoperirii de cunoștințe din text* este aceea în care documentele sunt etichetate de un set de cuvinte cheie, iar descoperirea cunoştinţelor se realizează prin analiza frecvenței de apariţie a cuvintelor cheie ce etichetează diferite documente. Documentele sunt etichetate prin cuvintele cheie luate dintr-un vocabular controlat care este organizat eventual într-o structură ierarhică. De exemplu, cuvintele cheie şi entităţile de nivel superior în ierarhia Feldman, Dagan şi Hirsh sunt folosite pentru a sprijini o gamă largă de operaţiuni KDD pe documente, pentru a indexa documentele în subcolecţii de interes, precum şi de a accesa diferitele documente dintr-o bază de date [78].

O măsură a calității căutărilor în lucrările ştiinţifice este relevanţa rezultatelor obţinute în procesul de căutare. Ordonarea rezultatelor după relevanță reprezintă dispunerea rezultatelor unui căutări realizate cu un motor de căutare, astfel încât cele mai relevante rezultate ale interogărilor făcute de utilizatori să fie afişate pe primele locuri din lista de rezultate. Relevanța este determinată de o combinaţie de parametri diferiţi, cum ar fi frecvența cuvintelor cheie sau de poziția în care apar ele într-un document [79].

O metodă care a fost folosită în mod tradiţional pentru a urmări şi măsura impactul unui articol în timp este *analiza citărilor*. Analiza citărilor permite unui cercetător să urmărească dezvoltarea şi impactul unui articol de-a lungul timpului, uitându-se înapoi la referinţele pe care autorul articolului le citează sau uitându-se la articolele autorilor care citează acel articol. Analiza citărilor a fost făcută populară de către Garfield [80], care a creat trei indici pentru a înregistra citări pentru articole: Science Citation Index, Social Science Citation Index şi Humanities Index. Aceste trei resurse au fost combinate într-o bază de date, Web of Science, actualmente ISI Web of Knowledge [58]***, care a constituit un instrument puternic de cercetare interdisciplinară [81]. Web of Science se bazează pe analiza citărilor pentru a determina factorii de impact a jurnalelor, definit în secțiunea următoare.

În secțiunile următoare prezentăm pe larg cele mai populare sisteme de măsurare a relevanței si impactului documentelor din literatura științifică. O trecere în revistă a câtorva dintre acestea și o comparație a rezultatelor obținute cu diferite motoare de căutare ce le folosesc a fost prezentată în [82].

1.3.1 Conceptul de indexare de citări

Referințele conţinute în articole ştiinţifice sunt folosite pentru a evidenția sursele de inspirație din literatura de specialitate şi pot fi gândite ca o legătură între articole *citate* şi articolele care *citează*. Un *index de citări* (în lb. engleză *citation index*) conţine referinţe citate de articole, ce leagă articolele de lucrările citate. Citările reprezintă o caracteristică semantică a unei publicaţii de cercetare care pot fi folosite pentru a determina relaţiile sale cu alte publicaţii. Indecșii de citări au fost iniţial concepuți în principal pentru extragerea de informaţii [83]. Lucrările pot fi localizate independent de limbă, titlu, cuvinte cheie sau document. Un index de citări permite navigarea înapoi în timp (lista de articole citate) şi înainte în timp (articolele mai noi citează articolele mai vechi), ceea ce îl face un instrument puternic pentru căutările în literatură.

1.3.2 Factorul de impact

Indicatori bibliometrici utilizați în prezent pentru a examina şi a evalua cunoştinţele publicate se bazează în principal pe *factorii de impact* (*Impact Factors*) a jurnalelor indexate în baza de date Science Citation Index, publicate anual din anul 1975 în Journal Citation Reports [84]. Conceptul de *factor de impact* a fost introdus de Eugen Garfield [85], fondatorul Institutului de Informații Științifice (Institute for Scientific Information (ISI)), institut deținut din 1992 de Thomson Scientific Company parte a Thomson Corporation, actualmente Thomson Reuters. *Factorii de impact* sunt calculați ca măsură a frecvenţei medie de citări pentru un anumit element citabil (articol, revizuire, lucrare de conferință, notă, sau abstract) dintr-un anumit jurnal, de-a lungul unui anumit an sau perioadă de timp mai lungă. În mod obișnuit, factorul de impact al unei reviste este definit ca raportul dintre citări şi elemente citabile recent publicate (ultimii doi ani) sau, altfel spus, este numărul mediu de citări dintr-un anumit an pentru articolele publicate într-un jurnal în ultimii doi ani [86].

În România un departament specializat a fost creat în cadrul Ministerului român al Educaţiei şi Cercetării, şi anume CENAPOSS[1], un acronim pentru Centrul Național Pentru Politica Științei și Scientometrie, înființat la sfârşitul anului 1999.

1.3.3 Ratele de citări așteptate

De-a lungul timpului au fost proiectate metode şi tehnici noi pentru evaluarea şi compararea nu numai a jurnalelor privite ca un tot unitar, ci și la articolelor privite la nivel individual, a grupurilor de cercetare sau a oamenilor de ştiinţă. Astfel putem aminti aşa-numitele *rate de citări așteptate* (*Expected Citation Rates (ECR)*) [88], *ratele medii de citări așteptate (Mean Expected Citation Rates (MECR))*, *ratele medii de citări observate (Mean Observed Citation Rates (MOCR))* și *ratele relative de citări (Relative Citation Rates (RCR))* [89].

ECR, propus de ISI, este bazat pe analiza structurii de bază a factorului de impact și este frecvenţa medie de citări pentru un anumit document științific (articol, revizuire, notă, abstracte, scrisoare etc.) într-un jurnal specific, pe parcursul unui an.

Această strategie permite compararea valorii articolelor de același fel. Lucrările din domeniul științei calculatoarelor vor fi probabil mult mai citate decât cele din ingineria nucleară. Drept urmare, comparaţiile trebuie să fie făcute între lucrări din acelaşi specialități, putându-se astfel obține o viziune echilibrată în cadrul respectivei specialități.

MECR reprezintă media ECR pe fiecare publicație, altfel spus,

$$MECR = \frac{numărul\ de\ citări\ aşteptate}{numărul\ de\ publicaţii}$$

[1] Ministerul Educației și Cercetării din Romania, CENAPOSS – un acronim pentru Centrul Național Pentru Politica Științei și Scientometrie, înființat la sfârşitul anului 1999, un departament al CNCSIS – Consiliul Național al Cercetării Științifice din Învățamantul Superior

unde numărul de citări așteptate este calculat pe baza mediei ratelor de citări a jurnalului publicat, adică fiecare lucrare este de aşteptat să primească rata de citări a unui document mediu de aceeaşi vârstă, din aceeaşi revistă [90].

MOCR reprezintă media ratei citărilor pe fiecare publicație, altfel spus,

$$MOCR = \frac{numărul\ de\ citări}{numărul\ de\ publicații}$$

Din cele două măsuri descrise mai sus se poate calcula RCR ca raportul dintre MECR și MOCR:

$$RCR = \frac{MECR}{MOCR}$$

1.3.4 Raportul de impact

O metodă de calcul a performanței unor grupuri de cercetare specifice sau a oamenilor de știință în mod individual este *raportul de impact (Impact Ratio)* și poate fi folosit în calculul fiecărui document științific în parte [91].

Raportul de impact este numărul de citări pentru una sau un set de lucrări împărţit la media pentru domeniul de cercetare în studiu. Prin această metodă, se încearcă compararea performanțelor lucrărilor publicate în funcție de alte lucrări similare ale aceluiași domeniu. S-a ajuns astfel la concluzia că este probabil mult mai important să ne uităm la performanţa individuală a cercetătorului, mai degrabă decât la grup, şi că studierea citărilor pe termen lung este adesea mai relevantă decât cea pe termen scurt [91].

1.3.5 Factorul de impact cumulativ

O evaluare scientometrică simplă a contribuţiilor individuale şi de grup în domeniul ştiinţelor fundamentale a fost propusă și în țara noastră şi a fost introdus un indicator scientometric deosebit de relevant numit *factorul de impact cumulativ* [92], definit prin suma

$$\sum \frac{(jurnal\ impact\ factor, q)}{(article\ autor\ number, a)}$$

sau pe scurt, *S (q / a)*, extinsă pe întreaga listă a publicaţiilor ştiinţifice ale unei persoane sau grup evaluat. Evident, factorul de impact cumulativ reprezintă numărul total de citări a unui autor, în primii doi ani de la data publicării. Pentru o lucrare, acest raport devine citări/autor și este echivalent cu cel pentru un singur autor (a = 1) ce publica într-o revistă cu factor de impact unitar (q = 1).

Indicatorul scientometric cumulativ definit mai sus a fost testat cu succes pentru stabilirea pragurilor de promovare în institutele de cercetare româneşti de fizică şi alte facultăţi [93] şi pentru acreditarea centrelor de excelenţă în cercetare matematică, fizică şi chimie în cadrul CENAPOSS. Pragurile necesare rezultate pentru orice candidat la un post universitar sau de

cercetare sunt, de exemplu, un scor minim de 6 - 8 puncte pentru conferențiar, de 9 - 12 puncte pentru profesor universitar, şi de 14 puncte pentru conducător de doctorat. Cu alte cuvinte, aceste scoruri minime de promovare sunt echivalentul scorurilor de 6-8, 9-12, şi 14 pentru lucrări cu un singur autor, publicate în reviste cu factor de impact unitar. Conversia acestor cifre din domeniul fizicii pentru orice alt domeniu ştiinţific poate fi uşor realizată.

1.3.6 Ordonarea după rangul calculat în funcție de citări

Deşi analiza citărilor nu este un concept nou (Science Citation Index a început publicarea în 1961), puterea de calcul mare disponibilă în prezent face acest concept mai util şi mai larg răspândit. Google PageRank se bazează pe principiul analizei citărilor.

Google, cel mai popular motor de căutare din prezent [94], introduce o altă abordare pentru a calcula relevanţa: algoritmul PageRank-ul [23]. PageRank determină o valoare numerică care reprezintă cât de importantă este o pagină web. Google presupune că, atunci când o pagină are legătură către altă pagină, aceasta garantează un vot de încredere pentru acea pagină. Deci cu cât sunt mai multe voturi exprimate pentru o pagină, cu atât mai importantă ar trebui sa fie pagina. De asemenea, importanţa paginii care dă votul determină cât de important este votul în sine. PageRank nu este singurul factor folosit de Google la rangului pentru pagini, dar este unul dintre cei mai importanți. Calculul matematic al PageRank-ului va fi detaliat în capitolul următor [95].

1.4 Modele și metode formale de data mining pentru realizarea grupării informației din documentația științifică după relevanță

În acest capitol vor fi discutate câteva modele si metode formale de data mining folosite în motoarele de căutare pentru ordonarea informației după relevanță. S-a pus accentul pe acele metode care se pot aplica pe obiecte ce au referințe către alte obiecte de același fel și pe metode de grupare a informației după similitudinea obiectelor din mulțimea de obiecte cercetate.

Vom începe acest capitol cu descrierea indicatorului PageRank, unde vom detalia calculul acestuia și vom continua cu prezentarea câtorva noțiuni generale despre clustere, încheind cu tipurile de clusterizare specifice datelor de tip text.

1.4.1 Calculul indicatorului PageRank

Internetul conține un număr impresionant de pagini, iar acest număr crește de la an la an. Datorită acestui fapt, găsirea informaţiilor relevante în paginile web este o adevărată provocare. Dar spre deosebire de documentele text, paginile web conțin informații suplimentare cum ar fi legăturile către alte pagini. PageRank propune folosirea acestor linkuri în determinarea importanței fiecărei pagini în parte, altfel spus, propune ordonarea după un anumit rang calculat în funcție de legăturile existente între pagini. Această ordonare ajută motoarele de căutare și utilizatorii să găsească rapid informații relevante în vastul și eterogenul World Wide Web [23].

1.4.1.1 Analiza structurii de linkuri Web

Fiecare pagină web are un set de linkuri către alte pagini și dinspre alte pagini către ea. Nu se poate spune cu exactitate numărul de linkuri către o pagină, dar dacă va fi descărcată se pot afla toate linkurile dinspre ea către alte pagini la un anumit moment dat.

În general, paginile care au un număr mare de linkuri către ele sunt mai mult importante decât paginile cu linkuri mai puţine. Numărarea simplă a citărilor a fost folosită ca exemplu pentru a specula viitorii câştigători ai premiului Nobel [96]. Însă PageRank-ul oferă o metodă mult mai sofisticată pentru numărarea citărilor. Ce face atât de interesant PageRank-ul este că sunt foarte multe cazuri în care numărul de citări simple nu reflectă corect importanța unei pagini. De exemplu, dacă o pagină web are un link înspre ea de la pagina de Home de la Yahoo, deși este doar un link, este unul foarte important. Această pagină ar trebui să aibă un rang mai mare decât multe altele cu un număr mai mare de linkuri către ele, dar care provin de la situri mai puțin cunoscute [23].

1.4.1.2 Definiția PageRank-ului

PageRank este definit astfel: o pagină web are rangul cu atât mai mare cu cât suma rangurilor paginilor cu linkuri înspre ea este mai mare. Acest lucru se referă atât la cazul în care o pagină are multe linkuri către ea şi cât și atunci când o pagină are una sau câteva pagini cu rang mare cu linkuri către ea [97].

Fie *u* o pagină de web. Apoi, fie F_u setul de pagini spre care pagina *u* are linkuri, şi B_u setul de pagini care au linkuri către pagina *u*. Fie $N_u = |F_u|$ numărul de linkuri de la *u* către alte pagini şi fie *c* un factor utilizat pentru normalizare (astfel încât rangul total al tuturor paginilor web să rămână constant după fiecare iterație a calculului rangului).

Este definit astfel rangul simplu, *R*, ce reprezintă o versiune simplificată a PageRank-ului:

$$R(u) = c \sum_{v \in B_u} \frac{R(v)}{Nv}$$

Această expresie formalizează conceptele din secţiunea anterioară. Rangul unei pagini este împărţit în mod egal linkurilor sale către alte pagini pentru a contribui in continuare la rangurile acestora. Observăm că $c < 1$, deoarece există un număr de pagini fără linkuri şi ponderea lor este pierdută din sistem. Expresia este recursivă, dar poate fi calculată începând cu orice set de ranguri şi reiterând calculul până când converge.

Calculul rangului în forma sa simplă nu ia în considerare câteva cazuri particulare ce pot influența negativ acuratețea rangului, cum ar fi posibilitatea unei pagini de a avea linkuri către ea însăși.

Ținându-se cont de toate aceste probleme din calculul rangului, s-a ajuns astfel la calculul PageRank-ului pentru o pagină *A* în următoarea forma:

$$PR(A) = (1-d) + d \sum_{v=1..n} \frac{PR(t_v)}{C(t_v)}$$

În această expresie, toți termenii t_v cu *v* de la 1 la *n* reprezintă toate paginile care au linkuri către pagina *A*, *C* este numărul de linkuri din pagina t_v către alte pagini, iar *d* este un factor de amortizare, de obicei setat la 0,85.

Suma $\sum_{v=1..n} \frac{PR(t_v)}{C(t_v)}$ din expresia PageRank-ului poate fi privită simplu ca suma unei părți din PageRank-ul fiecărei pagini cu linkuri către pagina *A*, în care parte din PageRank-ul paginii cu link către pagina *A* înseamnă PageRank-ul împărţit la numărul de linkuri ale paginii.

Astfel, o pagină „votează” câte o parte din PageRank-ul propriu pentru fiecare pagină către care are linkuri. Valoarea PageRank-ului cu care votează este un pic mai mică decât propria valoare PageRank (valoarea proprie * 0,85). Această valoare este împărţită în mod egal între toate paginile către care are linkuri [97].

1.4.2 Metode de clusterizare

În cele ce urmează vom prezenta câteva noțiuni generale despre o importantă tehnică de data mining și anume clusterizarea. Vom defini clusterizarea, vom prezenta câteva tipuri de clustere, tipuri de clusterizări și pentru fiecare tip se va defini distanța între clustere și analiza complexității calculului. Vom încheia cu clusterizarea textului, metoda folosită ca punct de plecare în gruparea brevetelor prezentată în capitolele următoare [99].

1.4.2.1 Definiție

Clusterizarea reprezintă clasificarea, divizarea colecției de obiecte (date) în grupuri diferite, sau mai exact, de partiţionare a unui set de date în subseturi (clustere), astfel încât datele din fiecare subgrup de date conţin aceleaşi caracteristici sau au acelaşi grad de importanţă sau se află una în apropierea celeilalte în funcţie de o măsură a distanţei definită anterior. Clusterizarea este o tehnică frecventă în analiza datelor statistice, utilizată în multe domenii, inclusiv psihologie şi alte ştiinţe sociale, biologie, bioinformatică, machine learning, data mining, pattern recognition şi analiza imaginilor.

În afară de termenul de clusterizare a datelor (sau doar clustering), există un număr de termeni cu sensuri similare, inclusiv analiza clusterelor, clasificare automată, taxonomie numerică sau analiză tipologică.

1.4.2.2 Clasificarea clusterizărilor

Clusterizările pot fi *ierarhice* sau *partiţionale*. **Clusterizarea partiţională** reprezintă diviziunea unui set de obiecte în subseturi distincte (clustere) până când fiecare obiect aparţine unui singur subset.

Dacă permitem unui cluster să aibă subclustere atunci obţinem o **clusterizare ierarhică**. Astfel clusterizarea ierarhică reprezintă un set de clustere imbricate organizate arborescent. Fiecare nod (cluster) din arbore, cu excepţia nodurilor frunze, reprezintă uniunea copiilor săi

(subclustere), iar rădăcina arborelui este clusterul ce conţine toate obiectele. De obicei frunzele reprezintă clustere cu un câte un singur obiect.

Clusterizarea ierarhică poate fi privită ca o secvenţă de clusterizări partiţionale, iar o clusterizare partiţională poate fi obţinută alegând orice membru al acelei secvenţe (de exemplu tăind arborele ierarhic la un anumit nivel).

1.4.2.2.1 Clusterizarea partiţională

Tehnicile de clusterizare partiţională creează un singur nivel de partiţii pentru obiectele unui set de date. Există mai multe astfel de tehnici de partiţionare însă una dintre cele mai folosite si mai eficiente dintre ele este **k-means**.

Problema clusterizării partiţionale poate fi definită după cum urmează: dându-se *n* obiecte într-un spaţiu metric *d*-dimensional, trebuie să se determine o partiţie de obiecte în *k* grupe sau clustere, în aşa fel încât caracteristicile obiectelor dintr-un cluster să aibă un grad de asemănare mai mare decât caracteristicile obiectelor ce aparțin altor clustere.

Algoritmul k-means iniţializează *k* clustere prin atribuirea în mod arbitrar a unui obiect care să reprezinte fiecare cluster. Fiecare dintre obiectele rămase sunt asignate unui cluster apoi criteriul de clusterizare este folosit pentru a calcula media clusterului. Aceste medii sunt folosite ca noi puncte ale clusterului şi fiecare obiect este reasignat clusterului celui mai asemănător. Aceşti paşi se repetă până când nu mai sunt schimbări la recalcularea clusterelor.

Algoritmul k-means

```
1    se selectează arbitrar k  clustere
2    se iniţializează centrii clusterelor cu cele k clustere
3    repetă
4      se partiţionează prin asignarea sau reasignarea tuturor obiectelor la cel mai
   apropiat centru de cluster
5      se calculează noii centri de clustere ca valoarea medie a obiectelor din
   fiecare cluster
6    până când nu mai avem modificare în calculul centrilor clusterelor
```

1.4.2.2.2 Clusterizarea ierarhică

Există două abordări de bază pentru algoritmii ierarhici. Aceştia pot fi aglomerativi sau divizivi (sus în jos sau de jos în sus).

Toţi algoritmii de **clusterizare ierarhici aglomerativi** pornesc cu fiecare obiect ca fiind un grup separat (cluster individual). Aceste grupuri sunt combinate apoi succesiv la fiecare iteraţie, pe baza asemănării dintre ele, până când rămâne un singur grup, sau până când este îndeplinită o condiţie. Pentru *n* obiecte, se execută *n-1* combinări. Gradul de asemănare între clustere stă la baza definirii noţiunii de distanţă dintre clustere.

Algoritmii ierarhici divizivi pornesc de la un singur cluster care conţine toate obiectele şi la fiecare pas se separă un cluster, până când se ajunge la clustere formate dintr-un singur obiect,

sau până când este îndeplinită o condiţie de oprire a algoritmului. În acest caz trebuie să decidem care cluster va fi divizat la fiecare pas şi cum va fi făcută divizarea.

Algoritmii ierarhici sunt rigizi, deoarece, odată făcută o combinaţie sau o divizare între două clustere, acestea nu mai pot fi redivizate sau reunite.

În contextul clusterizării ierarhice, graful ierarhic este numit **dendogramă**.

Spre deosebire de algoritmul k-means, în clusterizarea ierarhică numărul clusterelor *k* nu este cunoscut. După ce a fost construită ierarhia, utilizatorul este cel care specifică numărul de clustere cerute, de la 1 la *n*. Primul nivel al ierarhiei este constituit dintr-un singur cluster, *k* = 1. Pentru a mări numărul de clustere trebuie doar ca să traversăm în jos ierarhia.

Algoritmul de bază pentru clusterizarea ierarhică aglomerativă:

```
1    Se calculează matricea de proximitate dacă e necesar
2    repetă
3      Se unesc două clustere cele mai apropiate unul de celălalt
4      Se actualizează matricea de proximitate pentru a reflecta distanţa dintre noul
   cluster şi clusterele originale
5    până când rămâne un singur cluster
```

Distanţa dintre clustere

Operaţia cea mai importantă din algoritmul de mai sus este calculul distanţei dintre două clustere, iar această distanţă este cea care diferenţiază diferitele tehnici de clusterizare ierarhică aglomerativă. Multe dintre tehnicile de clusterizare ierarhică aglomerativă, cum ar fi **MIN**, **MAX**, sau **media grupului**, provin din vizualizarea bazată pe grafuri a clusterelor.

MIN defineşte distanţa dintre clustere ca fiind distanţa cea mai mică dintre două obiecte aflate în clustere diferite sau, folosind termeni din grafuri, cea mai scurtă legătură între două noduri din două subseturi de noduri diferite.

MAX defineşte distanţa dintre clustere ca fiind distanţa cea mai mare dintre două obiecte aflate în clustere diferite sau, în termenii grafurilor, cea mai lungă legătură între două noduri din două subseturi de noduri diferite.

O altă abordare bazată pe grafuri, **tehnica mediei grupului**, defineşte distanţa între clustere ca fiind media distanţelor perechilor tuturor obiectelor din clustere diferite (în termenii grafurilor media tuturor legăturilor dintre noduri).

1.4.2.3 Măsurarea distanţelor

Un pas important al clusterizării îl reprezintă alegerea tipului de măsură care va determina gradul de similaritate dintre două obiecte. Alegerea tipului de măsură va influenţa forma unui cluster, pentru că e posibil ca unele obiecte să fie considerate apropiate folosindu-se un tip de măsură sau să fie considerate din clustere diferite conform altui tip de măsură.

Dându-se două obiecte p-dimensionale $i = (x_{i1}, x_{i2}, \dots, x_{ip})$ şi $j = (x_{j1}, x_{j2}, \dots, x_{jp})$, pot fi definite următoarele funcţii distanţă cel mai des folosite:

Distanţa Euclidiană:

$$d(i,j) = \sqrt{|x_{i1} - x_{j1}|^2 + |x_{i2} - x_{j2}|^2 + \cdots + |x_{ip} - x_{jp}|^2}$$

Distanţă Manhattan:

$$d(i,j) = |x_{i1} - x_{j1}| + |x_{i2} - x_{j2}| + \cdots + |x_{ip} - x_{jp}|$$

Dacă sunt folosite scale diferite pentru atributele obiectelor, valorile mari ale atributelor din scalele largi fac nesemnificative valorile mici ale atributelor din scalele mici. Pentru a preveni această problemă, valorile atributelor se normalizează în intervalul unitar.

În afară de funcţiile de distanţă descrise mai sus, mai există o serie de măsuri mai puţin folosite si anume: distanţa Mahalanobis şi distanța Hamming. Unghiul dintre doi vectori poate fi folosit ca măsură a distanţei când sunt implicate clustere de mari dimensiuni. De obicei, acesta se foloseşte atunci când datele sunt documente (vectori de cuvinte). Se poate defini astfel cosinusul unghiului dintre doi vectori *d1* și *d2*:

$$cosine(\, d1, d2\,) \;=\; \frac{d1 \;\bullet\; d2}{|d1|\;|d2|},$$

unde ”•” reprezintă produsul scalar, iar |d| reprezintă lungimea vectorului d.

1.4.2.4 Clusterizarea textului

Fundamentul teoretic ce stă la baza motoarelor de căutare este modelul spaţiului de vectori (*Vector Space Model*). Acest model este folosit la reprezentarea textului şi multe din metodele de clusterizare a documentelor folosesc acest model.

În modelul spaţiului de vectori, fiecare text dintr-un set de texte este reprezentat de un vector în spaţiu multidimensional, cu atâtea dimensiuni câte cuvinte există în setul de texte. Fiecare text are câte o măsură (valoare) pentru fiecare indice (dimensiune) calculată în funcţie de apariţia cuvintelor în text. Aceste măsuri cuantifică importanţa fiecărui cuvânt în contextul respectivului text şi depind de cât de des apare fiecare cuvânt în text şi în setul de texte. Textele ale căror vectori sunt apropiaţi unul de celălalt în acest spaţiu, sunt considerate similare ca şi conţinut.

Reprezentarea unui spaţiu de vectori se face în felul următor:

Se consideră un set de *n* texte ce utilizează un set *s* de cuvinte diferite. Fiecare text este reprezentat de un vector

$$d_j = (w_{1j}, w_{2j}, \dots, w_{sj})$$

unde $j \in \{1 \dots n\}$ şi w_{ij} reprezintă ponderea dată cuvântului *i* în textul *j*. Prin reunirea acestor vectori rezultă matricea cuvinte per documente cu elemente w_{ij}

$$\begin{pmatrix} w_{11} & \cdots & w_{1j} & \cdots & w_{1n} \\ \vdots & \ddots & \vdots & & \vdots \\ w_{i1} & \cdots & w_{ij} & \cdots & w_{in} \\ \vdots & & \vdots & \ddots & \vdots \\ w_{s1} & \cdots & w_{sj} & \cdots & w_{sn} \end{pmatrix}$$

Deşi există mai multe versiuni de calcul pentru măsura cuvintelor, cea mai des folosită formulă de calcul este produsul a doi factori şi anume **frecvenţa termenilor (tf)** şi **frecvenţa inversă a documentului (idf)**:

$$w_{ij} = tf_{ij} * idf_i$$

Frecvenţa termenului este o funcţie a numărului de apariţii ale unui cuvânt în document, divizat la numărul de cuvinte din întregul document. Un cuvânt ce apare frecvent într-un text este considerat mult mai important pentru descrierea conţinutului decât un cuvânt care apare mai rar. Frecvenţa inversă a documentului diferenţiază puterea fiecărui cuvânt din setul de texte. Cu cât sunt mai puţine documente ce conţin un cuvânt cu atât mai multă valoare este dată cuvântului în setul de texte. Există mai multe variaţii ale măsurii *idf*, de exemplu $idf_i = \log (n/n_{word(i)})^2$ unde $n_{word(i)}$ este numărul de documente în care apare cuvântul *i*. O schemă de calcul mai apropiată de cazurile reale din practică ar putea fi:

$$tf_{ij} = c_1 + (1 - c_1) * \frac{n_{ij}}{max_i\, n_{ij}}$$

$$idf_i = c_2 + log \frac{n - n_{word(i)}}{n_{word(i)}}$$

unde n_{ij} este numărul de apariţii a cuvântului *i* în textul *j*, iar $max_i\, n_{ij}$ este numărul cel mai mare de apariţii ale unui cuvânt în textul *j*. Constantele c_1 şi c_2 sunt setate manual în funcţie de necesităţile problemei în care apar [100].

În motoarele de căutare, utilizatorul introduce un set de cuvinte şi doreşte ca rezultat un set de documente relevante. Aceste documente sunt relevante pentru utilizator dacă conţin sau au legătură cu termenii introduşi. Termenii introduşi de utilizator sunt trataţi în acelaşi fel ca textele deja existente în setul de texte şi anume sunt reprezentaţi ca un vector în spaţiul vectorilor. Dacă notam cu *q* acest vector, putem spune că textele relevante pentru utilizator sunt acele texte din spaţiul vectorilor mai apropiate de *q*.

Cea mai comună măsură folosită pentru determinarea apropierii sau similitudinii a două texte este **calculul cosinusului** unghiului dintre doi vectori, în cazul nostru între *q* şi textele din set.

$$similar(q, d_j) = \frac{q \bullet d_j}{|q|\,|d_j|} = \frac{1}{|q|\,|d_j|} \sum_i q_i\, w_{ij}$$

Calculul cosinus nu este afectat de dimensiunea documentelor, el măsurând proporţia cuvintelor în documente. Acest lucru este util, pentru că două texte de dimensiuni diferite care aparţin aceluiaşi subiect, sunt similare în conţinut.

Preprocesarea textului

Înainte de a aplica vreun algoritm de clusterizare pentru texte, s-a dovedit a fi utilă, din punct de vedere al performanţelor, aplicarea câtorva modificări asupra spaţiului vectorilor.

Una dintre cele mai uzuale modificări este folosirea unei liste de cuvinte ce nu vor fi folosite la indexare (**stopwords**). Această listă de cuvinte va fi exclusă din spaţiul vectorilor. O astfel de listă poate conţine cuvinte care apar frecvent în seturi mari de texte, cuvinte non-descriptive, şi cuprinde de obicei prepoziţii, articole etc. (de exemplu: and, or, the, to)

Metadatele, se găsesc în paginile web si pot oferi informaţii suplimentare despre importanţa cuvintelor din documente la indexare. Cuvintele care apar în antete (în lb. engleză *header*) sau care apar în format îngroșat (*bold*), sunt probabil mai importante decât alte cuvinte din document. Multe din motoarele de căutare actuale folosesc această informaţie.

Bisecting k-means

În literatura de specialitate au fost propuși mai mulţi algoritmi de clusterizare a documentelor printre care se numără **Scatter/Gather**, **SuffixTree Clustering** sau **bisecting k-means**. În comparaţie cu ceilalţi algoritmi, în particular faţă de algoritmii de clusterizare ierarhici, bisecting k-means s-a dovedit a fi mai rapid şi cu rezultate mai bune de clusterizare [101].

Algoritmul bisecting k-means este un algoritm diviziv, care poate fi folosit pentru clusterizarea documentelor. El se bazează pe algoritmul de bază k-means pentru a împărţi în mod repetat cel mai mare cluster în două, până când se atinge numărul dorit de clustere. Bisecting k-means poate produce o clusterizare partiționată sau ierarhică.

1.5 Concluzii

În acest capitol am comparat diferite tipuri de motoare de căutare și metodele folosite de acestea pentru calculul relevanței documentelor indexate. Între documentele web și brevetele de invenție se pot identifica o serie de asemănări. Ambele tipuri de documente au un sistem similar de citări, astfel încât și în cazul brevetelor am putea aplica metode de calcul al rangului în funcție de aceste citări (cazul motorului de căutare de la Google). De asemenea ambele tipuri de documente au conținut de tip text, astfel încât metodele de clusterizare întâlnite la meta motorul de căutare Yippy ar putea fi împrumutate și la brevete.

Datorită însă structurii speciale ale brevetelor și a importanței diferite a textului în fiecare secțiune dintr-un brevet de invenție, aplicarea directă a metodelor folosite în motoarele de căutare web pe baze de date cu brevete nu este posibilă. Drept pentru care aceste metode vor trebui adaptate la cazul particular al brevetelor.

După cum am arătat în secțiunea 1.3, în cadrul motoarelor de căutare, la ora actuală sunt folosiți foarte mulți indicatori bibliometrici pentru a ordona documentele după relevanța și impactul acestora în lumea științifică.

Deși utilizarea factorilor de impact în jurnale reprezintă o formă de apreciere și de ordonare după importanță și impact și sunt folosiți pe scară largă în afișarea rezultatelor căutărilor în baze de date cu articole, în momentul actual, există studii care contestă corectitudinea evaluării făcută cu acești indicatori. Astfel s-a arătat că factorii de impact ascund o mare diferenţă în ratele de citări per articol. Articolele cele mai citate din prima jumătate a listei de articole sunt citate de 10 ori mai des decât articolele din a doua jumătate. [102]

Prin analiza făcută asupra motoarelor de căutare existente, ce aparțin celor trei domenii considerate (web, lucrări științifice și brevete de invenție), și prin analiza făcută asupra metodelor de calcul al relevanței folosite la afișarea rezultatelor căutărilor pentru fiecare motor de căutare în parte, am putut trage concluzia că doar foarte puține metode pot sta la baza realizării unui motor de căutare eficient pentru brevete.

Drept urmare, am considerat că pentru a obține rezultate relevate pentru căutările realizare în baze de date cu brevete una din cele mai eficiente metode ar fi cea în care relevanța este dată de către popularitatea brevetelor (cele mai citate brevete e posibil sa fie și cele mai relevante). Ca model, cel mai apropiat este graful paginilor web, deci punctul de plecare în calcularea unui rang pentru brevete ar putea fi PageRank-ul.

Dat fiind faptul că specificul căutărilor în brevete este de a căuta documente similare ca și conținut, o altă metodă de calcul al relevanței brevetelor de invenție ce poate fi împrumutată de la motoarele de căutare existente este clusterizarea.

În finalul acestui capitol au fost astfel analizate câteva modele si metode formale de data mining ce pot fi folosite pentru eficientizarea unui motor de căutare pentru brevete. În capitolul 1.4.1 au fost descrise modele și metode pentru calcularea unui rang în vederea ordonării informației după relevanță, aici amintindu-se de algoritmul PageRank, iar in capitolul 1.4.2 au fost explicate câteva generalități cu privire la clusterizarea documentelor de tip text.

2 Contribuţii teoretice la conceperea și utilizarea motoarelor de căutare bazate pe relevanță în inventică

În acest capitol, vom încerca să îmbunătățim căutarea în multitudinea de rezultate obținute și vom propune o soluție pentru extragerea și ordonarea după relevanță a rezultatelor căutării în bazele de date de brevete.

Similar cu cazul PageRank-ului aplicat la paginile web, în cazul brevetelor putem spune că un brevet este cu atât mai important, deci va avea un rang mai mare, cu cât e mai citat de alte brevete, sau un brevet va avea un rang mai mare dacă este citat de un număr mai mic de brevete, dar care au un rang mare.

În acest caz, pentru brevete, PageRank ar putea fi definit în felul următor: un brevet are un rang mare dacă suma rangurilor brevetelor care îl citează este mare. Drept urmare, în secțiunea următoare vom arăta cum putem aplica calculul rangului paginilor web în cazul brevetelor de invenție.

2.1 Calculul rangului simplu

Pornind de la definiția simplificată a PageRank-ului, se poate defini rangul brevetelor după cum urmează:

Fie *p* un brevet (***p***atent). Fie B_p toate brevetele care citează brevetul *p* (”cited ***b***y”). Fie T_p toate brevetele care sunt citate de către brevetul *p* (cite ***t***o). Fie $N_p = |T_p|$ numărul tuturor brevetelor citate de către brevetul *p* şi *c* o constantă. În aceste condiţii rangul (weight) brevetului *p*, *W(p)*, se poate defini:

$$W(p) = c \sum_{v \in B_p} \frac{W(v)}{N_v} \qquad 2.1$$

Această expresie calculează rangul unui brevet ca sumă a rapoartelor între rangul fiecărui brevet care citează brevetul căruia îi calculăm rangul și numărul de brevete citate. Altfel spus fiecare brevet care citează un alt brevet, creditează rangul brevetului citat cu o parte din rangul său.

Referindu-ne tot la teoria din jurul PageRang-ului, o variantă îmbunătățită a expresiei 2.1 este cea în care este considerat și un factor de amortizare. Noua expresie se scrie astfel:

$$W(p) = (1 - \mathrm{c}) + c \sum_{v \in B_p} \frac{W(v)}{N_v} \qquad 2.2$$

Pentru factorul de amortizare *c* s-a ales valoarea 0,85. (v. subcapitolul 1.4.1.2)

Observaţie. Datorită citărilor cronologice stricte, posibilitatea citării reciproce sau în buclă este exclusă [103].

În comparație cu numărul extrem de mare al paginilor web indexate în bazele de date web, bazele de date cu brevete au un număr redus de înregistrări (miliarde versus câteva milioane în cazul în care se reușește reunirea câtorva baze de date mai importante). Astfel, în unele cazuri, folosind doar calculul rangului simplu, se obține o departajare a brevetelor bazată pe numărul de citări reciproce, însă rezultatele obținute nu sunt cu mult mai spectaculoase decât simpla ordonare a brevetelor după numărul de citări către ele. Rezultatele în cele două cazuri sunt relativ comparabile.

Diferențe mici între rezultatele ordonate după cele două criterii menționate mai sus, apar cel mai adesea atunci când sunt căutate cuvinte cu frecvență mică în baza de date a brevetelor, unde în urma căutării după aceste cuvinte, numărul de rezultate este oricum mic, caz în care brevetele listate pot fi ușor de parcurs. Adevărata valoare a ordonării după ranguri apare în momentul în care în urma unei căutări este returnat un număr mare de brevete, caz în care între cele două tipuri de ordonări discutate mai sus, apar diferențe substanțiale.

2.1.1 Concluzii

Folosind calculul rangului simplu, descris în acest capitol, putem calcula un rang pentru fiecare brevet în parte. Ordonând rezultatele unei căutări în baza de date după cuvinte cheie, șansa de a obține cele mai relevante brevete pe primele poziții este mult mai mare decât dacă am lista rezultatele simplu după dată sau după oricare alt câmp, așa cum se întâmplă la ora actuală în majoritatea motoarelor de căutare existente (de exemplu EPO).

Ținând cont de particularitățile brevetelor, numele inventatorilor, numele aplicantului, numele agentului care a reprezentat brevetul la aplicare, data aplicării, etc. putem îmbunătăți calculul rangului incluzând și acești parametri specifici. În capitolul următor prezentăm câteva modele bazate pe relevanță, în care calculul rangului folosește parametri specifici brevetelor.

2.2 Calculul rangului specializat

În cele ce urmează vom prezenta un model îmbunătățit de calcul al rangului pentru brevete în care vom ține cont și de unele dintre secțiunile conținute de metadata brevetelor și anume: numele inventatorilor, numele aplicantului, numele agentului care a reprezentat inventatorul / aplicantul, data aplicării, etc.

2.2.1 Contribuții privind calculul rangului, ţinând cont de relevanţa aplicantului

În practică, inventatorii sau agenții care fac cercetări „prior art”, găsesc relevante cu precădere brevetele provenite de la firme cu renume în domeniul respectiv. Acest lucru se întâmplă din două motive. Primul ar fi că firmele mari au o existență îndelungată în respectiva industrie și aplica pentru noi invenții cu o frecvență destul de mare, devenind astfel etalon în industria respectivă. Astfel, experienţa şi numărul mare de brevete ale firmelor mari contribuie la importanța brevetelor aplicate de aceştia. Este mult mai probabil ca un brevet scris de către o firmă de renume să fie mai important decât un brevet scris de o firmă care nu are decât un

singur brevet. Cel de-al doilea motiv ține de aspectul legal al brevetelor. Un eventual proces de încălcare a drepturilor de autor (infringement) apare de obicei din partea firmelor mari și mai rar din partea celor mici. Firmele mari au suficiente resursele și timp pentru a duce până la capăt un astfel de proces. Firmele mici nu au suficiente resurse pentru a urmări în permanenţă ultimele brevete apărute în domeniul lor de activitate, astfel încât să poată identifica și mai apoi deschide un eventual proces de încălcare a drepturilor de autor.

Studiind brevete ale firmelor mari și, implicit, de succes, inventatorul își poate face o idee despre piața de desfacere a produselor sau a metodelor tehnice inventate. În cele din urmă, aplicarea unui brevet are ca scop protejarea invenției în vederea producerii și comercializării cu succes a obiectelor sau metodelor inventate.

O deosebită importanţă în calculul rangului brevetelor o reprezintă astfel aplicantul. Aplicantul este firma în numele căreia s-a făcut invenția, beneficiarul invenției.

Pentru a creşte mai mult rangul brevetelor aplicate de către firmele de prestigiu, în vederea listării acestora pe primele poziții în urma unei căutări, se introduce un nou rang calculat pe baza numărului de brevete scrise de către un aplicant, care intră apoi în calculul rangului total al brevetului.

Fie *WA(a)* rangul aplicantului *a* (applicant weight), şi *W(p)* rangul brevetului *p* (patent weight). Fie P_a toate brevetele aplicantului a, B_p toate brevetele care citează brevetul *p*, şi $citecount_p$ numărul de brevete citate de către brevetul *p*. Fie *c1*, *c2*, *c3*, *w1*, *w2* şi *w3* constante.

$$WA(a) = c1 + w1 * \sum_{v \in P_a} W(v) \qquad 2.3$$

$$W(p) = c2 + w2 * \sum_{v \in B_p} \frac{W(v)}{citecount_v} + WA(a) * w3 \qquad 2.4$$

Calculul rangului aplicantului introdus în algoritmul de calcul al rangului brevetelor, face ca acest model să crească după câteva iteraţii rangul foarte mult pentru brevetele aplicate de firme cu nume mari.

Creșterea sau importanța dată de aplicant poate fi ponderată de constanta *w3*, însă nu este suficient. S-a constatat experimental că termenul rangului dat de citări devine nesemnificativ în comparaţie cu valoarea dată de termenul rangului dat de aplicant. Astfel practic cele mai relevante brevete sunt strict ale acelor firme care au cel mai mare număr de brevete scrise: IBM, SIEMENS, BASF, CANON, MATSUSHITA, SONY, HITACHI, etc.

Este necesară astfel o ajustare suplimentară a influenţei termenului dat de rangul aplicantului. Folosind funcția logaritmică putem diminua importanța excesivă dată de termenul *WA(a)*:

$$W(p) = c2 + w2 * \sum_{v \in B_p} \frac{W(v)}{citecount_v} + \log\left(1 + \frac{WA(a)}{c3}\right) * w3 \qquad 2.5$$

S-a ales funcţia logaritmică pentru a reduce influenţa numărului mare de brevete scrise de un aplicant, astfel încât pe măsură ce numărul de brevete creşte, influenţă rangului aplicantului creşte din ce în ce mai puţin.

Prin încercări succesive s-a determinat că cele mai bune rezultate se obţin dacă constantele iau următoarele valori:

c1 = 0.15; *c2* = 0.15; *c3* = 10000; *w1* = 0.85; *w2* = 0.85; *w3* = 0.01;

Algoritmul pentru calculul rangului în care se ține cont de metadata aplicant, calculează mai întâi rangul aplicantului $WA(a)$ cu expresia 2.3, și mai apoi cu valoarea astfel obținută se determină rangul brevetelor $W(p)$ folosindu-se expresia 2.4. Algoritmul converge după 8, 9 paşi, de fiecare dată calculându-se consecutiv, rangul aplicantului, apoi rangul brevetului ţinând cont de rangul aplicantului.

2.2.2 Contribuții privind calculul rangului ținând cont de parametrul timp

Atât brevetele de invenție cât și articolele științifice, au un atribut comun ce influențează relevanța lor și anume data publicării. Cu cât un articol sau un brevet este mai vechi, cu atât șansele acestuia de a fi citat cresc. În cea mai mare parte, articolele științifice noi și într-o manieră mai accentuată brevetele noi, nu au citări către ele, pentru simplul motiv că nu a existat timpul necesar să poată apărea noi teorii, metode, invenții care sa fie înrudite cu aceste noi articole sau brevete pentru a fi citate.

Astfel, dacă folosim pentru calculul relevanței brevetelor doar rangul bazat pe citări, este evident faptul că brevetele mai noi vor fi dezavantajate, deoarece au puține citări. Acestea vor apărea în mare parte spre sfârșitul listelor cu rezultatele returnate în urma unei căutări în baza de date.

În continuare este propus un model în care rangul brevetelor este calculat ținând cont și de influența anului apariției brevetului. Se adaugă astfel o pondere mai mare brevetelor noi, și pe măsură ce anul apariției brevetelor scade, scade și rangul brevetelor din acel an [104].

Atenuarea problemei induse de vechimea apariției brevetelor a fost abordată în felul următor:

- la pasul unu se injectează un termen de ponderare calculat în funcție de anul apariției brevetelor,
- la pasul al doilea se normalizează toate rangurile brevetelor pe fiecare an în parte

Folosind rangul simplu pentru calculul rangului brevetelor de invenție, se observă în lista de rezultate obţinute că ies pe primele poziții brevetele din ani mai vechi. După cum am arătat mai sus, explicaţia este că datorită creşterii în timp a numărului de brevete şi datorită dependenței unidirecţională a citărilor, de la brevetele mai noi către brevetele mai vechi, brevetele vechi sunt mai citate decât cele noi, astfel încât au posibilitatea să cumuleze rang mai mare de la mai multe brevete.

Din acest motiv este necesară o corecţie la calculul rangului și anume, introducerea unui termen *inject*, cu valori diferite pentru fiecare an:

$$inject = \frac{1}{(y_{present} + 2 - y)} \quad 2.6$$

unde $y_{present}$ este anul curent în care se face calculul, la care se adaugă apoi valoarea doi, iar *y* este anul apariției brevetului.

Deoarece funcția folosită în calculul termenului *inject* este o funcție putere cu exponent negativ, s-a ales pentru $y_{present}$ anul în care se face calculul plus încă două unități pentru ca valoarea termenului *inject* să nu fie foarte mare pentru anii apropiați anului curent. De exemplu daca ultimele brevete din baza de date sunt din anul 2009, $y_{present}$ + 2 ia valoarea 2011.

Pentru adăugarea unei ponderi mai mari la rangul brevetelor mai noi, algoritmul de calcul al rangului simplu va fi modificat în felul următor:

Fie *W(p)* rangul unui brevet *p*. Fie B_p toate brevetele care citează brevetul *p*. Fie $citecount_p$ numărul de brevete citate de către brevetul *p*. Fie *inject* variabila descrisă mai sus, cu rolul de creștere a ponderii rangului brevetelor mai noi. Atunci,

$$W(p) = (1 - inject) + inject * \sum_{v \in B_p} \frac{W(v)}{citecount_v} \quad 2.7$$

După calculul unui pas al rangului se aplică apoi normalizarea lui pe ani.

Fie $WN(p)$ rangul normalizat al brevetului *p*. Fie B_y mulțimea tuturor brevetelor apărute în anul *y*, iar $N_y = |B_y|$ numărul brevetelor apărute în anul *y*. Atunci,

$$WN(p) = \frac{W(p)}{media_y}$$

unde

$$media_y = \frac{\sum_{v \in B_y} W(v)}{N_y}$$

Spre deosebire de rangul simplu unde factorul de amortizare este o constantă, în cazul calculului rangului ținând cont de parametrul timp, termenul *inject* este ajustat pentru fiecare brevet în parte, în funcție de anul apariției. Procedând astfel, tendința rangului este cea corectă. Pe măsură ce ne apropiem de prezent, rangurile calculate pentru brevetele noi crește. Această creștere a rangului brevetelor noi este compensată de valoarea calculată a rangului brevetelor vechi, acest lucru datorându-se numărului de citări mai mare de care beneficiază brevetele mai vechi.

În cadrul calculului a necesară folosirea normalizării pentru omogenizarea valorilor rangurilor pe ani, diminuând astfel valoarea rangurilor folosite la pasul următor în algoritm.

Algoritmul după rulare, converge în aproximativ 9 paşi. După cel de-al nouălea pas, poziţiile brevetelor nu se mai schimbă semnificativ între ele comparativ cu dimensiunea întregii baze de date.

2.3 Concluzii

Folosind pentru calculul rangului modelele ce țin cont de parametrii specifici brevetelor, cum ar fi anii sau aplicantul, se observă o îmbunătățire a relevanței rezultatelor obținute în urma efectuării unor căutări față de cazul în care se folosește rangul simplu.

În cazul rangului ce ține cont de aplicant, pe primele poziții vor apărea preponderent brevetele cu aplicanții ce au un număr mare de invenții înregistrate, ceea ce îi va ajuta pe utilizatori în identificarea eventualilor mari concurenți din domeniul lor de cercetare sau în identificarea de brevete relevante ce aparțin marilor companii ce activează în domeniul lor de interes.

În cazul calculului rangului ce ține cont de anul brevetării obținem o distribuire mai omogenă a brevetelor în lista de rezultate, astfel încât brevetele noi nu mai sunt localizate pe ultimele poziții ale listei de rezultate.

Algoritmul de calcul ce ține cont de aplicant se poate aplica cu succes și în cazul în care ne interesează relevanța brevetelor din punctul de vedere al inventatorilor. Se poate spune că un brevet este cu atât mai important cu cât inventatorul lui are mai multe brevete create de-a lungul timpului. Drept urmare, la fel ca și în cazul aplicanților, se poate calcula o pondere mai mare la rangul brevetelor scrise de inventatori cu un număr mai mare de brevete semnate.

Problema practică ce apare în calculul rangului după aplicant sau după numele inventatorului, este că aceștia nu sunt întotdeauna identificați corect. E posibil ca de la brevet la brevet să difere numele aplicantului, deși este vorba de același aplicant, sau numele inventatorilor să fie scris într-un brevet ca prenume și nume, iar în alt brevet ca nume și prenume, iar sistemul să nu îl identifice ca fiind vorba de aceeași persoană. Deci o importanță deosebită trebuie dată prelucrării inițiale a datelor și identificarea corectă a parametrilor comuni din brevete.

În capitolul 4 este propus un motor de căutare bazat pe aceste ranguri care ordonează rezultatele căutării în funcție de rangul calculat pentru fiecare brevet.

3 Contribuţii teoretice privind clusterizarea brevetelor de invenție

De multe ori în căutarea „prior art”, când este găsit un brevet relevant, este util de studiat și grupul de brevete înrudite cu acesta. Acest lucru se poate realiza fie studiind referințele brevetului găsit, fie studiind brevetele din clasa principală a brevetului găsit. De multe ori însă numărul brevetelor dintr-o clasă este de ordinul zecilor și nu toate sunt relevante pentru căutarea noastră.

De asemenea, într-o formă de căutare în care se urmărește evoluția pieții dintr-un anumit domeniu este important să se identifice care sunt firmele reprezentative din domeniul tehnic de interes. O atenție deosebită trebuie dată brevetelor ce aparțin acestor firme. Din multitudinea de brevete aplicate de aceste firme trebuie selectate însă numai cele înrudite cu domeniul pentru care se face căutarea.

Apare astfel nevoia unei grupări a brevetelor după conținut, independentă de clasificarea existentă. Acest lucru poate fi realizat aplicând o tehnică frecvent folosită în data mining și anume clusterizarea.

3.1 Sisteme de clasificare a brevetelor

După cum am prezentat în capitolul 1.1.1, fiecare brevet are o secţiune în componenţa sa numită clasificare. În funcţie de tipul de brevet, aceasta poate avea unul sau mai multe sisteme de clasificare:

- U.S. Patent Classification System (USPCS) – sistemul de clasificare US
- International Patent Classification (IPC) – sistemul de clasificare internaţional
- European Classification (ECLA) – sistemul de clasificare european, care este o extensie a IPC şi conţine cu aproximativ 64000 mai multe subdiviziuni faţă de IPC, deci este mai precis

În brevetele aplicate în US sunt prezente atât clasificarea USPCS cât şi IPC. În brevetele europene sunt prezente clasificările IPC şi ECLA.

În studiul de caz prezentat în lucrare s-a luat în considerare clasificarea IPC, deoarece aceasta este prezentă printre secțiunile majorității tipurilor de brevete existente la nivel global. În cele ce urmează vom explica cum se formează notația claselor din clasificarea IPC.

IPC oferă un sistem ierarhic de simboluri independente de limbă, folosit în clasificarea brevetelor în funcţie de zonele tehnologice la care se referă invenția brevetată [105]***. Notația claselor în clasificarea IPC are următoarea forma: X NN X N / NN, în care pornind de la stânga la dreapta fiecare grup reprezintă un simbol al unei secțiuni din ierarhia de tehnologii și mai apoi subsecțiunea acestuia. O clasă se compune astfel dintr-o ierarhie de 5 nivele de subsecțiuni, fiecărei subsecțiuni fiindu-i asociat un simbol.

Multe dintre clasele IPC au corespondent în mai multe domenii tehnologice, astfel că un brevet poate fi încadrat la fel de bine în oricare din aceste clase, alegerea finală aparținând experților umani. Din acest motiv s-a și ajuns la soluția folosirii pentru brevete a unei clase principale și a unei liste de clase secundare.

Clasele brevetelor sunt inițial alese de către persoanele care întocmesc brevetul, de obicei inventatorii, și mai apoi sunt validate sau modificate de către reprezentanţii oficiilor de brevetare în momentul în care se face verificarea brevetului aplicat. Alegerea claselor, mai ales a claselor principale, este făcută de cele mai multe ori în funcție de domeniul tehnologic în care va fi aplicată invenția și nu neapărat în funcție de conținutul brevetului. Astfel același brevet poate la fel de bine să aparțină atât unei clase cât și a alteia dintr-un domeniu adiacent. Alegerea este de multe ori făcută după considerente subiective astfel încât încadrarea brevetului nu este întotdeauna făcută în cea mai potrivită clasă.

Prin însăși natura de noutate adusă de către brevete există, în timp, posibilitatea apariției unei noi ramuri tehnologice, neclasificată până în prezent, ce necesită o nouă clasă în nomenclatorul de clase. Dacă s-ar efectua o grupare în funcție de similaritatea conținutului brevetelor existente, este posibil să se poată identifica mai ușor aceste noi industrii, respectiv noi clase de brevete.

Identificarea industriei la care aparține un brevet este esențială pentru relevanța practică a unei invenții. Un proces de sinteză chimică inventat în industria auto va fi improbabil să fie util în industria farmaceutică, din cauza nevoilor și reglementărilor diferite. Un protocol de transfer de date pentru industria aerospațială este mai puțin relevant pentru industria calculatoarelor din cauza focalizării pe siguranță în loc de performanță. Pe de altă parte, o invenție mecanică pentru industria electronică de consum poate include soluții tehnice în mai multe clase de brevete, inclusiv mecanică, electronică, chiar algoritmi informatici. În acest caz, sistemul de clasificare existent poate determina un obstacol în accesarea invenției de către competitori, prin folosirea unor clase mai puțin relevante. Sistemele de clasificare ale brevetelor bazate pe ramuri ale științei și tehnologiei au fost criticate din cauza focalizării pe aspecte tehnice în detrimentul focalizării pe necesitățile utilizatorilor.

În [106] au fost făcute studii în care s-a urmărit predicția și identificarea de noi tehnologii ce pot apărea în industrie. Studiul se bazează pe rețeaua de citări între brevetele din baza de date de brevete US și pe ideea că graful de citări între brevete oferă o reprezentare a procesului de inovare. Brevetele citate anterior reprezintă mici părți de cunoștințe pe care se bazează noile brevete. Pe baza acestui graf de citări autorii au identificat o serie de clustere și mai apoi au făcut predicții referitoare la evoluția temporală a acestor clustere.

În cazul unei căutări „prior-art”, datorită diversității tipurilor de clasificări existente, munca inventatorului ar putea fi îngreunată, iar o clasificare independentă de sistemele existente i-ar ajuta la găsirea de referinţe relevante.

Din aceste motive se poate introduce clusterizarea în brevetele de invenţie care va realiza această clasificare automat, fără intervenţia manuală a utilizatorului şi fără a ţine cont de stricteţea vreunui sistem de clasificare.

3.2 Clusterizarea brevetelor de invenție

Spre deosebire de documentele text simple, în care toate cuvintele au aceeași importanță, iar ponderea lor se calculează cu aceeași formulă, în cazul paginilor web, a documentelor științifice sau a brevetelor de invenție, lucrurile stau diferit. Acestea din urmă poartă în structura lor informații suplimentare ce pot fi folosite la îmbunătățirea clusterizării. În cazul paginilor web acestea se găsesc în metadata. Cuvintele cheie din metadata paginilor web, constituie un set de cuvinte cu o importanță mai mare decât aceleași cuvinte ce s-ar putea regăsi în corpul paginii. Tot în cazul paginilor web, un alt parametru cu o valoare deosebită îl constituie URL-ul paginii. Acesta poate fi folosit la gruparea paginilor pe domenii.

În cazul brevetelor de invenție, există o serie de secțiuni de o importanță deosebită și, așa cum s-a arătat în capitolul 1.1.1, acestea se regăsesc în metadatele brevetelor. Printre aceste secțiuni putem număra numele aplicantului, numele inventatorului și, nu în ultimul rând, clasificarea existentă folosită în brevete.

În practică, în cazul paginilor web, nu e recomandată folosirea cuvintelor cheie din metadata, deoarece utilizatorii nu descriu fidel paginile în metadata, iar în goana după poziții cât mai înalte în rezultatele motoarelor de căutare, utilizatorii folosesc metadata ca artificiu de SEO (Search Engine Optimization) și nu ca un parametru de încredere ce poate fi folosit ca relevanță. Drept urmare, motoarele de căutare la ora actuală nu mai acordă o pondere foarte mare acestor date atunci când calculează rangul paginilor.

În cazul articolelor științifice, folosirea eficientă a parametrilor suplimentari într-o clusterizare, întâmpină probleme de consistență. Există foarte multe tipuri de structuri diferite pentru articole. Un simplu exemplu este faptul că nu toate articolele sau lucrările științifice au un set de cuvinte cheie disponibil.

În cazul brevetelor lucrurile stau însă diferit. Brevetele sunt controlate și verificate de mai multe ori, la diferite niveluri, iar datele înregistrate sunt precise și corecte. Astfel există garanția că parametrii suplimentari ce se găsesc în metadata brevetelor au o acuratețe mare și reprezintă o informație de încredere. Aceste informații din metadata, datorită omogenității lor, pot fi astfel mai ușor supuse unor prelucrări ulterioare.

Am amintit la începutul capitolului 2 câteva tipuri de căutări ”prior art”. În același capitol au fost explicate și câteva dintre dezavantajele motoarelor actuale de căutare pentru brevete, printre care se număra și lipsa unei ordonări după relevanță a numeroaselor rezultate obținute în urma căutărilor. Datorită numărului mare de rezultate ce se obțin într-o căutare, de multe ori este necesară o filtrare a rezultatelor pentru a obține o relevanță îmbunătățită a rezultatelor. Clusterizarea brevetelor permite filtrarea unor astfel de rezultate ale căutărilor. În loc de sute de brevete rezultate ce ar trebui examinate unul câte unul, rezultatele pot fi împărţite în grupuri de brevete cu conţinut similar. Relaţiile dintre brevete devin rapid vizibile atunci când sunt afişate grupuri de documente similare pe baza cuvintelor extrase din brevete [107].

În continuare, sunt descriși algoritmii ce stau la baza obținerii de clustere ce vor fi ulterior folosite în motorul de căutare propus în capitulul 4. Mai întâi, este prezentat algoritmul de clusterizare cel mai frecvent folosit în clusterizarea textului și, mai apoi, este propusă o variantă

nouă a acestuia adaptată la cazul particular al brevetelor, în care ponderile sunt calculate diferit pentru cuvintele din descriere și pentru cuvintele din metadata brevetelor.

3.2.1 Clusterizarea brevetelor de invenție cu algoritmul k-means

Dată fiind dimensiunea mare a bazelor de date de brevete pe care se va aplica clusterizarea, algoritmul de clusterizare de la care pornim și care se pretează cel mai bine în această situaţie este k-means.

Ideea de bază a clusterizării k-means este că obiectele de clusterizat, în cazul de față brevetele de invenție, sunt grupate în k clustere astfel încât toate obiectele din acelaşi cluster sunt similare într-un grad cât mai mare, iar obiectele care nu sunt în acelaşi cluster sunt cât mai diferite posibil. Pentru a calcula similaritatea și diferența dintre obiectele clusterelor se folosesc diferite metrici. Unul dintre conceptele importante în k-means este centroidul (centrul de greutate al clusterului): fiecare cluster are un centroid, care este considerat ca fiind obiectul cel mai reprezentativ al clusterului.

Performanţa algoritmului k-means este influenţat de numărul şi de parametrii obiectelor inițiale alese ca centroizi ai clusterelor.

Cel mai utilizat model pentru documentele text pe care se bazează algoritmul k-means este modelul spațiului de vectori. În cadrul acestui model se construiește o matrice de ponderi, unde liniile reprezintă atributele, iar coloanele obiectele. În cazul documentelor de tip text, liniile sunt reprezentate de cuvintele din text, iar coloanele reprezintă documentele (în secțiunea 1.4.2.4 este detaliat algoritmul k-means).

Se definește astfel matricea *M*:

$$M = \begin{pmatrix} w_{11} & \cdots & w_{1j} & \cdots & w_{1n} \\ \vdots & \ddots & \vdots & & \vdots \\ w_{i1} & \cdots & w_{ij} & \cdots & w_{in} \\ \vdots & & \vdots & \ddots & \vdots \\ w_{w1} & \cdots & w_{wj} & \cdots & w_{wn} \end{pmatrix} \qquad 3.1$$

unde w_{ij} reprezintă ponderi calculate cu TF-IDF (produsul dintre frecvența termenilor (tf) și frecvența inversă a documentului (idf), termeni detaliați în capitolul 1.4.2.4). *n* reprezintă numărul de documente, iar *w* numărul de cuvinte extrase din întreg setul de documente.

În prezent, există mai multe sisteme de ponderare utilizate pentru calculul măsurilor tf_{ij} și idf [100]. Unul dintre cele mai comune şi eficiente astfel de sisteme de ponderare utilizat pentru clusterizarea documentelor de tip text este definit mai jos (3.2 și 3.3). Aceeași schemă de calcul a fost utilizată și în prezenta lucrare pentru clusterizarea brevetelor.

$$w_{ij} = tf_{ij} * idf_i$$

$$tf_{ij} = c_1 + (1 - c_1) * \frac{n_{ij}}{max_i\, n_{ij}} \qquad 3.2$$

$$idf_i = c_2 + log\frac{n - n_{word(i)}}{n_{word(i)}} \qquad 3.3$$

unde n_{ij} este numărul de apariţii a cuvântului *i* în textul *j*, $max_i\, n_{ij}$ este numărul cel mai mare de apariţii a unui cuvânt în textul *j*, iar c_1 şi c_2 sunt constante.

După stabilirea modelului și după calcularea matricei ponderilor, următorul pas este clusterizarea efectivă, cu algoritmul k-means. Pe parcursul rulării algoritmului, la fiecare pas sunt calculați, după cum am precizat mai sus, centroizii fiecărui cluster.

În abordarea de față, centroizii sunt considerați obiecte generice care, pentru fiecare cluster, se calculează ca medie a atributelor fiecărui obiect. Având în vedere faptul că obiectele în cazul nostru sunt brevetele, iar atributele sunt cuvintele din brevete, centroizii sunt documente generice care au ca valori media ponderilor cuvintelor conținute de brevetele din fiecare cluster.

În expresia 3.4 este definit centroidul C_t corespunzător clusterului *t*, ca fiind documentul generic ce ia ca valori pentru fiecare linie *i*, media ponderilor conținute pe linia *i* de brevetele din clusterul *t* (cu mențiunea că fiecărei linii *i* îi corespund ponderile unui cuvânt în fiecare din cele *n* brevete).

$$\mathrm{C}_t = \begin{pmatrix} \frac{1}{|S_{t_1}|}\sum_{w_{1j}\in S_{t_1}} w_{1j} \\ \vdots \\ \frac{1}{|S_{t_i}|}\sum_{w_{ij}\in S_{t_i}} w_{ij} \\ \vdots \\ \frac{1}{|S_{t_w}|}\sum_{w_{wj}\in S_{t_w}} w_{wj} \end{pmatrix} \qquad 3.4$$

În expresia 3.4, S_t reprezintă mulțimea tuturor brevetelor din clusterul *t*, iar S_{t_i} mulțimea ponderilor corespunzătoare liniei *i* din toate brevetele ce aparțin clusterului *t*.

Un pas important în aplicarea algoritmilor de clusterizare este alegerea funcției de similaritate sau funcția distanță folosită pentru gruparea obiectelor în clustere. Pentru funcții de similaritate diferite, se pot obține rezultate diferite ale clusterizării. Pentru clusterizarea brevetelor s-a ales ca măsură de calcul al similarității funcția cosinus, aceasta fiind una dintre cele mai folosite funcții distanță în cazul clusterizării documentelor text. De-a lungul timpului funcția cosinus și-a dovedit eficiența în calculul similarității clusterelor de tip text. Există studii comparative care o plasează printre primele opțiuni în vederea folosirii în cadrul algoritmilor de clusterizare a textului [110], [111].

Se calculează astfel cosinusul unghiului dintre doi vectori, fiecare reprezentând câte un document.

$$similar(q,d) = \cos(q,d) = \frac{q \bullet d}{|q|\,|d|} = \frac{\sum_i w_{i,q}\, w_{i,d}}{\sqrt{\sum_i {w_{i,q}}^2}\ \sqrt{\sum_i {w_{i,d}}^2}} \qquad 3.5$$

unde *q* și *d* reprezintă vectorii a două documente pentru care se calculează similaritatea.

Rezultatele clusterizării k-means obținute cu algoritmul descris mai sus sunt ulterior integrate în motorul de căutare propus în capitolul 4.

În urma clusterizării folosind algoritmul k-means prezentat, s-au obținut clustere ce conțin brevete cu un conținut asemănător. După integrarea clusterelor în motorul de căutare, se pot face căutări în urma cărora rezultatele să poată fi prezentate sub formă de grupe de brevete similare, putându-se ulterior rafina căutarea în funcție de interesul arătat pentru un anumit grup relevant de brevete. Un utilizator ce va face astfel de căutări va putea alege din zecile de rezultate numai grupul de brevete pe care îl va considera relevant domeniului său de expertiză.

Clusterizarea efectuată cu algoritmului k-means prezentat nu ia însă în considerare și informațiile suplimentare disponibile în metadata. Chiar dacă acestea ar fi totuși introduse în dicționarul de cuvinte obținut pentru întregul set de brevete, conform algoritmului k-means prezentat, cuvintele din metadata brevetelor vor avea ponderea calculată în același fel ca a oricărui alt cuvânt din descrierea brevetelor.

3.2.2 Contribuții privind clusterizarea documentelor text folosind parametri cu ponderi diferite

În continuare, am propus un nou model pentru documentele de tip brevete de invenție [112].
Pentru un brevet de invenție este definit 3-uplul:

$$P = \langle P_c, P_m, M_n \rangle$$

unde $P_c = \{wn_1, wn_2, \dots wn_{nc}\}$ este setul de cuvinte extras din descrierea brevetului și *nc* este numărul de cuvinte din set, $P_m = [mv_1, mv_2, \dots mv_{nm}]$ este vectorul de valori metadata din brevet și *nm* este numărul de câmpuri metadata din brevet, $M_n = [mn_1, mn_2, \dots mn_{nm}]$ este vectorul ce conține numele câmpurilor metadata din brevet.

Se definește *PDB*, întregul set de brevete din baza de date, după cum urmează:

$$PDB = \{P_1, P_2, \dots P_n\}$$

unde *n* este numărul brevetelor din baza de date.

Dicționarul de cuvinte format din conținutul întregului set de brevete, *DC*, se definește astfel:

$$DC = \{wn_i \in P_{c_1} \cup P_{c_2} \dots \cup P_{c_n}\}$$

Fie $nd = |DC|$ numărul de cuvinte din dicționarul *DC*.

În același fel definim dicționarul de cuvinte pentru fiecare câmp metadata *t*:

$$DM_{mn_t} = \{mv_{t_i} \in P_{m_1}[t] \cup P_{m_2}[t] \dots \cup P_{m_n}[t]\}$$

unde $t = 1 \dots nm$ și $P_{m_k}[t]$ este al *t*-ulea element din vectorul P_{m_k} din fiecare brevet P_k.

Fie $nmn_t = |DM_{mn_t}|$ numărul de elemente din DM_{mn_t}.

În matricea *M* definită în capitolul anterior pentru modelul spațiului de vectori (expresia 5.1), toate ponderile sunt calculate cu aceeași funcție TF-IDF. În acest capitol, am propus o nouă matrice pentru modelul spațiului de vectori compusă dintr-un set de subseturi de atribute. Fiecare subset conține ponderi calculate cu o funcție TF-IDF diferită. În expresia 3.6 este definită această matrice.

$$MP = \begin{pmatrix} MC \\ MM_{mn_1} \\ MM_{mn_2} \\ \dots \\ MM_{mn_{nm}} \end{pmatrix} \quad 3.6$$

unde

$$MC = \begin{pmatrix} wwn_{11} & \cdots & wwn_{1j} & \cdots & wwn_{1n} \\ \vdots & \ddots & \vdots & & \vdots \\ wwn_{i1} & \cdots & wwn_{ij} & \cdots & wwn_{in} \\ \vdots & & \vdots & \ddots & \vdots \\ wwn_{nd\,1} & \cdots & wwn_{nd\,j} & \cdots & wwn_{nd\,n} \end{pmatrix} \quad 3.7$$

și

$$MM_{mn_t} = \begin{pmatrix} wwv_{t_{11}} & \cdots & wwv_{t_{1j}} & \cdots & wwv_{t_{1n}} \\ \vdots & \ddots & \vdots & & \vdots \\ wwv_{t_{i1}} & \cdots & wwv_{t_{ij}} & \cdots & wwv_{t_{in}} \\ \vdots & & \vdots & \ddots & \vdots \\ wwv_{t_{nmn_t\,1}} & \cdots & wwv_{t_{nmn_t\,j}} & \cdots & wwv_{t_{nmn_t\,n}} \end{pmatrix} \quad 3.8$$

Matricea combinată *MP* din expresia 3.6 are $nd + \sum_{t=1..nm} nmn_t$ linii și *n* coloane. Liniile matricei *MP* reprezintă ansamblul cuvintelor din dicționarul format din conținutul tuturor brevetelor și cuvintele din dicționarul format din metadata brevetelor. Coloanele matricei *MP* reprezintă brevetele.

3.2.3 Alegerea funcțiilor de ponderare

Ideea aflată la baza modelului propus în prezenta lucrare este că informațiile conținute în metadata brevetelor pot aduce informaţii suplimentare în clusterizarea brevetelor. Putem considera, de exemplu, informațiile din metadata "Applicant" în modelul prezentat în secțiunea anterioară. Realizând o clusterizare folosind acest model, avem ca scop gruparea brevetelor similare ca text şi, în plus, de a adăuga în fiecare grup brevetele relevante ale aplicanților ce aparţin fiecărui grup în parte.

De obicei, aplicanții sunt specializați într-un singur domeniu tehnic, deci este foarte probabil ca brevetele care aparţin unui aplicant să fie din acelaşi domeniu de activitate. Analizând baza de date de brevete am observat că aplicanții au toate sau marea majoritate a brevetelor aplicate într-o singură categorie. Prin urmare, brevetele aceluiaşi aplicant ar trebui să fie în aceeaşi grup sau în grupe apropiate.

3.2.3.1 Analiză pe marginea alegerii soluției celei mai potrivite

La prima vedere, gruparea făcută exclusiv după aplicant, ar trebui să fie suficientă, iar clusterizarea nu ar mai fi necesară. Însă, nu toate brevetele unui aplicant sunt relevante pentru o interogare de căutare, astfel încât doar un set specific de brevete aparținând unui aplicant este relevant.

Pentru a avea o importanță mai mare a metadata ”Applicant” în similitudinea brevetelor, influenţa aplicanților ar trebui să fie controlată și mai puternică decât cea a cuvintelor obişnuite din descrierea unui brevet. Pentru a realiza acest lucru, am utilizat funcții TF-IDF diferite pentru fiecare subset de date.

În modelul spaţiului vectorial clasic (matricea 3.1), folosind schemele de ponderare TF-IDF 5.2 și 5.3, importanţa unui termen într-un document este cu atât mai mare cu cât valoarea frecvenţei termenului este mai mare într-un număr mai limitat de documente. Două documente sunt mai asemănătoare, dacă au mai mulţi termeni în comun, iar frecvenţele acestor termeni au o valoare mare și mai apropiată între ele. Pe de alte parte, cu cât un termen apare mai des în întreg setul de documente, cu atât este mai diminuată importanța acestuia în calculul ponderilor.

Din punct de vedere al relevanței căutărilor în brevete, în cazul nostru particular al aplicanților, importanța unui aplicant este cu atât mai mare cu cât acesta a aplicat mai multe brevete. Dacă pentru aplicanți s-ar aplica schemele de ponderare TF-IDF clasice 5.2 și 5.3 efectul asupra aplicanților cu un număr mare de brevete aplicate ar fi invers decât s-ar dori. Numărul mare de apariții al acestor aplicanți ar conduce la diminuarea importanței acestora în setul de date, ori noi dorim exact contrariul.

Astfel, pentru a îndeplini cele două necesități descrise mai sus, și anume, că influența aplicanților ar trebui să fie controlată și mai puternică pe măsură ce aceștia au aplicat mai multe brevete, ponderile wwn_{ij} din expresia 3.7 și ponderile $wwv_{t_{ij}}$ din expresia 3.8 sunt calculate ca produsul dintre TF și IDF, însă după cum vom arăta în cele ce urmează, funcția TF este diferită pentru fiecare set de ponderi:

$$wwn_{ij} = tf_{ij} * idf_i$$

$$wwv_{t_{ij}} = tf_{t_{ij}}{}' * idf_i$$

Pentru calculul ponderilor wwn_{ij} a fost folosită schema de calcul descrisă în capitolul anterior, mai exact expresiile 3.2 și 3.3.

Pentru a identifica corect avantajele modelului propus, în cele ce urmează vom particulariza modelul brevetelor descris in aceasta secțiune pentru cazul în care luam în considerare doar metadata “Applicant”. Influenţa termenilor din metadata "Applicant" poate fi modificată dacă modificăm funcția TF după cum urmează:

Termenul n_{ij} din expresia 3.2, ce reprezintă numărul de apariții a cuvântului *i* în documentul *j*, a fost schimbat cu $n_{ij}{}'$:

$$n_{ij}{}' = c_3 * \frac{\ln(n_{word(ij)})}{\max(\ln(n_{word(ij)}))} \qquad 3.9$$

unde $n_{word(ij)}$ este numărul de documente în care apare termenul *i*, dacă termenul *i* apare în documentul *j*. Mai exact, reprezintă un logaritm natural din numărul de brevete aplicate de aplicantul *i* daca *i* este aplicant în brevetul *j*. $\max(\ln(n_{word(ij)}))$ reprezintă maximul logaritmilor naturali ai numărului de documente în care apare un termen din metadata ”Applicant”, altfel spus, reprezintă logaritmul natural din numărul de brevete ale aplicantului cu cel mai mare număr de brevete. c_3 este o constantă de ponderare. Astfel tf_{ij} devine:

$$tf_{ij}{}' = c_1 + (1 - c_1) * \frac{n_{ij}{}'}{max_i\, n_{ij}{}'} \qquad 3.10$$

Folosind expresia 3.9 semnificația frecvenței din expresia 3.2 a fost modificată. Astfel frecvența termenilor din documente, sau, mai exact, frecvența termenilor din metadata ”Applicant” a fost modificată din 1 în logaritmul numărului de brevete în care apare un anumit aplicant. Este folosit logaritmul natural pentru că se dorește evitarea apariției unei influențe foarte mari din partea unor aplicanți cu un număr mare de brevete (există aplicanți cu sute de brevete).

Introducând în calculul funcției TF-IDF numărul de brevete aplicat de către aplicanți în întreg setul de date (mai exact logaritm natural din numărul de brevete în care apare un anumit aplicant), s-a inversat astfel semnificația termenului ”Applicant” în matricea modelului spațiului de vectori. Astfel, o frecvență mare a termenului în setul de documente nu va mai însemna o diminuare a influenței termenului în calcul, ci dimpotrivă o creștere a influenței.

Constanta c_3 este folosită pentru ponderarea importanței aplicanților. Prin creșterea valorii c_3 , contribuția aplicanților în model crește.

Pentru c_3 se poate alege o valoare în funcție de frecvențele calculate în matricea *MC*. Într-o abordare, c_3 poate fi determinat ca maximul valorilor frecvențelor termenilor din toate documentele (din descrierile tuturor brevetelor). O abordare similară ar putea fi alegerea lui c_3 ca medie a maximului frecvențelor termenilor din fiecare document. În continuare, a fost aleasă ca valoare pentru constanta c_3 maximul frecvențelor termenilor din toate documentele, ponderat cu constanta c_4.

$$c_3 = c_4 * \max(n_{ij}) \qquad 3.11$$

Modificând valoarea constantei c_4 este controlată importanța câmpului metadata ”Applicant” referitor la întreg setul de date.

Valoarea constantei c_4 se poate determina experimental în funcție de cât de mult se dorește influențarea rezultatelor clusterizării de către aplicanți.

Experimental s-a constatat că, dacă se aplică o ponderare prea mare asupra aplicanților, în urma clusterizării e posibil să se obțină un cluster cu foarte multe brevete, disproporționat ca dimensiune față de celelalte clustere, în care vor fi atrași cei mai mari competitori din varii domenii. Acest lucru ar trebui evitat prin alegerea unei ponderări mai mici aplicate aplicanților. De asemenea, o ponderare prea mică aplicată pe termenii reprezentați de aplicanți va conduce la o influență neglijabilă față de clusterizarea clasică în care nu sunt folosite metadatele. Trebuie astfel aleasă o valoare de echilibru care să evite problemele menționate anterior.

3.3 Concluzii

Folosind clusterizarea în cazul brevetelor de invenție s-a putut genera un set de grupe de brevete cu un conținut apropiat, eliminându-se astfel dezavantajul folosirii exclusive a clasificării manuale. S-au obținut astfel grupuri de brevete independente de sistemul de clasificare al brevetelor existent.

Prin modificarea și introducerea în calculul matricei din modelul spațiului de vectori a unor seturi de atribute suplimentare, a fost posibilă includerea informațiilor metadata din brevete în clusterizare. Prin folosirea unor funcții diferite TF-IDF de calcul al ponderilor pentru diferite seturi de atribute s-a putut controla influența fiecărui set de atribute asupra similarității dintre brevete.

Folosind algoritmul k-means cu modelul propus în capitolul 0, în cazul particular în care s-a ales metadata ”Applicant”, avantajele clusterizării, care generează documente cu conţinut apropiat şi avantajele grupării de brevete ale unui anumit aplicant au fost combinate, obținându-se astfel grupuri de brevete ce pot reprezenta rezultate relevante pentru utilizatorilor care efectuează operații de căutare.

Prin controlarea separată a influenţei atributelor din câmpul metadata ”Applicant”, se obțin clustere care conţin preponderent brevetele relevante ale aplicanților din cluster, şi nu toate brevetele aplicanților din cluster. Cu cât este mai mare influența aplicanților în clusterizare, cu atât mai multe brevete ale aceluiaşi solicitant apar în acelaşi cluster, în același timp însă scăzând similitudinea dată exclusiv de conţinutul brevetelor.

Modelul este de asemenea aplicabil și altor metadate, aici putând fi menționate titlul sau inventatorul. De exemplu, o mai mare importanţă pot fi date cuvintelor din titlu, astfel brevetele ce conţin cuvinte similare în titlu va putea fi considerate mai apropiate, chiar dacă acestea au un conţinut al brevetelor mai puțin similar.

4 Contribuții aplicative

Pentru punerea în valoare și pentru validarea experimentală a teoriei expuse în prezenta lucrare, am realizat un motor de căutare ce are la bază calculul rangului brevetelor prin metodele prezentate. Motorul de căutare a fost realizat ca aplicație web pentru a putea fi ușor accesat de oriunde din Internet.

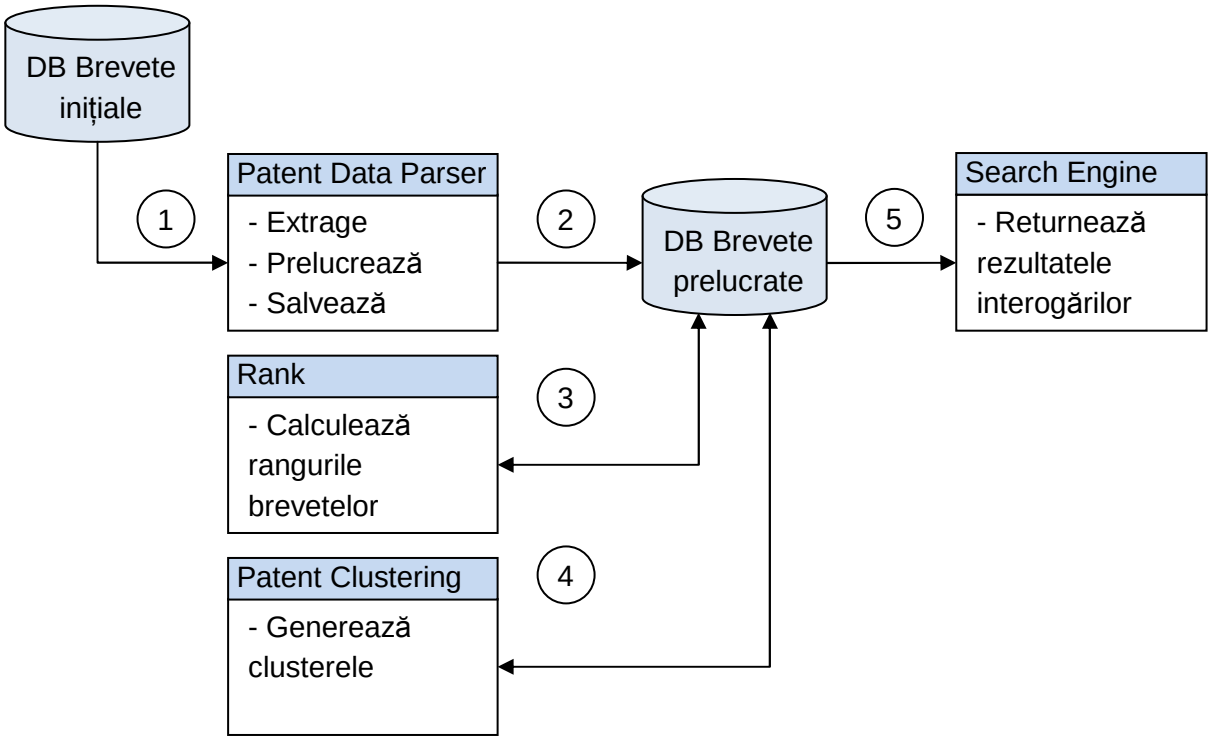

Figura 2 Arhitectura întregului motor de căutare propus

Atât pentru calculul diferitelor ranguri necesare motorului de căutare, cât și pentru diferitele forme de clusterizare a brevetelor au fost realizate câteva aplicații auxiliare.

Aplicația web corespunzătoare motorului de căutare a fost proiectată de la început scalabil, astfel încât să poată fi extinsă la nivel comercial, având în spate date reale extrase din baze de date cu brevete. Proiectând de la început aplicația în acest mod, ulterior se pot face actualizări ale bazei de date prin adăugarea ultimelor brevete apărute.

Prelucrarea datelor în baza de date, calculul rangurilor și clusterizările sunt realizate de către aplicații separate, neintegrate cu motorul de căutare. Aplicația pentru motorul de căutare este de sine stătătoare, însă se bazează pe toate datele prelucrate de către aplicațiile auxiliare. S-a ales această soluție deoarece atât algoritmii de calcul al rangurilor cât și algoritmii de clusterizare, aplicați pe o bază de date de dimensiunile celei folosită în prezenta lucrare, sunt mari consumatori de memorie și timp. Și pentru a asigura continuitatea serviciului de căutare, s-a ales ca toate prelucrările datelor să se realizeze independent.

Având în vedere că bazele de date de brevete au o rată a creșterii în volum cu mult mai mică decât bazele de date web (câteva zeci de noi brevete apar săptămânal, spre deosebire de paginile web care apar cu zecile de mii zilnic), am considerat că putem calcula rangurile și clusterizările cu o frecvență scăzută (o dată pe lună). Drept urmare s-a putut alege soluția în care, mai întâi, sunt calculate rangurile și sunt realizate clusterizările pe o unitate de calcul locală și, mai apoi, sunt utilizate în motorul de căutare online. Pentru calculele făcute pe unitatea locală sunt folosite astfel aplicațiile auxiliare amintite mai sus.

Arhitectura întregului sistem este sintetizată în Figura 2.

4.1 Setul de date

Pentru experimentare a fost folosită baza de date European Patent Office (EPO) ce conține toate brevetele de invenție europene acceptate în perioada 09 ianuarie 1980 – 29 decembrie 2004. Anual apar actualizări la această perioadă ce conțin noile brevete acceptate. Baza de date folosită în studiu cuprinde 707594 brevete de invenție.

În operațiile de clusterizare au fost folosite informațiile din metadata brevetelor, iar din conținutul brevetelor s-a folosit doar secțiunea de revendicări (claims) ce se presupune a fi o sinteză a descrierii brevetului. În urma rulării algoritmilor de clusterizare propuși, sunt așteptate rezultate apropiate calitativ celor obținute în cazul în care s-ar fi folosit întregul conținut text al unui brevet (inclusiv abstractul și descrierea). În studiul [113] au fost făcute teste de clasificare atât pentru abstract, cât și pentru întregul text al brevetelor, iar una dintre concluzii a fost că precizia clasificării creste cu doar 2-9% în cazul în care se folosește întregul text din brevete decât în cazul în care au fost folosite doar rezumatele.

4.2 Pregătirea datelor

În forma lor originală, informațiile referitoare la brevetele EPO se găsesc într-un format de bază de date propriu EPO. Această bază de date poate fi citită cu o aplicație desktop pusă la dispoziție de EPO, Espace Mimosa V5[1]. Din această aplicație au putut fi exportate toate informațiile despre brevete în format text. Pentru trecerea din format text într-un format ce poate fi folosit pentru prelucrări ulterioare, am dezvoltat o aplicație care prelucrează datele obținute în formatul text exportat din aplicația EPO și le salvează într-o bază de date MySQL. Înainte de a fi salvate în baza de date, datele trec printr-o primă operație de filtrare, și anume, pentru fiecare câmp din brevetele extrase s-au eliminat caracterele ne-printabile.

4.2.1 Descrierea aplicației Patent Data Parser (PDP)

Aplicația PDP a fost dezvoltată în Java și conține două interfețe. Prima este compusă din două secțiuni: una în care se introduc datele de conectare la baza de date și una din care se poate selecta numele fișierului care va fi prelucrat. După introducerea acestor date se poate executa operația de prelucrare a datelor propriu-zisă apăsând butonul ”*Extract data from file to DB*” din interfață.

[1] Espace Mimosa V5 – Aplicație desktop dezvoltată pentru efectuarea de diverse operații asupra bazelor de date în format specific EPO. 2005

Fișierul text introdus ca parametru în secțiunea *Files*, reprezintă fișierul cu datele brute exportate direct din baza de date originală EPO.

Cea de-a doua interfață a aplicației PDP conține doar două butoane. Primul lansează în execuție operația de pregătire a tabelelor din baza de date pentru calculul rangurilor, iar cel de-al doilea buton execută operația de identificare și eliminare a citărilor în buclă (brevet ce citează alt brevet care la rândul lui citează primul brevet).

În prezentul studiu, la calculul rangurilor și în clusterizări s-au folosit doar o parte din câmpurile extrase cu ajutorul acestei aplicații auxiliare.

4.2.2 Descrierea algoritmilor ce stau la baza prelucrării datelor inițiale

După cum s-a arătat mai sus, aplicația PDP permite prelucrarea datelor exportate din baza de date EPO într-o bază de date nouă în care se pot face prelucrări ulterioare. Algoritmul presupune execuția următoarelor acțiuni:

- Se deschide fișierul cu datele exportate din baza de date EPO
- Se deschide o conexiune la baza de date nouă
- Se citește linie cu linie fișierul până la sfârșitul acestui și se extrag datele pentru fiecare brevet în parte
- Sunt salvate în baza de date datele extrase

Tot în cadrul acestei aplicații auxiliare remarcăm încă două operații specifice pregătirii inițiale ale datelor, și anume, pregătirea tabelelor din baza de date și identificarea și eliminarea citărilor în buclă.

În cadrul operației de pregătire a tabelelor sunt implicate următoarele acțiuni:

- Se deschide o conexiune la baza de date
- Se șterge întregul conținut din tabelele de lucru
- Se copiază datele din tabelele cu datele extrase din baza de date EPO în tabele de lucru. Datele extrase din baza de date EPO sunt păstrate în forma lor nemodificată, datele copiate în tabelele de lucru fiind cele asupra cărora se aplică prelucrări. Acest lucru permite verificarea calității prelucrării prin compararea formei originale și formei prelucrate a informației.

4.3 Aplicația de calcul a rangului

Pentru calculul rangurilor folosind tehnicile descrise în partea teoretică din capitolul 2 am dezvoltat o aplicație web, cu ajutorul căreia vor fi generate diferitele tipuri de ranguri pentru brevete. Aplicația are patru secțiuni după cum urmează:

- **Computing simple rank**
- **Computing time-based rank**
- **Clean applicants**
- **Computing applicant-based rank**

4.3.1 Secțiunea Computing simple rank

Această secțiune permite calcularea rangului simplu descris teoretic în capitolul 2.1. În cadrul acestei secțiuni sunt disponibile următoarele acțiuni:

- Generarea și / sau inițializarea tabelelor necesare calculării rangului
- Pentru fiecare brevet în parte se calculează numărul de brevete citate de acesta (ex. brevetul A citează 2 alte brevete)
- Pentru fiecare brevet în parte se calculează numărul de brevete care citează brevetul curent (ex. brevetul A este citat de către alte 3 brevete)
- Inițializarea variabilelor folosite în calculul rangului, inclusiv setarea valorilor inițiale la primul pas din algoritmul de calcul
- Calcularea iterativa a rangului
- Salvarea rangului calculat

Datorită volumului mare de informații de prelucrat, s-au folosit intensiv tabelele din bazele de date. Dacă pentru inițializarea tabelelor și a valorilor inițiale din algoritm pașii sunt clari, pentru calculul iterativ efectiv al rangului s-a ales următoarea abordare:

- În primul rând s-au calculat și salvat într-o tabelă de lucru toate elementele din expresia de calcul al rangului simplu (expresia 2.2) care nu se schimbă pe parcursul rulării algoritmului: constanta *c*, raportul $\frac{1}{N_v}$ specific fiecărui brevet, numărul de brevete citate dinspre și înspre brevetul curent.
- S-a inițializat ulterior rangul la primul pas al algoritmului cu aceeași valoare pentru toate brevetele.
- În continuare algoritmul rulează astfel:

```
1  cât timp există schimbări de poziții între brevete între doi pași consecutivi
   execută
2    calculează ∑_{v∈B_p} W(v)/N_v într-o tabela temporară
3    copiază rezultatele din tabela temporară într-o coloană separată din tabela de
   lucru
4    calculează rangul pasului curent cu expresia 2.2 care devine (1-c)+c*(suma
   calculată în coloana separata în tabela de lucru)
5    salvează rangul pasului curent într-o coloană separată din tabela de lucru
6  sfârșit cât timp
```

- după ce s-a terminat calcularea iterativă a rangului (pasul anterior), ultima coloană salvată cu rangul calculat este copiată în tabela finală, în care se vor face căutările cu motorul de căutare.

4.3.2 Secțiunea Computing time-based rank

Această secțiune generează rangul ce ține cont de anul brevetării, după cum este descris în capitolul 2.2.2. În această secțiune se pot executa următoarele acțiuni:

- Generarea și / sau inițializarea tabelelor necesare calculării rangului cu corecție pe ani
- Pentru fiecare brevet în parte se calculează numărul de brevete citate de acesta (ex. brevetul A citează 2 alte brevete)

- Pentru fiecare brevet în parte se calculează numărul de brevete care citează brevetul curent (ex. brevetul A este citat de către alte 3 brevete)
- Inițializarea variabilelor folosite în calculul rangului, inclusiv setarea valorilor inițiale la primul pas din algoritmul de calcul
- Calcularea iterativă a rangului bazat pe citări și mai apoi la fiecare pas ajustarea rangului cu corecția pe ani
- Salvarea rangului calculat

Dintre toți acești pași cel mai important este cel în care se face calcularea iterativă a rangului. Acesta se bazează pe următorul algoritm:

- După inițializarea variabilelor, setarea valorilor inițiale ale rangurilor de calculat și calcularea și salvarea într-o tabelă de lucru a părților din expresia 2.7 care nu se modifică de-a lungul rulării algoritmului, la fel cum s-a procedat și în cazul calculării rangului simplu, se inițializează calculul iterativ. Trebuie menționat faptul că pentru acest algoritm parametrul *inject* (constanta *c* în cazul anterior) este calculat pentru fiecare brevet în parte în funcție de anul aplicării acestuia, folosind expresia 2.6.

4.3.3 Secțiunea Clean applicants

Această secțiune permite o serie de acțiuni ce presupun prelucrarea aplicanților în vederea folosirii lor în calculul rangului brevetelor.

- Generarea și / sau inițializarea tabelelor necesare calculării rangului ce ține cont de aplicanți
- Prelucrarea numelor aplicanților. În această etapă sunt eliminate din numele aplicanților sufixele și prefixele specifice numelor firmelor (ex. din numele firmei ”s.c. nume firma s.r.l.” sunt eliminate s.c. și s.r.l.). De asemenea s-au eliminat caractere non-text, inclusiv spatiile multiple (ex. ”, ‘, -, +, . sau ,). Multe dintre numele aplicanților în forma lor originală neprelucrată, așa cum o găsim în brevetele aplicate de-a lungul timpului, nu sunt consistente. Multe nume de aplicanți conțin în unele brevete prefixe sau sufixe. Prin prelucrările aplicate la acest pas, aceste inconsistențe sunt drastic reduse.
- După prelucrările de la pasul anterior se pot identifica brevetele și numărul acestora deținute de același aplicant. Aceste informații sunt folosite la calculul rangului ce ține cont de aplicanți

4.3.4 Secțiunea Computing applicant-based rank

În această ultimă secțiune sunt implementate funcțiile ce permit calculul rangului brevetelor ce ține cont și de aplicanți, așa cum este prezentat teoretic în capitolul 2.2.1.
Pașii necesari calculării rangului sunt:

- Generarea și / sau inițializarea tabelelor necesare calculării rangului ce ține cont de aplicanți
- Pentru fiecare brevet în parte se calculează numărul de brevete citate de acesta (ex. brevetul A citează 2 alte brevete)
- Pentru fiecare brevet în parte se calculează numărul de brevete care citează brevetul curent (ex. brevetul A este citat de către alte 3 brevete)

- Inițializarea variabilelor folosite în calculul rangului, inclusiv setarea valorilor inițiale la primul pas din algoritmul de calcul
- Calcularea iterativă a rangului bazat pe citări și mai apoi ajustarea rangului bazat pe frecvența brevetelor aplicate de aplicanții fiecărui brevet.
- Salvarea rangului calculat

4.4 Aplicația de clusterizare a brevetelor

Diferitele tipuri de clusterizări descrise teoretic în capitolul 3 au fost implementate într-o aplicație auxiliară scrisă în limbajul JAVA, numită **Patent Clustering**. La baza acestei aplicații stă implementarea algoritmului pentru clusterizare k-means aplicat pe diferite modele ale brevetelor, modele descrise în capitolul 3. Arhitectura acestei aplicații este descrisă în Figura 3 și după cum se poate observa este formată din două mari zone: prima reprezintă clusterizarea propriu-zisă și cea de-a doua constă în evaluarea rezultatelor obținute.

Această aplicație este formată dintr-o serie de secțiuni, primele trei fiind dedicate operației de clusterizare și ultimele conținând funcții adiacente clusterizării:

- **Preprocessing** – secțiune ce grupează funcțiile necesare prelucrării și pregătirii datelor pentru clusterizare
- **TF-IDF** – secțiune ce grupează funcțiile de calcul al componentei TF-IDF din calculul clusterizării brevetelor
- **Clustering** – secțiune în care se pot seta parametrii algoritmului k-means și unde se poate lansa în execuție algoritmul de clusterizare propriu-zis.
- **Results** – secțiune în care se pot evalua rezultatele clusterizării.
- **Clustering & Results** – secțiune ce permite rularea mai multor sesiuni de clusterizare în vederea alegerii rezultatului clusterizării celei mai bune.
- **Tools** – secțiune ce conține funcții de prelucrare a rezultatelor
- **Settings** – secțiune în care sunt setați parametri ai aplicației, inclusiv conectarea la baza de date

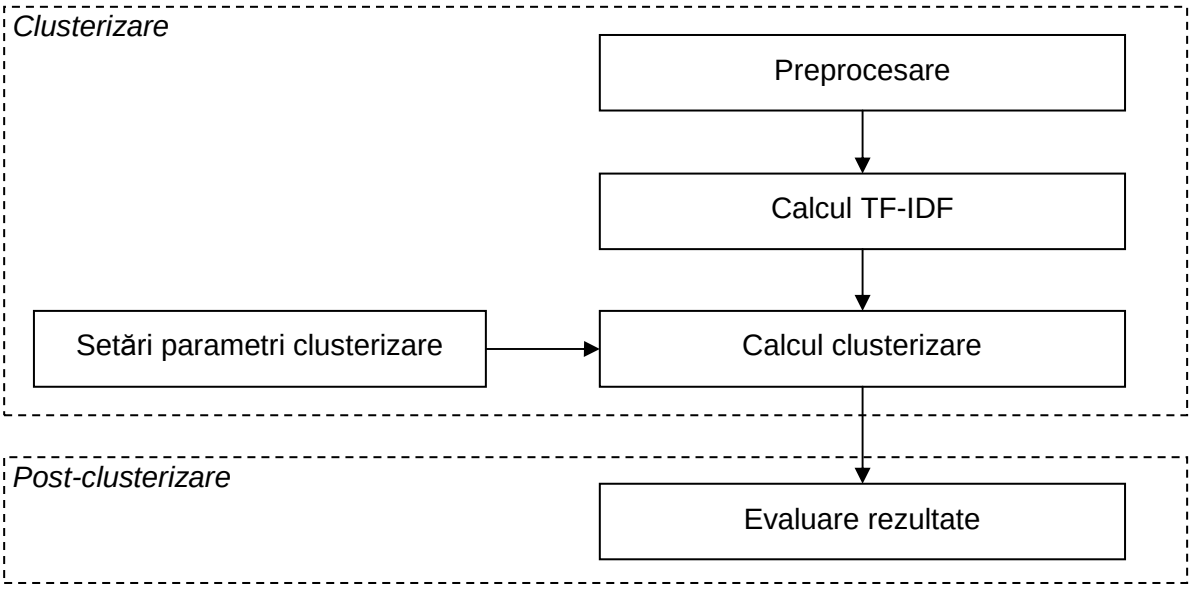

Figura 3 Arhitectura aplicației de clusterizare a brevetelor

În cele ce urmează vor fi explicate în detaliu funcțiile disponibile fiecărei secțiuni a aplicației Patent Clustering.

4.4.1 Secțiunea Preprocessing

Înaintea rulării algoritmului de clusterizare k-means propriu-zis, sunt necesare o serie de prelucrări ale documentelor ce urmează a fi clusterizate. În primul rând, trebuie generat modelul pe baza căruia se va aplica k-means. Acest lucru presupune extragerea dicționarului de cuvinte existent în setul de documente, calculul frecvențelor cuvintelor din fiecare document în vederea creării modelului spaţiului de vectori.

În această secțiune sunt grupate toate funcțiile ce se execută în baza de date pentru pregătirea datelor în vederea clusterizării. Avem astfel următoarele funcții:

- **Clear words tables** – execută o serie de interogări în baza de date care au rolul de a goli tabelele din baza de date folosite în procesul de clusterizare pregătindu-le astfel pentru o nouă rulare a clusterizării.
- **Generate dictionary and frequency matrix** – această prelucrare execută o serie de funcții ce au ca finalitate generarea dicționarului de cuvinte și a matricei de frecvență, ambele fiind componente necesare calculului din algoritmul k-means. Pașii executați sunt următorii:
    - Extrage din baza de date seturi de câte *n* brevete
    - Se extrag toate cuvintele din brevete
    - Se identifică și se adaugă la dicționar cuvintele noi, iar pentru cuvintele ce se găsesc deja în dicționar se incrementează numărul de apariții
- Aplicația permite, prin intermediul a două câmpuri **From:** și **To:**, introducerea doar a unui set de brevete spre prelucrare.
- **Generate dictionary and frequency matrix for Applicants** – această prelucrare este folosită în cazul în care se dorește luarea în calcul și a metadatelor (în cazul nostru **Applicants**) și permite extragerea și adăugarea datelor din metadata Applicants în dicționarul de cuvinte destinat acestei metadate.
- **Update word count** – calculează pentru fiecare cuvânt din dicționarul de cuvinte, numărul de documente în care acesta apare.
- **Update frequncy matrix for Applicants** – dat fiind faptul că pentru datele din metadata Applicants frecvențele per document sunt calculate în mod diferit de modelul clasic, acest pas actualizează frecvențele apariției aplicanților în documente, conform modelului de calcul specific metadata Applicant, așa cum este descris în partea teoretică a prezentei lucrări.

Întregul flux de acțiuni ce trebuie efectuate în această etapă este descris în Figura 4.

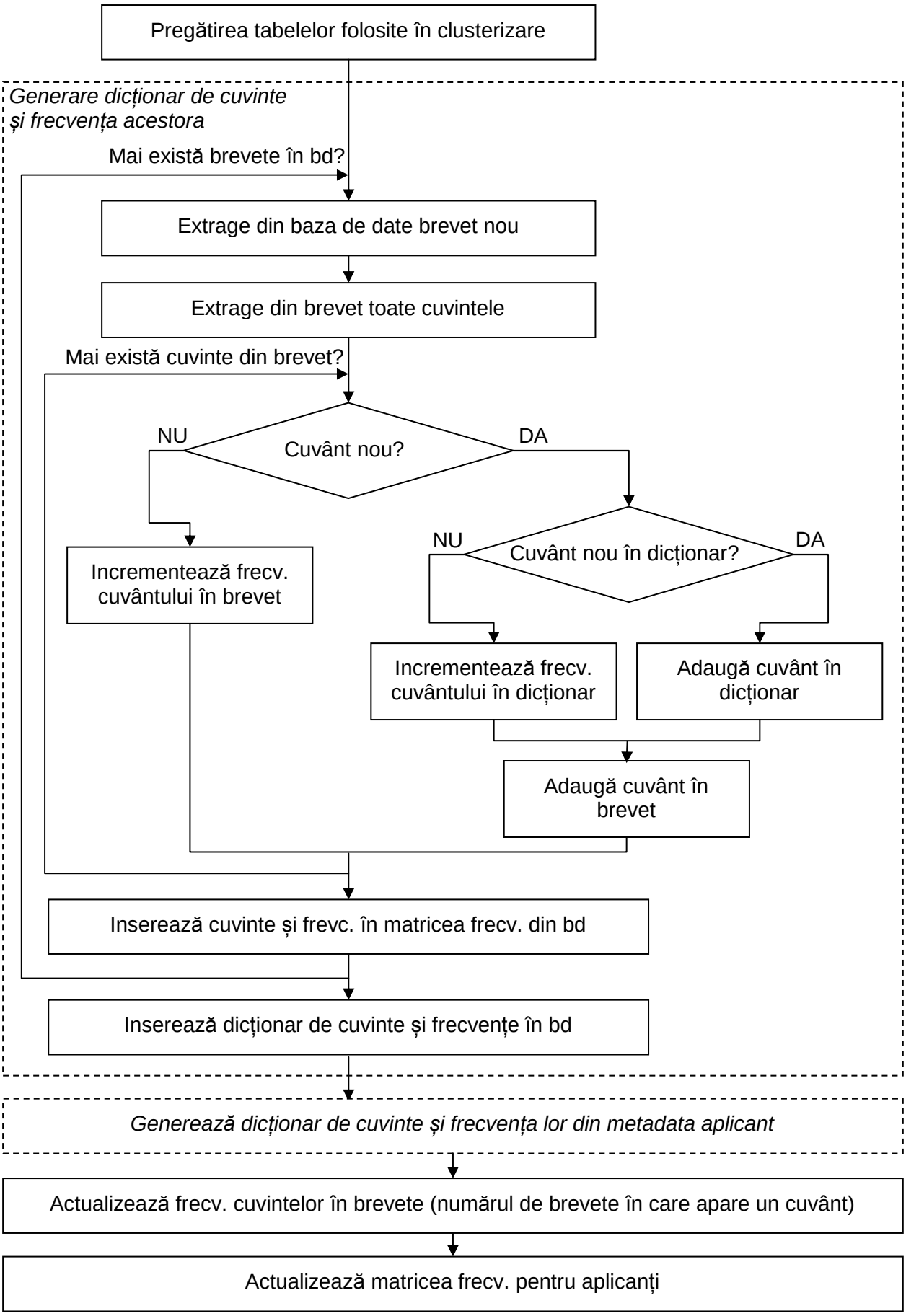

Figura 4 Fluxul de acțiuni executate pentru pregătirea datelor înaintea aplicării algoritmului k-means

Daca se dorește rularea algoritmului k-means în forma sa simplă, nu se vor mai folosi prelucrările **Generate dictionary and frequency matrix for Applicants** și **Update frequncy matrix for Applicants.**

În Figura 4 secțiunea *Generare dicționar de cuvinte și frecvența lor din metadata aplicant* are aceiași pași ca și secțiunea *Generare dicționar de cuvinte și frecvența acestora* cu următoarele diferențe:

- În loc de a extrage cuvintele din brevet se vor extrage metadata aplicant
- Frecvența unui aplicant într-un brevet este întotdeauna 1, deci se va verifica doar dacă avem un aplicant nou în dicționarul de aplicanți.

La etapa de *Generare a dicționarului de cuvinte și frecvența acestora*, pentru a crește viteza de procesare, se extrag mai întâi din baza de date seturi de mai multe brevete, după care sunt iterate unul câte unul și extrase cuvintele conținute. Se evită extragerea tuturor brevetelor odată deoarece ar fi necesare resurse substanțiale de memorie. Este de evitat și cazul în care este extras cate un brevet la fiecare pas, întrucât lucrul intens cu baza de date este mai costisitor ca timp decât operațiile executate direct în memorie.

4.4.2 Secțiunea TF-IDF

În urma execuției operațiilor din secțiunea **Preprocessing**, cu sau fără contribuția metadatei Applicant, rezultă toate datele (salvate în tabele în baza de date) necesare calculului matricei TF-IDF ce stă la baza clusterizării. Calculul matricei cu termenii TF se bazează pe matricea frecvențelor cuvintelor în documente, generată în secțiunea **Preprocessing**. La fel, termenii IDF sunt calculați pe baza numerelor de brevete în care apare fiecare cuvânt, de asemenea generate în secțiunea anterioară.

Astfel, în secțiunea TF-IDF nu mai rămâne de făcut decât execuția pe rând a funcțiilor de calcul pentru TF, IDF cu ajutorul expresiilor 3.2 și 3.3, iar mai apoi a calculului TF-IDF. În urma execuției acestor funcții rezultă matricea de ponderi TF-IDF specifică modelului spaţiului de vectori.

4.4.3 Secțiunea Clustering

În această secțiune se pot seta parametrii clusterizării k-means, se poate lansa în execuție clusterizarea propriu-zisă și se pot evalua clusterele obținute. Această secțiune conține următoarele elemente:

- **Number of clusters (k)** – Numărul de clustere ce vor fi generate în urma rulării algoritmului (numărul k de clustere).
- **Level of subclass** – daca se dorește identificarea numărului de subclase din baza de date și pe baza acestui număr să se stabilească valoarea k (numărul de clustere), atunci se poate alege un nivel de subclase și apoi se poate apăsa butonul **Compute k from subclasses**. Se poate alege unul din 5 nivele pentru subclase. Numele complet al unei clase are forma A 61K 47/00, codificarea celor 5 nivele fiind: 1 223 44/55.
- **Use subclasses** – din această secțiune se poate seta de asemenea opțiunea de a folosi clasificarea existentă a brevetelor ca punct de plecare pentru clusterizare.

- **Run clustering** – prin apăsarea acestui buton se execută algoritmul de clusterizare propriu-zis.
- **Evaluate clusters (SSE)** – după terminarea clusterizării se poate evalua rezultatul apăsând acest control. Evaluarea se face prin calcularea SSE (Sum of Squared Error) – suma erorilor la pătrat.

Algoritmul de clusterizare implementat în aplicația de clusterizare propusă în prezenta lucrare este schematizat în Figura 5.

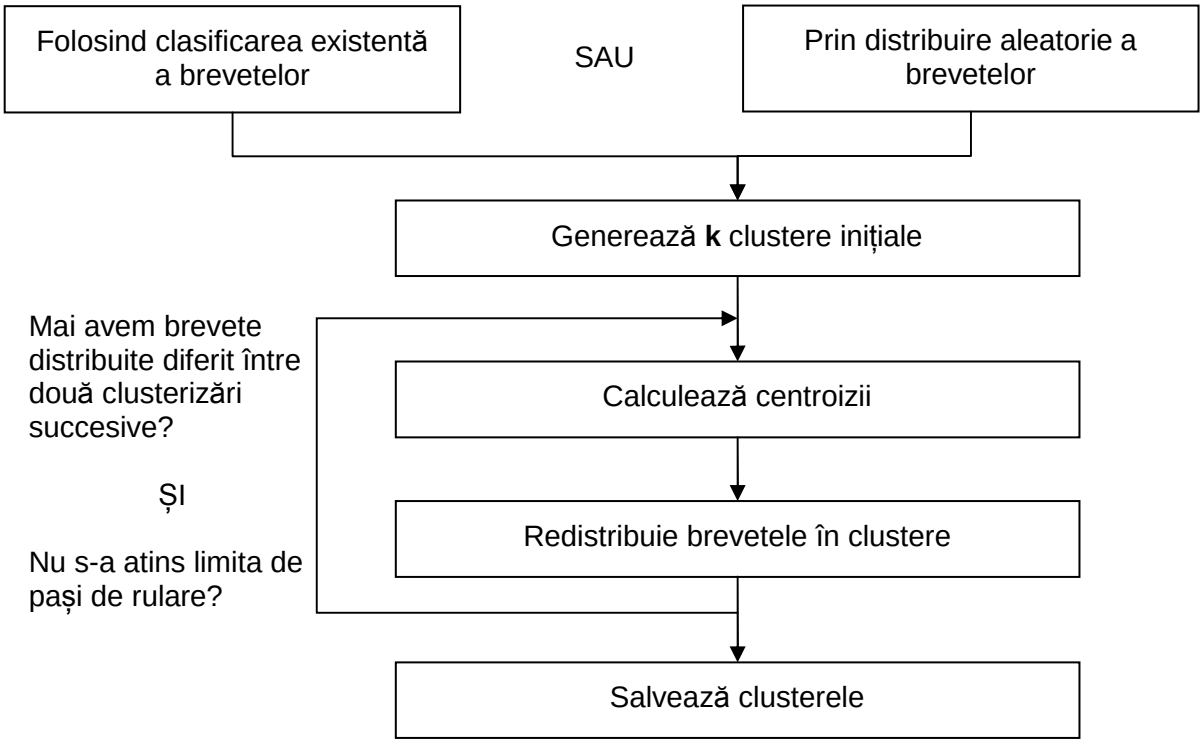

Figura 5 Pașii algoritmului de clusterizare k-means implementat în aplicația Patent Clutering

În Figura 5, la etapa de *Generează k clusterele inițiale*, în funcție de opțiunea utilizatorului, se pot genera clusterele inițiale pe de o parte în funcție de clasificarea existentă în brevete sau, pe de altă parte, prin distribuirea aleatorie a brevetelor din setul de documente ce urmează a fi clusterizat.
În etapa *Calculează centroizii*, pentru fiecare cluster în parte sunt calculați centroizii folosind expresia 3.4. După cum a fost precizat și în capitolul de contribuții teoretice 3.2.1, centroizii sunt documente generice, calculate ca medie a ponderilor brevetelor din fiecare cluster în parte.

La pasul *Redistribuie brevetele în clustere*, fiecare brevet din întreg setul de brevete este comparat pe rând cu toți centroizii calculați la pasul anterior. Comparația se face prin intermediul funcției de similaritate (funcția distanță). În cazul nostru a fost aleasă funcția cosinus (expresia 3.5). În urma comparației se alege distanța minimă între brevetul curent și centroizi, iar mai apoi se asignează brevetul la clusterul de care aparține centroidul care are distanța cea mai mică față de brevet. Pașii din această etapă au fost descriși în Figura 6.

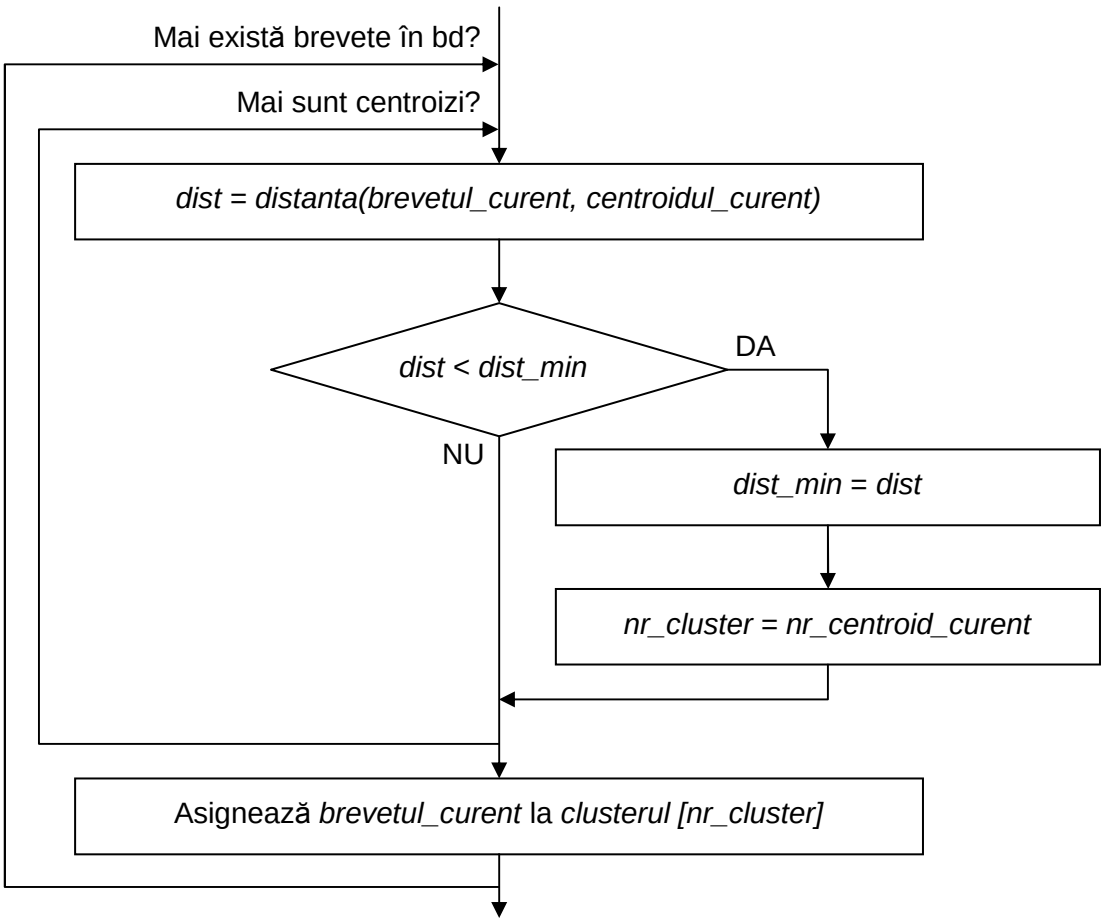

Figura 6 Detalierea pasului *Redistribuie brevetele în clustere* din algoritmul k-means

În cadrul ultimului pas din algoritmul descris în Figura 5, *Salvează clusterele*, sunt salvate în baza de date distribuțiile brevetelor pe fiecare cluster obținut în urma rulării algoritmului.

4.4.4 Secțiunea Results

Pentru testări și pentru o analiză aprofundată a rezultatelor clusterizării realizate cu ajutorul acestei aplicații, au fost dezvoltate o serie de module grupate în această secțiune. Implicit, setul de clustere generat în urma clusterizării este salvat în baza de date. Opțional însă, aplicația permite extragerea clusterelor într-un fișier CSV. Testele și analiza ce poate fi făcută în această secțiune se referă la prelucrările asupra acestor fișiere CSV cu rezultate.

- **Get patent classes** – prelucrează datele din fișierul indicat în primul câmp text
- preia clusterele din fișierul indicat în primul câmp din interfață
- pentru fiecare brevet din cluster extrage din baza de date clasa și (opțional) aplicantul
- salvează un nou fișier cu noile date extrase din baza de date
- **Evaluate clusters (SSE)** – pe baza datelor extrase din fișierul CSV indicat în primul câmp din interfață se calculează SSE (suma erorilor la pătrat)
- **Evaluate clusters (F-measure)** – dacă dispunem de un set de date de referință ce conține pre-clasificarea brevetelor (ținta), atunci clusterizarea obținută în urma procesării poate fi evaluată calculând F-measure. În primul câmp din interfață se indică fișierul CSV cu rezultatele clusterizării, iar în al doilea câmp se indică fișierul cu pre-clasificarea brevetelor (referința). Pentru evaluare se utilizează această prelucrare.

4.4.5 Secțiunea Clustering & Results

În cazul în care se dorește rularea clusterizării de mai multe ori în vederea selectării celei mai bune soluții, se poate utiliza secțiunea **Clustering & Results**. Această secțiune cuprinde parametrii necesari rulării multiple a clusterizării si alegerea în final a soluției celei mai bune.

- **Number of clustering runs** – numărul de rulări ale clusterizării
- **Number of clusters k** – numărul de clustere k
- **Run Clustering** – Prin apăsarea acestui buton se pun în execuție clusterizările propriu-zise. În ultimul câmp din interfață se va selecta fișierul CSV de referință ce conține clasificarea brevetelor și pe baza căruia se calculează indicatorul de evaluare F-measure. Pe baza SSE și F-measure calculate se poate selecta soluția cea mai bună.

4.4.6 Secțiunea Tools

Această secțiune cuprinde doar un buton ce deschide o fereastră nouă folosită la conversia fișierelor CSV exportate de RapidMiner în fișiere CSV ce pot fi importate si analizate de către prezenta aplicație.

4.4.7 Secțiunea Settings

În secțiunea **Settings** se vor putea seta parametrii de conectare la baza de date. La rularea aplicației pentru prima dată, sau în cazul în care nu se poate face accesul la baza de date, această secțiune va apărea în mod implicit. După introducerea datelor de acces la baza de date, se poate valida conexiunea apăsând butonul **Validate database connection**.

4.5 Motorul de căutare în brevete

Toate rangurile calculate cu ajutorul aplicațiilor auxiliare prezentate în secțiunile anterioare au fost integrate ulterior în motorul de căutare în brevete propus.

Motorul de căutare este compus din trei secțiuni distincte fiecare acoperind un aspect teoretic discutat anterior:

- **Advanced search** – Căutare avansată
- **Citing Documents** – Documente citate
- **Find in Clusters** – Căutarea în clustere

În continuare vor fi discutate fiecare din secțiunile disponibile în motorul de căutare.

4.5.1 Advanced search

Secțiunea ”Căutare avansată” permite căutarea unuia sau mai multor cuvinte cheie în întreaga bază de date de brevete. Prin intermediul interfeței se pot selecta diferite criterii de căutare.

Câmpurile de căutare disponibile sunt următoarele:

- **Keywords in English title or claims** – căutare de cuvinte cheie în titlul și revendicările brevetului
- **Keywords in English title** – căutare numai în titlul brevetului

- **Keywords in English claims** – căutare numai în revendicări
- **Select class** – rafinarea căutării numai într-o anumită clasă
- **Select subclass** – rafinarea căutării numai într-o anumită subclasă dintr-o clasă anterior selectată
- **From** – Căutarea brevetelor cu anul aplicării cel puțin egal cu cel selectat
- **To** – Căutarea brevetelor cu anul aplicării cel mult egal cu cel selectat
- **Applicant/Assignee** – Căutare după numele unui anumit aplicant
- **Inventor** – Căutare după numele unui anumit inventator
- **Sort by** – Afișarea rezultatelor sortate după unul dintre criteriile disponibile

Pentru această secțiune sunt disponibile patru tipuri de sortare. Fiecărui tip de sortare îi corespunde câte un rang calculat anterior, după cum a fost descris mai întâi teoretic și apoi practic în capitolele anterioare:

- **Citing index** – reprezintă sortarea descrescătoare după numărul de brevete care citează fiecare brevet rezultat în urma căutării
- **Citation weight** – sortare descrescătoare după rangul calculat cu algoritmul descris în capitolul 2.1 (Calculul rangului simplu)
- **Citation weight normalized by year** – sortare descrescătoare după rangul calculat cu algoritmul descris în capitolul 2.2.2 (Contribuții privind calculul rangului ținând cont de parametrul timp)
- **Citation weight with applicant influence** – sortare descrescătoare după rangul calculat cu algoritmul descris în capitolul 2.2.1 (Contribuții privind calculul rangului, ţinând cont de relevanţa aplicantului)

În urma rulării unei căutări, rezultă o listă de brevete ordonate după unul din tipurile de sortare ales.

În lista cu rezultatele căutării brevetelor sunt disponibile următoarele câmpuri:

- **No.** – numărul
- **IPC Class** – clasa principală a brevetului (din clasificarea IPC)
- **Title** – titlul
- **Applicants** – lista aplicanților corespunzători brevetului
- **Year** – anul aplicării brevetului
- **Citing** – numărul de brevete ce citează brevetul curent

Interfața cu lista brevetelor rezultate în urma unei căutări are câteva elemente de prezentare care facilitează accesul rapid la diverse informații suplimentare:

- Numărul brevetelor sunt generate ca linkuri către descrierea completă direct către situl EPO, unde se găsesc detaliile complete despre brevete. Cînd utilizatorii vor da click pe numele brevetului vor fi redirecționați pe situl EPO direct în pagina de descriere a brevetului.
- Când cursorul mouse-ului este poziționat deasupra numărului unui brevet apare o fereastră cu descrierea brevetului (mai exact prima revendicare din brevet)
- Cuvintele cheie căutate sunt marcate direct în text acolo unde apar.

- Formularul de căutare a fost ascuns în pagina de rezultate pentru a nu lua din spațiul destinat listei cu brevete rezultate în urma căutării. Accesul la formularul căutării se face apăsând pe butonul **Show search form**.

Toate aceste elemente de prezentare au fost adăugate pentru a îmbunătăți interacțiunea utilizatorului cu aplicația.

4.5.2 Citing documents

Secțiunea ”Citing documents” este un instrument ce permite afișarea în mod ierarhic a brevetelor ce citează un brevet selectat inițial. Acest instrument este de o deosebită importanță pentru utilizatorii care doresc ca în urma identificării unui brevet de interes să exploreze si brevete în strânsă legătura cu acesta. Plecând de la un brevet dat, se poate genera ierarhia anterioară de citări între brevete. Se poate merge astfel de la un brevet vechi către identificarea unor brevete mai noi din același domeniu cu brevetul inițial.

Ca funcționare, acest instrument permite introducerea unui brevet în câmpul text din partea de sus a paginii și apoi se apasă butonul **Citing for**. În partea de jos va apărea inițial brevetul introdus în câmpul text. Prin apăsarea acestuia vor fi listate toate brevetele care îl citează. Prin apăsare succesivă a brevetelor listate se poate merge până când considerăm necesar sau până când ajungem să avem toate brevetele expandate și nu mai există brevete care să citeze nici unul dintre brevetele expandate.

În cadrul acestui instrument există o serie de facilități care au fost implementate, ca de exemplu:

- Pentru fiecare brevet a fost generat un link direct către situl EPO unde se pot consulta toate detaliile brevetului selectat. Linkul poate fi accesat apăsând pătratul roșu al fiecărui brevet.
- Ducând cursorul mouse-ului deasupra unui pătrat albastru din interfață, se evidențiază titlul și clasa IPC corespunzătoare brevetului asociat.
- Pentru a opri temporar comportamentul de ”click pe brevet – expandează”, în scopul de a copia numărul brevetului de exemplu, se poate debifează opțiunea **click action on patents number**. Se bifează la loc opțiunea dacă se dorește revenirea la comportamentul normal.

4.5.3 Find in clusters

Ultima secțiune disponibilă în motorul de căutare propus este *Căutarea în clustere* (”Find in Clusters”). Această secțiune sintetizează în practică rezultatele teoretice descrise în capitolul 3.2. Căutările ce se pot face în această secțiune se bazează pe rezultatele clusterizărilor realizate offline cu ajutorul aplicației auxiliare **Patent Clustering** (vezi capitolul 4.4).
Pentru motorul de căutare propus, cu aplicația **Patent Clustering**, au fost rulate clusterizări după cum urmează:

- S-au extras doar brevetele aplicate pe fiecare an în parte (spre exemplificare au fost extrase pe rând doar brevetele din 1995 și 2001).
- Pentru fiecare an s-au făcut 2 clusterizări k-means: odată ținând cont și odată neținând cont de aplicanți.

- K a fost ales în funcție de numărul de subclase de nivel 2 (se poate alege unul din 5 nivele pentru subclase. Numele complet al unei clase are forma A 61K 47/00, codificarea celor 5 nivele fiind: 1 223 44/55.)
- Centroizii inițiali au fost calculați în funcție de brevetele din subclasele identificate în fiecare set de date din fiecare an.

Pentru a iniția o căutare în această secțiune va trebui mai întâi ales anul și apoi se optează pentru căutarea în clusterele în care s-a ținut sau nu cont de aplicanți. După această selecție utilizatorii vor fi redirecționați către pagina de căutare propriu-zisă.

Formularul de căutare în clustere are disponibile următoarele câmpuri:

- **Keyword in English title or claims** – căutare de cuvinte cheie în titlul si revendicările brevetului
- **Keywords in English title** – căutare numai în titlul brevetului
- **Keyword in English claims** – căutare numai în revendicări
- **Sort by** – afișarea rezultatelor sortate după unul dintre criteriile disponibile

Pentru sortare sunt disponibile aceleași 4 tipuri, ca și în cazul căutării avansate (capitolul 4.5.1).

În urma unei căutări inițiate, în această secțiune se obține o pagină cu rezultate grupate pe clustere. Rezultatele returnate sunt brevete care conțin cuvintele cheie introduse ca parametrii ai căutării și sunt grupate în clusterele din care fac parte.

În pagina de rezultate se regăsesc aceleași facilități de prezentare ca și în pagina de rezultate din secțiunea *Căutare avansată*, respectiv linkuri directe către EPO, descrierea brevetului, marcarea cuvintelor cheie etc. În plus, au fost adăugate câteva elemente noi:

- Utilizatorii au posibilitatea de a extinde listarea brevetelor dintr-un cluster. Dacă se dă click pe un cluster acesta va fi extins și vor fi afișate toate brevetele din acel cluster. Ordonarea acestora este făcută tot după tipul de sortare ales la inițierea căutării.
- Într-un cluster extins, brevetele care nu conțin cuvintele cheie introduse ca parametrii ai căutării sunt marcate diferit față de brevetele ce conțin cuvintele cheie.

Această ultimă secțiune a fost proiectată în ideea de a acoperi două nevoi importante în căutările specifice brevetelor. Pe de o parte, s-a căutat gruparea brevetelor după similaritatea conținutului acestora, astfel încât să se poată obține grupe de brevete cu același specific sau din același domeniu. Pe de altă parte, s-a încercat ca, în urma identificării unui cluster de interes, să se poată extinde căutarea listându-se toate brevetele din respectivul cluster, în vederea găsirii și a altor brevete care nu conțin neapărat cuvintele cheie introduse de utilizator la inițierea căutării.

Tot în această secțiune, clusterizarea propriu-zisă a fost abordată din două perspective. Una se referă la gruparea brevetelor bazată în mod exclusiv pe conținutul acestora, mai exact pe textul din prima revendicare. Cea de-a doua abordare a fost aceea în care gruparea / clusterizarea s-a făcut ținându-se cont și de celelalte elemente specifice brevetelor, respectiv metadatele. Dintre metadatele disponibile, aplicanții au fost interpretați si modelați astfel încât să se poată ajunge la rezultate ale căutării în urma cărora brevete ale aceluiași aplicant să fie aduse în același cluster.

5 Rezultate experimentale

5.1 Îmbunătățirea performanțelor de calcul în clusterizare

După cum a fost prezentat în capitolele 3.2.1 și 0, la baza efectuării clusterizărilor de tip text stă modelul spațiului de vectori. Acest model se bazează în principiu pe calculul unor ponderi pentru fiecare cuvânt din documentele ce aparțin setului de documente ce urmează a fi clusterizat. Calculul ponderilor pornesc de la frecvența cuvintelor în documente. Reprezentarea matematică a acestor frecvențe, și ulterior ponderi, este matricea. Numărul de coloane ale acestei matrice corespunde documentelor ce urmează a fi analizate, iar liniile reprezintă toate cuvintele din dicționarul de cuvinte extras din setul de documente.

În cazul clusterizării textului, numărul de obiecte / documente și numărul de atribute / cuvinte este semnificativ de mare. Aceste valori mari ale documentelor și respectiv cuvintelor necesită în generarea modelelor și în calculul clusterizării propriu-zise memorie de capacitate foarte mare. Pentru a acoperi necesarul de memorie generat de setul de documente ce se dorește a fi clusterizat, în aplicația auxiliară de clusterizare Patent Clustering propusă în capitolul 4.4 s-a folosit intensiv bazele de date. Astfel, în baza de date au fost stocate atât brevetele împreună cu descrierea lor, cât și modelele necesare clusterizării, respectiv matricea ponderilor din modelul spațiului de vectori.

Pentru a înțelege mai bine dimensiunea memoriei necesară pentru generarea matricei ponderilor, vom lua un exemplu:

- Dacă dorim să clusterizăm toate brevetele din baza de date aplicate în anul 2001 vom avea de clusterizat 26 181 de brevete.
- Pentru aceste brevete se generează un dicționar de 54 778 de cuvinte (inclusiv aplicanții).
- Valorile ponderilor calculate în matricea poderilor vor trebui salvate în date de tipul float, care în cele mai comune limbaje de programare, inclusiv în JAVA necesită 32 biți pentru memorare.

Așadar, pentru a genera o matrice cu aceste date, sunt necesari 26 181 coloane x 54 778 linii x (32 / 8) bytes = 5 736 571 272 bytes = 5,342 GB. Pentru prelucrarea ei succesivă este de obicei necesară menținerea a încă unei matrice rezultat, deci este nevoie de dublul acestei dimensiuni. Pe baza acestui model se generează în timpul rulării clusterizării matricea cu distanțele între brevete. Aceasta are dimensiunea 26 181 x 26 181 x 4 bytes = 2 741 779 044 bytes = 2,553 GB.

După cum se poate observa, aceste cerințe de memorie sunt costisitoare, chiar și pentru actuala tehnologie. Mai mult, în JAVA, la aplicațiile cu cerințe mari de memorie, apar pauze de lucru frecvente și consumatoare de timp, pentru a permite Garbage Collector-ului să elimine din memorie variabilele nefolosite.

Mai departe, a fost analizată structura matricei și s-a constatat că datele semnificative din cadrul ei sunt foarte rare. Astfel de matrice poartă numele de **matrice rare** (**sparse matrix**). Marea

majoritate a datelor conținute în matrice au valori nule, pentru că foarte puține cuvinte din dicționarul de cuvinte sunt folosite în descrierea unui brevet, restul valorilor având frecvența 0.

Ținând cont de acest aspect, în baza de date s-au salvat doar perechile (document – cuvânt), cu frecvența acestora, dimensiunea de stocare a acestui tip de matrice rare reducându-se considerabil. În orice moment, pe baza acestor informații salvate în baza de date se poate genera o coloană specifică matricei ponderilor. Practic din baza de date sunt extrase datele specifice unei coloane, iar când e necesar, se generează coloana completă în memorie și se prelucrează cu o altă coloană de aceeași dimensiune. În final, rezultatul este salvat din nou în baza de date.

În urma acestei abordări, aplicația JAVA auxiliară de clusterizare Patent Clustering poate rula cu o alocare a memoriei Mașinii Virtuale JAVA de 2 MB.

5.1.1 Aplicarea calculului paralel în cadrul algoritmului de clusterizare

Dintre pașii algoritmului de clusterizare implementat în aplicația de clusterizare Patent Clustering, descriși în Figura 5, cel mai costisitor, din punct de vedere al timpului computațional, s-a remarcat pasul *Redistribuie brevetele în clustere*. Acesta, după cum a fost detaliat și în Figura 6, presupune compararea între fiecare brevet din setul de brevete de clusterizat și fiecare dintre cei *k* centroizi ai clusterelor.

Complexitatea în timp al pasului *Redistribuie brevetele în clustere* este $O(nk)$, unde *n* este dat de numărul de brevete din setul de brevete de clusterizat, iar *k* reprezintă numărul de clustere.

Au fost făcute o serie de teste cu aplicația Patent Clustering, în care a fost implementat algoritmul descris în Figura 5, cu parcurgerea secvențială a pasului *Redistribuie brevetele în clustere*. Aplicația a rulat pe un calculator echipat cu un procesor Intel Core 2 Quad, ce conține 4 procesoare / 4 fire de execuție. Pentru testare s-a ales un set de 6975 de brevete ce urmau să fie clusterizate în 116 clustere. După 23 de pași, algoritmul s-a oprit, indicând un timp de procesare de 5 ore și 44 minute.

Aplicația Patent Clustering s-a dorit a fi folosită la clusterizarea unor seturi de câteva zeci de mii de brevete (30000..40000) în câte 100-150 de clustere, ceea ce ar fi însemnat un timp extrem de mare de procesare a acestora. Drept urmare, s-a încercat optimizarea aplicației în vederea obținerii de timpi cât mai rezonabili.

După identificarea secțiunii de algoritm care consumă cel mai mult timp de procesare, și anume pasul *Redistribuie brevetele în clustere* din algoritmul descris în Figura 5, s-a încercat aplicarea de calcul paralel pentru acest pas. Pentru aceasta algoritmul a fost modificat, folosind două clase care au parametri și metode care se apelează reciproc. Una dintre clase generează o serie de taskuri de executat, iar cea de-a doua reprezintă un task și are rolul de a executa operația care consumă cel mai mult timp din algoritm. În cazul nostru, această operație este compararea brevetelor cu centroizii. Taskul va beneficia de calcul paralel.

Astfel, în Algoritmul 5-1 sunt prezentați pașii prin care sunt generate taskuri ce au rolul de a calcula distanța între brevete și centroizi.

```
1   pas_curent <- 1
2   cât timp sunt brevete în setul de brevete execută
3     creează un nou task
4     transmite noului task parametrul brevet_curent
5     adaugă task-ul la lista de taskuri_pentru_execuție
6     dacă pas_curent % task_set = 0 atunci
7       așteaptă rularea tuturor taskurilor din lista de taskuri_pentru_execuție
8     sfârșit dacă
9     pas_curent <- pas_curent + 1;
10  sfârșit cat timp
```

Algoritmul 5-1 Generarea taskurilor de calcul a distanței între brevete și centroizi

Pașii rulați în cadrul unui task sunt prezentați în Algoritmul 5-2. Un task are rolul de a calcula distanța dintre un brevet și centroizi, și în final de a distribui brevetul în clusterul ce conține centroidul cu distanța minimă față de brevet.

```
1   pentru fiecare centroid execută
2     calculează distanța dintre brevet și centroid_curent
3     dacă distanța < distanța_minimă atunci
3       distanța_minimă <- distanța
4       număr_cluster <- număr_centroid_curent
5     sfârșit dacă
6   sfârșit pentru
7   asignează brevet la clusterul număr_cluster
```

Algoritmul 5-2 Pașii de rulare a unui task de calcul a distanței între un brevet și centroizi

După efectuarea modificărilor prezentate mai sus, respectiv modificarea algoritmului prezentat în Figura 6 din forma sa secvențială (liniară) în calcul paralel, și implementarea sa în JAVA, s-a retestat aplicația Patent Clustering. Condițiile de test au fost aceleași, respectiv același calculator echipat cu un procesor Intel Core 2 Quad, ce conține 4 procesoare / 4 fire de execuție și același set de 6975 de brevete ce se dorește a fi clusterizate în 116 clustere. Aplicația a rulat 25 de pași, într-un timp de o oră și 31 de minute.

Se poate observa deci că în urma modificării algoritmului și al aplicației pentru calculul paralel, s-au obținut timpi de prelucrare semnificativi mai mici. În urma testelor a rezultat o îmbunătățire a performanțelor, în care timpul de execuție s-a micșorat de aproximativ 4 ori față de cazul în care a fost folosit algoritmul secvențial de calcul.

Aplicația Patent Clustering modificată a fost folosită ulterior pentru clusterizarea tuturor brevetelor din anul 2001 în 118 clustere. Numărul total de brevete din setul de brevete de clusterizat în acest caz a fost de 26181. Pentru această clusterizare a fost utilizat un calculator cu un procesor mai puternic, Intel Xeon cu 4 nuclee, HT (în total poate rula 8 fire de execuție simultan). În acest test, după cum era de așteptat, s-au obținut timpi de execuție și mai spectaculoși. După o rulare de 29 de pași algoritmul s-a oprit, cu un timp de execuție de 3 ore și 55 minute.

5.2 Rezultate experimentale ale calculului rangului

Pentru a valida implementarea practică a elementelor teoretice expuse în prezenta lucrarea se vor lua în continuare câteva cazuri de căutări din lumea reală.

Cu ajutorul secțiunii ”Căutare avansată” se pot obține rezultate ordonate după patru criterii:

- Indexare după numărul de brevete care citează fiecare brevet în parte
- Indexare după rangul calculat cu algoritmul simplificat
- Indexare după rangul calculat ținând cont de metadata timp
- Indexare după rangul calculat ținând cont de relevanța aplicantului

Advanced Search

▼ Show Search form ▼

Found 324 patents

No.	IPC Class	Title	Applicants	Year	Citing
EP0415438	H 04B 10/16	Optical amplifier and optical communication system provided with the optical amplifier	FUJITSU LIMITED	1991-03-06	9.00000
EP0467396	H 04B 10/16	Wavelength-multiplexed optical communication system and optical amplifier used therefor	CANON KABUSHIKI KAISHA	1992-01-22	4.00000
EP0298598	H 04J 14/02	Optical communication system with a stabilized group of frequencies	AT&T Corp.	1989-01-11	6.00000
EP0329186	H 04B 10/148	Polarization diversity optical receiver for coherent optical communication	FUJITSU LIMITED	1989-08-23	5.00000
EP0040706	G 02B 06/34	Optical communication system	IBM DEUTSCHLAND GMBH International Business Machines Corporation	1981-12-02	6.00000
EP0193190	H 04J 15/00	Optical-information transmission system in the subscriber region	Standard Elektrik Lorenz Aktiengesellschaft	1986-09-03	5.00000
EP0381102	H 04J 14/02	Fibre-optical communication network with frequency division multiplexing	ALCATEL N.V.	1990-08-08	5.00000
EP0476830	H 04B 10/16	Method of operating concatenated optical amplifiers	Tyco Submarine Systems Ltd.	1992-03-25	2.00000
EP0499065	H 04B 10/20	Optical transmission system for the subscriber connection area with optical amplifiers	Alcatel SEL Aktiengesellschaft	1992-08-19	2.00000
EP0577036	H 01S 03/25	A tunable-filter control method, tunable-filter control apparatus and optical communication system using the same	CANON KABUSHIKI KAISHA	1994-01-05	2.00000
EP0488241	H 04J 14/02	Optical frequency division multiplexing network	Hitachi, Ltd.	1992-06-03	3.00000

Figura 7 Rezultatele căutării după cuvintele cheie ”optical communication” ordonate după rangul calculat cu algoritmul simplificat

Primul criteriu poate fi interpretat ca indexarea brevetelor în funcție de cât de important este el considerat de către brevetele care îl citează. Cu cât este mai citat un brevet, cu atât este considerat a fi mai important. În acest criteriu, însă nu se ține cont de CINE citează. Toate brevetele care citează au aceeași valoare. Ultimele trei indexări însă calculează pentru fiecare brevet în parte o valoare diferită, bazată tocmai pe numărul de citări.

Spre exemplu, dacă am căuta după cuvintele cheie ”*optical communication*”, lăsând celelalte opțiuni de căutare nemodificate și ordonăm după al doilea criteriu, rangul calculat cu algoritmul simplificat, obținem rezultatele din Figura 7.

Se poate observa, în acest caz, că pe prima poziție avem un brevet cu nouă citări, însă, pe a doua poziție, se găsește un brevet cu doar patru citări. Acest lucru a fost posibil datorită faptului că cele patru brevete care citează al doilea brevet din lista de rezultate sunt la rândul lor citate de un număr mare de brevete, conferindu-le astfel un rang mare care ulterior se regăsește în acest al doilea brevet din lista de rezultate. Mai mult, valoarea mare a rangului celui de-al doilea brevet dată de un număr redus de brevete citate ar putea însemna că acest brevet a stat la baza unui set ierarhic de citări de brevete, de unde se poate afirma că acest brevet este unul de referință în industria din care face parte.

În Figura 8, se pot observa rezultatele căutării pentru aceleași cuvinte cheie folosite ca și în cazul anterioar, ordonate însă după rangul calculat ținând cont de metadata timp.

Advanced Search

▼ Show Search form ▼

Found 324 patents

No.	IPC Class	Title	Applicants	Year	Citing
EP0415438	H 04B 10/16	Optical amplifier and optical communication system provided with the optical amplifier	FUJITSU LIMITED	1991-03-06	9.00000
EP0467396	H 04B 10/16	Wavelength-multiplexed optical communication system and optical amplifier used therefor	CANON KABUSHIKI KAISHA	1992-01-22	4.00000
EP0794599	H 04B 10/12	OPTICAL COMMUNICATION SYSTEM	FUJITSU LIMITED	1997-09-10	3.00000
EP0298598	H 04J 14/02	Optical communication system with a stabilized group of frequencies	AT&T Corp.	1989-01-11	6.00000
EP0577036	H 01S 03/25	A tunable-filter control method, tunable-filter control apparatus and optical communication system using the same	CANON KABUSHIKI KAISHA	1994-01-05	2.00000
EP0476830	H 04B 10/16	Method of operating concatenated optical amplifiers	Tyco Submarine Systems Ltd.	1992-03-25	2.00000
EP0488241	H 04J 14/02	Optical frequency division multiplexing network	Hitachi, Ltd.	1992-06-03	3.00000
EP0499065	H 04B 10/20	Optical transmission system for the subscriber connection area with optical amplifiers	Alcatel SEL Aktiengesellschaft	1992-08-19	2.00000
EP0329186	H 04B 10/148	Polarization diversity optical receiver for coherent optical communication	FUJITSU LIMITED	1989-08-23	5.00000

Figura 8 Rezultatele căutării după cuvintele cheie ”optical communication” ordonate după rangul calculat ținând cont de metadata timp

Spre deosebire de cazul anterior când lista rezultatelor a fost ordonată după rangul calculat cu algoritmul simplificat, în acest caz se remarcă faptul că printre primele rezultate avem mai puține brevete vechi și mai multe brevete mai noi. Prin normalizarea rangurilor pe ani s-a putut obține o distribuire mai uniformă a brevetelor, astfel încât s-a reușit atenuarea efectului de

acumulare în timp a mai multor citări de către brevetele mai vechi față de cele mai noi care nu au avut încă șansa de a fi citate.

În Figura 9, avem cel de-al patrulea tip de listare unde ordonarea este făcută după rangul calculat ținând cont de relevanța aplicantului. În acest caz, dacă comparăm cu Figura 7 unde ordonarea este realizată după rangul calculat cu algoritmul simplificat, se poate observa că acum sunt favorizate brevetele aplicate de către companiile mari cu renume la nivel mondial. Cele două companii mai puțin cunoscute care apar în Figura 7 (Standard Elektrik Lorentz Aktiengesellschaft și Tyco Submarine Systems Ltd.) se regăsesc mai jos în lista de rezultate obținute în ce-a de-a patra listare.

Se pot identifica astfel companiile relevante din diversele categorii în care se pot face căutări. Identificând aceste companii, inventatorii pot cunoaște care sunt competitorii din industria în care dorește să breveteze și ulterior să producă și să comercializeze produsele inventate.

Advanced Search

▼ Show Search form ▼

Found 324 patents

No.	IPC Class	Title	Applicants	Year	Citing
EP0415438	H 04B 10/16	Optical amplifier and optical communication system provided with the optical amplifier	FUJITSU LIMITED	1991-03-06	9.00000
EP0467396	H 04B 10/16	Wavelength-multiplexed optical communication system and optical amplifier used therefor	CANON KABUSHIKI KAISHA	1992-01-22	4.00000
EP0298598	H 04J 14/02	Optical communication system with a stabilized group of frequencies	AT&T Corp.	1989-01-11	6.00000
EP0329186	H 04B 10/148	Polarization diversity optical receiver for coherent optical communication	FUJITSU LIMITED	1989-08-23	5.00000
EP0040706	G 02B 06/34	Optical communication system	IBM DEUTSCHLAND GMBH International Business Machines Corporation	1981-12-02	6.00000
EP0577036	H 01S 03/25	A tunable-filter control method, tunable-filter control apparatus and optical communication system using the same	CANON KABUSHIKI KAISHA	1994-01-05	2.00000
EP0488241	H 04J 14/02	Optical frequency division multiplexing network	Hitachi, Ltd.	1992-06-03	3.00000
EP0794599	H 04B 10/12	OPTICAL COMMUNICATION SYSTEM	FUJITSU LIMITED	1997-09-10	3.00000
EP0381102	H 04J 14/02	Fibre-optical communication network with frequency division	ALCATEL N.V.	1990-08-08	5.00000

Figura 9 Rezultatele căutării după cuvintele cheie ”optical communication” ordonate după rangul calculat ținând cont de relevanța aplicantului

5.3 Rezultate experimentale pentru căutarea în clustere

În secțiunea ”Find in clusters” (Căutare în clustere), după cum a fost prezentată în capitolele anterioare, se pot face căutări după cuvinte cheie cu afișarea rezultatelor în grupe de brevete cu conținut similar. Inițial se afișează în fiecare cluster doar brevetele ce conțin cuvintele cheie căutate, cu posibilitatea ca ulterior clusterele de interes să poată fi expandate.

Un cluster expandat va lista toate brevetele cuprinse în el. În acest fel se poate extinde căutarea și la brevetele similare ca și conținut, dar care nu conțin în mod necesar cuvintele cheie căutate. Printr-o astfel de căutare prin extinderea clusterelor, se poate îmbunătăți lista de cuvinte cheie de căutat, sau se pot chiar găsi brevetele relevante pentru inventator.

Să luam de exemplu cuvintele cheie: ”liquid crystal”. Dacă se va face o căutare după aceste cuvinte cheie cu opțiunile de căutare: brevete din anul 1995, fără folosirea aplicantului în calculul clusterelor, se va obține un set de 220 de rezultate. Cele 220 de brevete rezultate sunt listate în 28 de clustere distincte. La o prima observație asupra conținutului clusterelor, se poate remarca un cluster cu o grupare a brevetelor ce țin de aplicații ale diverselor substanțe și materiale, iar în în alt cluster o grupare a brevetelor ce țin de elementele optice ale invenției. În același mod, se poate identifica specificul fiecărui cluster în parte și ulterior atenția poate fi concentrată pe clusterul de interes.

Implicit, brevetele în fiecare cluster sunt ordonate după numărul de citări pe care le au. Dacă inventatorul este interesat de un cluster de dimensiuni mai mari sau dacă un cluster a fost extins, este posibil ca numărul de brevete să fie mare. Se poate alege astfel una dintre metodele de sortare disponibile, pentru a lista brevetele in clustere după specificul căutării.

Rezultatele obținute din combinarea celor două tehnici de data mining, calculul rangului și clusterizarea, conduc, pe de o parte, la obținerea de rezultate mult mai relevante decât în cazul listării simple ordonate după ani a documentelor ce se potrivesc căutării direct din baza de date, și, pe de altă parte, la un proces de căutare mai simplu cu trecerea facilă prin documente relevante.

În cele ce urmează, vom analiza rezultatele căutării după cuvintele cheie ”liquid crystal” din perspectiva celor două tipuri de clusterizări disponibile, cea care se bazează pe modelul clasic și cea care ține cont de metadata aplicant. Datorită faptului că cele două clusterizări au fost făcute pe același set de date, numărul de brevete rezultate în urma căutării este același. Ce diferă este distribuția acestora în clustere.

În primul rând, cele 220 de brevete rezultate în urma căutării sunt distribuite într-un număr diferit de clustere după cum urmează: în primul caz (clusterizare bazată pe modelul clasic) sunt 28 de clustere, iar în al doilea caz (clusterizare ce ține cont de metadata aplicant) sunt 26 de clustere.

În Tabelul 1 de mai jos este prezentată în detaliu distribuția brevetelor pe clustere în fiecare din cele două cazuri de clusterizare reprezentate în paralel.

Pentru o observare mai clară a rezultatelor a fost făcută o corespondență pe fiecare linie după conținutul clusterelor. Fiecărui cluster i s-a dat un număr de identificare (coloanele *Nr. cluster*), astfel încât corespondența să fie mai ușor de urmărit în tabel. Coloanele *Nr. brevete* reprezintă numărul de brevete dintr-un cluster pe o linie, în fiecare dintre cele două cazuri de clusterizare discutate.

Tabelul 1 *Distribuția în clustere a brevetelor rezultate în urma căutării după cuvintele cheie "liquid crystal"*

	Cazul 1		**Cazul 2**	
	Nr. cluster	**Nr. brevete**	**Nr. cluster**	**Nr. brevete**
	12	1	-	-
	13	10	13	22
	16	2	16	1
	-	-	18	9
	36	8	36	10
	37	3	37	1
	38	1	38	1
	51	1	51	1
	56	38	56	35
	57	3	57	2
	58	4	58	7
	59	1	-	-
	66	2	66	2
	67	7	-	-
	68	100	68	92
	69	1	69	1
	70	1	70	1
	80	1	80	1
	82	1	82	1
	-	-	89	1
	91	8	91	8
	95	2	95	2
	101	1	-	-
	102	4	102	4
	105	6	105	4
	106	1	106	2
	109	3	109	4
	115	2	115	2
	116	2	116	2
	117	6	117	4
Numărul total de clustere	**28**		**26**	

Numărul mai mic de clustere în care sunt distribuite brevetele rezultate în urma căutării în al doilea caz, ne indică faptul că brevetele au fost redistribuite în alte clustere, apropiindu-le de brevetele ce au fost aplicate de aceleași companii, lucru urmărit de altfel în al doilea tip de clusterizare, unde, în calculul clusterelor, s-a ținut cont de metadata aplicant.

Tabelul 2 *Redistribuirea brevetelor în clustere, din clusterele generate cu k-means clasic în clusterele generate cu algoritmul ce ține cont de metadata aplicant.*

Cazul 1		**Cazul 2**	
Nr. cluster	**Nr. brevete**	**Nr. cluster**	**Nr. brevete**
13	9	13	22
16	1		
36	2		
56	3		

57	1		
68	4		
101	1		
116	1		
16	1	16	1
67	7	18	9
117	2		
36	6	36	10
37	2		
68	2		
37	1	37	1
38	1	38	1
51	1	51	1
56	35	56	35
57	2	57	2
13	1	58	7
58	4		
59	1		
68	1		
66	2	66	2
68	92	68	92
69	1	69	1
70	1	70	1
80	1	80	1
82	1	82	1
12	1	89	1
91	8	91	8
95	2	95	2
102	4	102	4
105	4	105	4
105	1	106	2
106	1		
109	3	109	4
68	1		
115	2	115	2
116	1	116	2
105	1		
117	4	117	4

În Tabelul 2, este prezentată redistribuirea brevetelor din clusterele generate cu algoritmul de clusterizare k-means bazat pe modelul clasic, în clusterele generate cu algoritmul de clusterizare ce ține cont de metadata aplicant. După cum se poate observa, multe dintre clusterele distribuite în primul caz se regăsesc în mod identic în al doilea caz. Unele clustere însă, cele generate cu algoritmul ce ține cont de metadata aplicant (coloana 3), sunt formate din brevete migrate din mai multe clustere generate cu k-means clasic. În aceste cazuri, cele mai multe dinte brevetele migrate aparțin aceluiași aplicant ce apare în clusterul destinație.

Dacă se urmărește redistribuirea brevetelor aplicate, de exemplu, de binecunoscuta companie ”Canon”, se va putea evidenția mai bine faptul că, în al doilea caz de căutare, brevetele aceleași companii se regăsesc în mai puține clustere, deci au suferit o grupare mai mare. În Tabelul 3, sunt evidențiate distribuțiile brevetelor aplicate de această companie în fiecare din cele două cazuri de căutare.

Tabelul 3 *Distribuția în clustere a brevetelor aplicate de către compania ”Canon” în urma căutării după cuvintele cheie ”liquid crystal”*

	Cazul 1		**Cazul 2**	
	Nr. cluster	**Nr. brevete**	**Nr. cluster**	**Nr. brevete**
	13	4	13	10
	16	1	-	-
	36	2	36	4
	37	3	37	1
	56	4	56	3
	66	1	66	1
	68	10	68	7
	105	1	105	1
	109	1	109	1
	116	1	-	-
	117	1	117	1
Numărul total de clustere	**11**		**9**	

Din Tabelul 3 se poate observa cu ușurință faptul că, în ce-l de-al doilea tip de căutare, brevetele aplicate de către compania ”Canon” apar într-un număr mai restrâns de clustere, față de primul caz, în care căutarea s-a făcut în clusterele calculate cu modelul clasic.

5.4 Concluzii

Prin dezvoltarea motorului de căutare descris, s-a urmărit îmbunătățirea calității căutărilor ce se pot face de către inventatori, pe de o parte, sau de către agenţii de brevetare care verifică brevetele aplicate, pe de altă parte.

Cu ajutorul uneltelor dezvoltate se poate răspunde cu succes la diferitele tipuri de căutare specifice lumii brevetelor. Se pot face astfel căutări generale, în care se urmărește identificarea unor eventuale produse asemănătoare cu cel pe care inventatorii doresc sa-l breveteze, lucru ce ar trebui făcut încă din momentul în care invenția este la nivel de idee. Se pot face căutări amănunțite, unde ținta este identificarea invențiilor celor mai apropiate ca concept de invenția ce se dorește a fi brevetată. Brevetele identificate în acest fel, pot fi mai apoi citate în brevetul aplicat. Tot astfel de căutări pot fi făcute și de către agenții de brevete, care verifică brevetele aplicate, cu scopul de a le valida originalitatea.

Nu în ultimul rând, cu ajutorul uneltelor de căutare în clustere, companiile dintr-un anumit domeniu pot face căutări sau pot genera rapoarte cu evoluția tendințelor din diferitele domenii sau industrii, prin identificarea facilă a concurenților din piața în care își desfășoară activitatea.

6 Concluzii generale, contribuţii originale şi perspective

În cercetarea realizată în această lucrare, ne-am propus să realizăm un motor de căutare specific pentru bazele de date cu brevete de invenție, pentru a rezolva probleme practice existente în una dintre cele mai importante etape din procesul de brevetare și anume căutarea „prior art”. S-a urmărit, de asemenea, afișarea rezultatelor căutării într-o formă cât mai intuitivă pentru utilizatori, cu ordonarea rezultatelor după relevanță.

Fundamentul teoretic folosit în prezenta lucrare pornește de la modelele bazate pe relevanță, în care fiecărui obiect din cadrul unei mulțimi i se calculează un rang. În funcție de valoarea acestui rang, rezultatele unei căutări în baza de date pot fi mai relevante sau mai puțin relevante.

În capitolul 1, am prezentat cele mai populare motoare de căutare și modele de relevanță folosite de acestea pentru trei tipuri de informație specifică disponibilă pe Internet: pagini web, articole științifice și brevete de invenție. Datorită multiplelor legături între cele trei tipuri de documente cercetate, în urma acestui studiu, s-a căutat îmbunătățirea soluțiilor existente de căutare cu aplicare directă în cazul particular al brevetelor de invenție.

Dată fiind asemănarea structurală a articolelor științifice și a brevetelor de invenție, în capitolul 1.3 am trecut în revistă cei mai importanți parametri bibliometrici folosiți în literatura științifică, punându-se în evidență avantajele și dezavantajele lor.

În capitolul 1.4, am descris bazele teoretice ale calculului relevanței, cu prezentarea câtorva modele și metode formale de data mining pentru gruparea și ordonarea informației de tip text după relevanță. Am prezentat algoritmul PageRank, ce va constitui și punctul de plecare în realizarea unor noi algoritmi de calcul al rangurilor brevetelor și, mai apoi, sunt descrise cele mai populare tehnici de clusterizare ale textului. În urma cercetării actualului context științific în domeniul clusterizării textului, am concluzionat că cel mai potrivit algoritm ce poate fi aplicat pe brevetele de invenție este k-means.

În urma experimentelor efectuate am putut trage concluzia că modelele și tehnicile de determinare a relevanței existente în prezent nu dau rezultate suficient de bune și a fost necesară căutarea unor metode și algoritmi de calcul noi. În urma analizei structurii brevetelor de invenție am constatat că acestea poartă informații utile ce pot fi valorificate în calculul relevanței, ca de exemplu: numele aplicantului, anul brevetării, citațiile către alte documente etc. Drept urmare în modelarea brevetelor trebuie ținut cont și de acești parametri.

Contribuțiile principale ale tezei sunt aduse în capitolele 2, 3, 4 și 5 prin propunerea unor algoritmi noi de calcul al relevanței specializați pe cazul particular reprezentat de brevetele de invenție. Acești algoritmi țin cont de parametrii suplimentari dați de structura specifică a brevetelor de invenție. Această abordare a dus la obținerea de rezultate relevante mai ales din perspectiva practică a căutării „prior art”.

În capitolul 2.1 am studiat adaptarea și aplicarea algoritmului PageRank, în forma sa de bază, pentru brevetele de invenție. Legăturile între pagini au fost înlocuite în cazul brevetelor de citările între brevete, astfel graful de legături web a fost înlocuit cu graful de citări. Din această abordare s-au putut trage următoarele concluzii:

- Dacă inițial rezultatele unei căutări puteau fi ordonate după unul din câmpurile existente în baza de date, cel mai adesea după anul apariției brevetului, acum, calculând un rang pentru fiecare brevet în parte, în urma unei căutări, brevetele pot fi ordonate după acest rang, obținându-se astfel rezultate relevante în primele poziții din lista de rezultate. Astfel, s-a îmbunătățit semnificativ calitatea unei căutări în brevete
- Datorită numărului relativ mic de brevete în comparație cu numărul de pagini web disponibile, valorile obținute cu algoritmul PageRank adaptat sunt într-o mare măsură proporționale cu valorile citărilor brevetelor. Acest lucru se datorează faptului că numărul de citații către un brevet este oarecum omogen pentru toate brevetele din baza de date
- Un alt aspect important ce se desprinde din această abordare este că în lista de rezultate relevante apar pe primele poziții brevetele vechi care au avut avantajul timpului de a fi citate. Datorită structurii arborescente a brevetelor, în funcție de apariția lor în timp, și de proprietatea brevetelor noi de a cita numai brevete existente, brevetele noi nu au încă de cine să fie citate, drept urmare nu apar în lista de rezultate relevante pe primele poziții

Pentru a compensa dezavantajele explicate în concluziile de mai sus, în capitolul 2.2 am adus o nouă contribuție prin dezvoltarea unui model mai performant ce este folosit la calculul rangului, și apoi au fost propuse două tipuri de ranguri specializate. Așadar în calculul noilor ranguri s-a ținut cont și de proprietățile specifice brevetelor de invenție. Am propus astfel calculul rangului ținând cont de relevanța aplicantului și, mai apoi, calculul rangului ținând cont de anul apariției brevetului.

În ambele abordări, s-au obținut rezultate practice mai bune. În primul caz, brevetele aplicanților cu un număr mare de brevete create apar pe primele poziții ale rezultatelor căutării, iar în cel de-al doilea caz, brevetele noi sunt distribuite omogen în lista de rezultate.

În capitolul 3, am căutat o soluție la o altă problema practică, de actualitate, și anume identificarea unor clase, domenii sau industrii noi pentru brevetele de invenție, independentă de tipurile de clasificare existente. Am adus astfel o nouă contribuție la realizarea unui model de clusterizare a brevetelor de invenție, în care se ține cont și de un set de parametri suplimentari la care se setează individual importanța pe care o au în ponderea valorilor de similitudine calculate pentru brevete.

Am testat astfel un model nou de reprezentare al brevetelor, în care s-a luat în calcul și metadata aplicant. Cu acest nou model s-au putut obține clustere în care proprietatea *numele aplicantului* influențează suplimentar valoarea similitudinii din grupul de brevete.

În urma cercetării teoretice din prezenta teză, am realizat practic o aplicație web și o serie de aplicații auxiliare (capitolul 4), în care s-au implementat algoritmii prezentați în lucrare, punându-se astfel în valoare și validând totodată teoria și modelele propuse.

Am creat astfel aplicația **Patent Data Parse**r, care este folosită pentru extragerea brevetelor din formatul inițial, prelucrarea lor și apoi salvarea în baza de date MySQL. Odată salvate în baza de date brevetele pot fi ulterior prelucrate sistematic.

O altă aplicație auxiliară dezvoltată e cea de calcul al rangurilor, cu ajutorul căreia sunt generate toate tipurile de ranguri discutate în partea teoretică.

Cea de-a treia aplicație auxiliară pe care am dezvoltat-o este aplicația **Patent Clustering**. Prin intermediul acesteia se pot realiza diferitele tipuri de clusterizări descrise în partea teoretică. Astfel, la baza acestei aplicații stă implementarea algoritmului k-means, aplicat pe diferitele modele ale brevetelor descrise teoretic în capitolul 3. Tot cu ajutorul acestei aplicații se poate evalua calitatea clusterelor obținute, folosirea măsurile SSE și F-measure.

În partea a doua a capitolului 4 am prezentat în detaliu motorul de căutare propriu zis.

Motorul de căutare este compus dintr-o serie de trei funcții de căutare, și anume:

- Căutare avansată, prin intermediul căreia utilizatorii pot introduce cuvinte cheie și obțin ca răspuns un set de brevete, ce conțin cuvintele cheie introduse, ordonate după unul dintre rangurile propuse în prezenta lucrare.
- Căutare de citări, ce permite afișarea în mod ierarhic a brevetelor ce citează un brevet selectat inițial. Acest instrument este de o deosebită importanță pentru utilizatorii care doresc să exploreze și alte brevete în strânsă legătura cu brevetul selectat.
- Căutare în clustere, unde am realizat o combinație a celor două părți ale cercetării, calculul rangurilor și clusterizările, putându-se astfel lista, în urma unei căutări, toate clusterele din care fac parte cuvintele cheie căutate, în cadrul clusterului rezultatele fiind ordonate după unul dintre tipurile de rang propuse în lucrare.

În acest ultim tip de căutare implementat în motorul de căutare propus, din lista clusterelor afișate ca rezultat, se poate selecta și extinde un cluster. În acest fel, utilizatorii sunt ajutați să identifice toate documentele relevante înrudite, astfel că, dacă un document este relevant într-un cluster, este posibil ca și alte documente din același cluster să fie relevante, chiar dacă acestea din urmă nu conțin cuvintele cheie căutate.

Una dintre provocările întâlnite în realizarea aplicației auxiliare ce realizează clusterizarea efectivă a fost timpul de procesare. După acum am arătat în capitolul 5.1, prin rafinarea succesivă a algoritmului de calcul s-a putut în final ajunge la un timp de procesare rezonabil în comparație cu volumul de date procesate. S-a folosit astfel intensiv baza de date în care au fost salvate structurile vectoriale implicate în calcule și calculul paralel ce au condus în final la folosirea optimă a memoriei și la creșterea vitezei de procesare.

Prin rezultatele experimentale prezentate în subcapitolele 5.2 și 5.3 au fost validate atât aspectele teoretice, cât și implementările practice ale acestora. Am arătat astfel că rezultatele ordonate după rangurile propuse în lucrare, obținute în urma căutărilor efectuate cu motorul de căutare propus, sunt mai relevante decât în cazul listării rezultatelor cu ordonare simplă după unul dintre câmpurile existente în brevete (de exemplu, doar după data aplicării brevetului). Pe primele poziții ale unei astfel de căutări s-au putut observa brevetele de referință din industria

din care fac parte, fiind ori foarte citate de către alte brevete, ori aplicate de către companii de prestigiu din domeniul respectiv.

Făcând o analiză a rezultatelor obținute din clusterizarea ce ține cont și de metadata aplicant, în subcapitolul 6.4, am arătat că, în acest caz, se obțin clustere care au tendința de a polariza brevetele acelorași aplicanți în aceleași clustere. Cu cât un aplicant are mai multe brevete aplicate, cu atât este mai mare polarizarea brevetelor acestora într-un cluster, aducând astfel în același grup și alte brevete relevante ale unui aceluiași aplicant.

6.1 Contribuții

În continuare sunt sintetizate contribuțiile principale aduse în cadrul tezei:

- Identificarea celor mai reprezentative modele de relevanță folosite la momentul actual în cadrul inventicii asistate de calculator.
- Adaptarea și aplicarea algoritmului PageRank la cazul particular al brevetelor de invenție
- Definirea unui nou tip de rang specializat ce ține cont de informațiile disponibile în metadatele brevetelor, de unde au fost derivate două tipuri de rang îmbunătățit:
    - Definirea rangului îmbunătățit ce ține cont de relevanța aplicantului
    - Definirea rangului îmbunătățit ce ține cont de parametrul timp
- Definirea unui model nou de reprezentare pentru documentele de tip brevetele de invenție
- Extinderea modelului spațiului de vectori prin adăugarea parametrilor suplimentari disponibili în metadate, pentru care au fost calculate ponderi cu formule de calcul specifice.
    - Adaptarea modelului spațiului de vectori extins la cazul particular al metadatei Aplicant, specific brevetelor
- Definirea și implementarea următorilor algoritmi:
    - Algoritmul de prelucrare inițială a brevetelor
    - Algoritmul de eliminare a buclelor din cadrul citărilor brevetelor
    - Algoritmul de calcul al rangului specializat ce ține cont de relevanța aplicantului
    - Algoritmul de calcul al rangului specializat ce ține cont de parametrul timp
    - Algoritmul de clusterizare k-means pentru cazul particular în care folosim modelul spațiului de vectori extins cu datele din metadata definit în teză
    - Algoritmul de clusterizare k-means adaptat la calculul paralel.
- Realizarea practică a unui motor de căutare în brevete
    - Realizarea aplicației auxiliare **Patent Data Parser**, folosită în extragerea și prelucrarea inițială a brevetelor
    - Realizarea unei aplicații auxiliare de calcul al rangurilor definite în teză
    - Realizarea aplicației auxiliare **Patent Clustering**, prin intermediul căreia se pot realiza clusterizări k-means folosind modelele definite în teză.
        - Implementarea modulului de clusterizare k-means
        - Implementarea modulului de evaluare a clusterelor folosind SSE și F-measure
    - Realizarea motorului de căutare propriu-zis
        - Implementarea funcției de căutare avansată ce returnează rezultate ordonate după tipurile de ranguri definite în teză
        - Implementarea funcției de căutare de citări

        - Implementarea funcției de căutare în clustere
- Identificarea experimentală a parametrilor folosiți în expresiile de calcul al rangurilor și clusterizărilor.
- Optimizarea implementării algoritmului de clusterizare k-means prin folosirea calculului paralel
    - Realizarea unor teste de clusterizare cu algoritmul inițial și algoritmul modificat ce folosește calculul paralel și interpretarea timpilor de execuție
- Realizarea unui experiment de căutare cu ajutorul funcției de căutare avansată din cadrul motorului de căutare implementat
- Realizarea unui experiment de căutare cu ajutorul funcției de căutare în clustere din cadrul motorului de căutare implementat

6.2 Direcții viitoare

Datorită elementelor comune ce pot fi găsite atât în structura brevetelor, cât și în articolele științifice, elementele teoretice prezentate în teză ar putea fi aplicate și în cazul acestora din urmă.

Modelele și aplicațiile prezentate sunt scalabile, putând fi aplicate pe baze de date cu un număr mare de brevete. Mai mult, eficiența lor este cu atât mai mare cu cât baza de date cuprinde mai multe brevete.

Algoritmii de calcul al rangului ar putea fi adaptați și la alte proprietăți ale brevetelor, dacă practica dovedește că ar putea fi de interes. Astfel, calculul rangului ce ține cont de relevanța aplicantului ar putea fi ușor adaptat la calculul rangului ce ține cont de numele inventatorilor.

În cadrul calculului rangului ce ține cont de parametrul timp, s-a luat în considerare normarea rangurilor pe fiecare an în parte. Pe viitor s-ar putea testa dacă nu cumva se pot obține rezultate mai relevante în cazul în care normarea rangului se face pe alte perioade de timp.

Integrarea într-o singură bază de date a mai multor baze de date de brevete este o provocare datorită faptului că, pentru diferite tipuri de brevete, există multe diferențe de structură. O altă problemă care apare în practică este folosirea de sisteme de notație diferită pentru câmpurile comune. Un simplu exemplu este tipul de clasificare, ce diferă între brevetele europene și cele nord americane.

În acest moment, în cadrul motorului de căutare se pot introduce cuvinte cheie de căutat. Foarte utilă este abordarea în care intrarea în motorul de căutare să poată fi un document întreg. În acest fel s-ar putea face o identificare a acelor documente similare cu documentul încărcat de utilizator în motorul de căutare, cu ordonare după gradul de similitudine. În foarte multe cazuri, utilizatorii caută documente similare cu un document țintă și nu o serie de documente referitoare la un anumit obiect, fenomen sau metodă.

Prezenta teză deschide calea unor cercetări ulterioare, în care s-ar putea testa și alți algoritmi de clusterizare, în afara algoritmului k-means folosit în acest moment, în eventualitatea creșterii performanțelor timpilor de calcul și a calității clusterelor obținute.

7 Listă lucrări publicate și prezentate

Mihai Vlase, Dan Munteanu, Adrian Istrate. (2012). Improvement of K-means Clustering Using Patents Metadata. LNCS. LNAI. Vol. 7376. Machine Learning and Data Mining in Pattern Recognition. 8th International Conference, MLDM 2012, Berlin, Germany. ISBN: 978-3-642-31537-4. p. 293-305

Tudorie Cornelia, **Vlase Mihai**, Nica Cristina, Munteanu Dan, Modeling fuzzy temporal criteria in database querying, 16th International Conference on System Theory, Control and Computing (ICSTCC 2012)

Mihai Vlase, Dan Munteanu. (2009). Ranking Patents for Better Search Capabilities. THE ANNALS OF „DUNAREA DE JOS" UNIVERSITY OF GALATI. FASCICLE III. Volume 32. Number 2. ISSN 1221-454X

Mihai Vlase, Dan Munteanu. (2009). Patent relevancy on patent databases. Networking in Education and Research, Proceedings of the 8th RoEduNet International Conference. Galati. Romania. ISBN: 978-606-8085-15-9

Mihai Vlase, Radu Negulewscu. (2006). Data Mining for Scientific Publications. THE ANNALS OF „DUNAREA DE JOS" UNIVERSITY OF GALATI. FASCICLE III. p.63-68, ISSN 1221-454X

Niculiță C., Istrate A., **Vlase M.,** Jâșcanu N. The Business Level Structure of WeBLE Platform. Load Tests - Proceedings of The 12th International Symposium on Modeling, Simulation and Systems' Identification – SIMSIS 12, Galați, Romania, ISBN 973-627-156-0, 2004

Brevete de invenție

Radu Negulescu, **Mihai Vlase**. (2008). (Patent). DEVICE AND METHOD FOR PROVIDING ELECTRONIC INPUT. WO200800417.
http://www.wipo.int/pctdb/en/wo.jsp?WO=2008004170

Radu Negulescu, **Mihai Vlase**. (2008). (Patent). DEVICE AND METHOD FOR PROVIDING ELECTRONIC INPUT. CA2656888.
http://brevets-patents.ic.gc.ca/opic-cipo/cpd/eng/patent/2656888/summary.html

Radu Negulescu, **Mihai Vlase**. (2009). (Patent). DEVICE AND METHOD FOR PROVIDING ELECTRONIC INPUT. US20090201248.
http://www.freepatentsonline.com/y2009/0201248.html

Radu Negulescu, **Mihai Vlase**. (2009). (Patent). DEVICE AND METHOD FOR PROVIDING ELECTRONIC INPUT. EP2082311.
http://v3.espacenet.com/publicationDetails/biblio?CC=EP&NR=2082311A1&KC=A1&FT=D&date=20090729&DB=EPODOC&locale=en_EP

Bibliografie

[1] Radu Negulescu, Mihai Vlase. (2008). (Patent). DEVICE AND METHOD FOR PROVIDING ELECTRONIC INPUT. WO200800417. http://www.wipo.int/pctdb/en/wo.jsp?WO=2008004170

[2] Radu Negulescu, Mihai Vlase. (2008). (Patent). DEVICE AND METHOD FOR PROVIDING ELECTRONIC INPUT. CA2656888. http://brevets-patents.ic.gc.ca/opic-cipo/cpd/eng/patent/2656888/summary.html

[3] Radu Negulescu, Mihai Vlase. (2009). (Patent). DEVICE AND METHOD FOR PROVIDING ELECTRONIC INPUT. US20090201248. http://appft1.uspto.gov/netacgi/nph-Parser?Sect1=PTO2&Sect2=HITOFF&u=%2Fnetahtml%2FPTO%2Fsearch-adv.html&r=1&p=1&f=G&l=50&d=PG01&S1=20090201248.PGNR.&OS=DN/20090201248&RS=DN/20090201248 or http://www.freepatentsonline.com/y2009/0201248.html

[4] Radu Negulescu, Mihai Vlase. (2009). (Patent). DEVICE AND METHOD FOR PROVIDING ELECTRONIC INPUT. EP2082311. http://v3.espacenet.com/publicationDetails/biblio?CC=EP&NR=2082311A1&KC=A1&FT=D&date=20090729&DB=EPODOC&locale=en_EP

[5] Mary Bellis. About.com Guide. http://inventors.about.com/od/definations/g/prior_art.htm. 2010

[6] Pressman D. (2005) Patent It Yourself, 11th Edition, Nolo

[7] *** WIPO – Glossary. (URL). http://www.wipo.int/pctdb/en/glossary.jsp#p. 2011

[8] *** WIPO Guide to Using PATENT INFORMATION. WIPO Publication No. L434/3(E). ISBN 978-92-805-2012-5. 2010

[9] M. Giereth, S. Brügmann, A. Stäbler, M. Rotard, T. Ertl. (2006). Application of semantic tech-nologies for representing patent metadata. Proceedings of the first international workshop on applications of semantic technologies

[10] C. Michael, H. Zan, Q. Jialun, Z. Yilu, C. Hsinchun. (2006). Building a scientific knowledge web portal: The NanoPort experience. Decision Support Systems. Volume: 42. Issue 2. p. 1216–1238

[11] *** comscore.com (UTL). http://comscore.com. 2011

[12] *** Google Corporate Information – Technology Overview. (URL). http://www.google.com/corporate/tech.html. 2010

[13] *** WordNet. (URL). http://wordnetweb.princeton.edu/perl/webwn. 2010

[14] *** Webopedia. (URL). http://www.webopedia.com/TERM/S/search_engine.html. 2010

[15] A. Abdollahzadeh Barfourosh, H. R. Motahary Nezhad, M. L. Anderson, D. Perlis. (2002). Information Retrieval on the World Wide Web and Active Logic: A Survey and Problem Definition, UM Computer Science Department. CS-TR-4291 UMIACS. UMIACS-TR-2001-69. Technical Report

[16] *** Wikipedia. (URL). http://en.wikipedia.org/wiki/Web_search_engine. 2010

[17] D. Hawking, N. Craswell, K. Griffiths. (2001). Which Search Engine is best at finding Online Services?. The Tenth International World Wide Web Conference. Hong Kong

[18] *** Wikipedia. (URL). http://en.wikipedia.org/wiki/Web_search_engine. 2010

[19] Kobayashi, M. and Takeda, K. (2000). Information retrieval on the web. ACM Computing Surveys (CSUR). ACM Press New York. Volume 32. Issue 2. p. 144–173
[20] Matthew Gray. Internet Growth and Statistics: Credits and Background. http//:www.mit.edu/people/mkgray/net/background.html
[21] Amanda Spink, Bernard J. Jansen. (2004). (Book). Web search: public searching on the Web (Information Science and Knowledge Management). Springer Science
[22] ***, Web Search Engines & Directories. (URL). http://www.webopedia.com/quick_ref/Internet_Search_Engines.asp. 2009
[23] Page Lawrence, Brin Sergey, Motwani Rajeev, Winograd Terry. (1998). The PageRank Citation Ranking: Bringing Order to the Web. Technical Report. Stanford University
[24] *** Get your own Bing experience. Sursa originală: Industry News. http://advertising.microsoft.com/asia/NewsAndEvents/News.aspx?pageid=2591&Adv_NewsID=1644. 2009
[25] *** Bing Community. Bing crawler: bingbot on the horizon. http://www.bing.com/toolbox/blogs/webmaster/archive/2010/06/28/bing-crawler-bingbot-on-the-horizon.aspx. 2010
[26] Eric J. Ray, Deborah S. Ray, Richard Selzer. (1997). (Book). The AltaVista Search Revolution (2nd ed.). Osborne/McGraw-Hill, Berkeley, California
[27] Lewis, Peter H. (1995). Digital Equipment Offers Web Browsers Its 'Super Spider'. The New York Times. Late Edition - Final, Section D, Page 4, Column 3
[28] Ricardo Baeza-Yates and Berthier Ribeiro-Neto. (1999). Modern Information Retrieval. Addison-Wesley/ACM Press, pp. 374, 390.
[29] Wendy Boswell. Web Directory - What Is a Web Directory http://websearch.about.com/od/enginesanddirectories/a/subdirectory.htm, 2010
[30] A. K. McCallum, K. Nigam, J. Rennie, and K. Seymore. (1999). Automating the construction of internet portals with machine learning. Information Retrieval Journal
[31] S. Chakrabarti, B. Dom, R. Agrawal, and P. Raghvan. (1997). Using taxonomy, discrimination, and signatures to navigate in text databases. VLDB
[32] S. Chakrabarti, B. Dom and P. Indyk. (1998). Enhanced hypertext categorization using hyperlinks. Proceedings of ACM SIGMOD
[33] S. Chakrabarti, B. Dom, R. Agrawal, and P. Raghvan. (1998). Scalable feature selection, classification and signature generation for organizing large text databases into hierarchical topic taxonomies. VLDB Journal
[34] *** Wikipedia. (URL). http://en.wikipedia.org/wiki/Yahoo!. 2010
[35] *** Search Engine Showdown. Review of Yahoo! Search. http://www.searchengineshowdown.com/features/yahoo/review.html. 2008
[36] ***] Geniac.net. ODP and Yahoo Size Charts. http://www.geniac.net/odp/. 1999
[37] *** Search Engine Showdown. Review of Ask.com. http://www.searchengineshowdown.com/features/ask/index.shtml. 2006
[38] *** Yippy Search. About Yippy Search. (URL). http://search.yippy.com/about-yippy-search. 2010
[39] *** WebCrawler. About WebCrawler. (URL). http://www.webcrawler.com/webcrawler/ws/about/_iceUrlFlag=11?_IceUrl=true. 2010
[40] *** CiteSeer. (URL). http://citeseerx.ist.psu.edu/. 2010
[41] *** Steve Lawrence , C. Lee Giles , Kurt Bollacker. (1999). Digital libraries and autonomous citation indexing. IEEE COMPUTER. Volume: 32. Issue:6. p. 67 - 71

[42] *** Soumen Chakrabarti, Martin van den Berg, Byron Dom. (1999). Focused crawling: a new approach to topic-specific Web resource discovery. Computer Networks. Volume 31. Issues 11-16. p. 1623-1640
[43] *** About CiteSeer. (URL). http://citeseerx.ist.psu.edu/about/site. 2010
[44] *** Google Scholar. (URL). http://scholar.google.com/. 2010
[45] *** About Google Scholar. (URL). http://scholar.google.com/intl/en/scholar/about.html. 2010
[46] *** Support for Scholarly Publishers. (URL). http://scholar.google.com/intl/en/scholar/about.html. 2010
[47] *** Inclusion Guidelines for Webmasters. (URL). http://scholar.google.com/intl/en/scholar/inclusion.html. 2010
[48] *** Association for Computing Machinery. (URL). http://www.acm.org. 2010
[49] Anita Cochran. (1987) ACM: the past 15 years, 1972-1987. Communications of the ACM. ACM Press, New York. Volume 30. Issue 10. p. 866-872
[50] *** ACM History. http://www.acm.org/about/history. 2010
[51] *** The Guide to Computing Literature. (URL). http://portal.acm.org/guide.cfm. 2010
[52] *** ACM Digital Library. (URL). http://portal.acm.org/dl.cfm. 2010
[53] *** The ACM Digital Library. http://librarians.acm.org/digital-library. 2010
[54] *** IEEE. (URL). http://www.ieee.org/index.html. 2010
[55] *** IEEE - History of IEEE. (URL). http://www.ieee.org/about/ieee_history.html. 2010
[56] *** IEEE. (URL). http://www.ieee.org/about/today/at_a_glance.html. Datele actuale din 31 decembrie 2009
[57] *** IEEE Xplore Digital Library. (URL). http://ieeexplore.ieee.org/. 2010
[58] *** ISI Web of Knowledge. (URL). http://wokinfo.com/. 2010
[59] *** Overview and Description. ISI Web of Knowledge. Thomson Reuters. 2008
[60] *** About Web of Knowledge. (URL). http://wokinfo.com/about/. 2010
[61] *** Digital Bibliography & Library Project. (URL). http://www.informatik.uni-trier.de/~ley/db/index.html. 2010
[62] Michael Ley and Patrick Reuther. (2006). Maintaining an Online Bibliographical Database: the Problem of Data Quality. Extraction et Gestion des Connaissances: EGC'2006. p. 5-10
[63] *** Springer. (URL). http://www.springer.com. 2010
[64] *** SpringerLink. (URL). http://www.springerlink.com/default.aspx. 2010
[65] *** About SpringerLink. (URL). http://www.springerlink.com/help/about.mpx. 2010
[66] *** United States Patent and Trademark Office. (URL). http://patft.uspto.gov/. 2010
[67] *** European Patent Office. (URL). http://ep.espacenet.com. 2010
[68] *** European Patent Office – National Offices of the Member States. (URL). http://www.espacenet.com/access/index.en.htm. 2010
[69] *** European Patent Office – Oficiul de Stat pentru Invenţii şi Mărci – Romania. (URL). http://ro.espacenet.com/. 2010
[70] *** Google Ptent Search. (URL). http://www.google.com/patents. 2010
[71] *** Free Patents Online. (URL). http://www.freepatentsonline.com, 2010
[72] *** Delphion Research intellectual property network. (URL). www.delphion.com/simple. 2010
[73] *** MicroPatent. (URL). http://www.micropatent.com/static/index.htm. 2010

[74] *** LexPat. (URL). www.lexis-nexis.com, 2010
[75] *** QPAT. (URL). www.qpat.com. 2010
[76] *** PatentMax. (URL). www.patentmax.com. 2010
[77] *** PatBase. (URL). www.patbase.com. 2010
[78] Ronen Feldman, Ido Dagan, Haym Hirsh. (1998). Mining Text Using Keyword Distributions. Journal of Intelligent Information Systems, Springer. Volume 10. Number 3. p. 281-300
[79] Michael Eisenberg, Carol Barry. (1988). Order effects: A study of the possible influence of presentation order on user judgments of document relevance. Journal of the American Society for Information Science. Volume 39. Issue 5. pages 293–300
[80] Presley R. L., Caraway, B. L. (1999). An Interview with Eugene Garfield. Serials Review, Volume 25. Issue 3. p. 67-80
[81] Kathleen Bauer, Nisa Bakkalbasi. (2005). An Examination of Citation Counts in a New Scholarly Communication Environment. D-Lib Magazine. Volume 11. Number 9
[82] Mihai Vlase, Radu Negulewscu. (2006). Data Mining for Scientific Publications. THE ANNALS OF „DUNAREA DE JOS" UNIVERSITY OF GALATI. FASCICLE III. p.63-68, ISSN 1221-454X
[83] Aparna Basu, Ritu Aggarwal. (2001). Indian Scientific Literature in Science Citation Index: A Report. Information Today & Tomorrow. Volume 20. Number 4. p.3-8, p.17, p.22
[84] Eugen Garfield. (Editor). (1980). Science Citation Index, Journal Citation Reports: A bibliometric analysis of science journals in the ISI database. Science Citation Index 1979 annual. Volume 14. Philadelphia: Institute for Scientific Information (ISI)
[85] Eugen Garfield. (1972). Citation Analysis as a Tool in Journal Evaluation. Science. Volume 178, p. 471- 479
[86] Citation Indexing, Its Theory and Application in Science, Technology, and Humanities. (1979). (Book). John Wiley & Sons, New-York
[87] Isidro F. Aguillo, Begoña Granadino, José L. Ortega, José A. Prieto. (2006). Scientific research activity and communication measured with cybermetrics indicators . Journal of the American Society for Information Science and Technology. Volume 57. Issue 10. p. 1296–1302
[88] Eugene Garfield (1994). Expected Citation Rates, Half-Life, And Impact Ratios: Comparing Apples To Apples In Evaluation Research. Current Contents.
[89] Tibor Braun, Wolfgang Glänzel, András Schubert. (1985). (Book). Scientometric indicators: a 32 country comparative evaluation of publishing performance and citation impact. World Scientific
[90] Alexandru T. Balaban, Eustratios N. Carabateas, Florin T. Tanasescu. (1998). (Book). Science and technology management. Series 4: Science and Technology Policy. IOS Press . Volume 22
[91] Plomp R. (1994). The highly cited papers of professors as an indicator of a research group's scientific performance. Scientometrics. Volume 29. Number 3. p. 377-393
[92] Ioan-Iovitz Popescu. (1994). Science Journal Ranking by Average Impact Factors. http://alpha2.infim.ro/~ltpd/Jo_rankingb.htm
[93] Romanian Ministry of National Education, Order No. 5103, Appendage 1-II, dated on 05.07.1999
[94] Fallows D. (2005). Search engine users. Washington D.C.: Pew Internet & American Life Project

[95] Phil Craven. (2002). Google's PageRank explained: and how to make the most of it. Technical report. WebWorkshop Document. http://webworkshop.net/pagerank.html
[96] Neeraja Sankaran. (1995). Speculation In The Biomedical Community Abounds Over Likely Candidates for Nobel. The Scientist. Volume 9. Issue 19. p. 1
[97] Sergey Brin and Lawrence Page. (1998). The Anatomy of a Large-Scale Hypertextual Web Search Engine. Computer Networks and ISDN Systems. Volume 30. Issues 1-7. p. 107-117
[98] Junghoo Cho, Sourashis Roy, Robert E. Adams. (2005). Page quality: in search of an unbiased web ranking. International Conference on Management of Data: Proceedings of the 2005 ACM SIGMOD international conference on Management of data. Baltimore. Maryland. p. 551 - 562
[99] Pang-Ning Tan, Michael Steinbach, Vipin Kumar. (2006). (Book). Introduction to Data Mining. Addison-Wesley
[100] Christopher D. Manning, Prabhakar Raghavan, Hinrich Schütze. (2009). (Book). An Introduction to Information Retrieval. Cambridge University Press. Cambridge, England
[101] Michael Steinbach, George Karypis, Vipin Kumar. (2000). A Comparison of Document Clustering Techniques. KDD Workshop on Text Mining
[102] Per O Seglen. (1997). Why the impact factor of journals should not be used for evaluating research. British medical journal. Volume 314(7079). p. 498–502
[103] Mihai Vlase, Dan Munteanu. (2009). Ranking Patents for Better Search Capabilities. THE ANNALS OF „DUNAREA DE JOS" UNIVERSITY OF GALATI. FASCICLE III. Volume 32. Number 2. ISSN 1221-454X
[104] Mihai Vlase, Dan Munteanu. (2009). Patent relevancy on patent databases. Networking in Education and Research, Proceedings of the 8th RoEduNet International Conference. Galati. Romania. ISBN: 978-606-8085-15-9
[105] *** WIPO - International Patent Classification. (URL). http://www.wipo.int/classifications/ipc/en/. 2011
[106] Péter Érdi, Kinga Makovi, Zoltán Somogyvári, Katherine Strandburg, Jan Tobochnik, Péter Volf, László Zalányi. (2013). Prediction of emerging technologies based on analysis of the US patent citation network. Scientometrics. Volume 95. Issue 1. p. 225-242
[107] Text Clustering. (URL). http://www.delphion.com/products/research/products-cluster
[108] S. Manish. (2009). Text Clustering on Patents. White Paper. Gridlogics Tech. Pvt Ltd.
[109] PatentCluster. (URL). http://www.patentcluster.com/. 2011
[110] Huang, Anna. (2008) "Similarity measures for text document clustering." Proceedings of the sixth new zealand computer science research student conference (NZCSRSC2008). Christchurch, New Zealand
[111] A. Strehl, J. Ghosh, and R. Mooney. (iulie 2000) Impact ofsimilarity measures on web-page clustering. In AAAI-2000: Workshop on Artificial Intelligence forWeb Search
[112] Mihai Vlase, Dan Munteanu, Adrian Istrate. (2012). Improvement of K-means Clustering Using Patents Metadata. LNCS. LNAI. Vol. 7376. Machine Learning and Data Mining in Pattern Recognition. 8th International Conference, MLDM 2012, Berlin, Germany. ISBN: 978-3-642-31537-4. p. 293-305
[113] Marc Krier, Francesco Zaccà. (2002). Automatic categorization applications at the European patent office. World Patent Information. Elsevier. Volume 24, Issue 3. p. 187-196