ISSN 1330-3651 (Print), ISSN 1848-6339 (Online)
DOI: 10.17559/TV-20131118180118

PSALM – PATENT MINING TOOL FOR COMPETITIVE INTELLIGENCE

Željko Tekić, Miroslava Dražić, Dragan Kukolj, Ljubiša Nikolić, Sandra Kukolj, Milana Vitas

Original scientific paper
Patent document is a valuable source of information. However, it is neither easy to extract useful information from patents nor simple to track evidence about all patents that may be relevant. This paper describes PSALM (Patent Search and Analysis for Landscaping and Management), a recently developed software tool for competitive intelligence based on patent data. PSALM enables transformation of raw patent data into meaningful and useful information for business decision making. The tool is based on MySQL database and web robot, both supported by routines developed in Java and PHP. PSALM tool assembles patent data from publicly available data bases, collects and analyses bibliographic parameters of patents, but also does text mining and clustering. The objective of this paper is to describe the structure and functions of developed software, to show efficiency and accuracy of its modules (text processing, clustering, visualisation), as well as to demonstrate its usability through an in-depth case study.

Keywords: *clustering; competitive intelligence; decision-making; patents; PSALM; software; text mining; visualisation*

PSALM – alat za analizu konkurenata baziran na podacima iz patenta

Izvorni znanstveni članak
Patentni dokument je vrijedan izvor informacija. Međutim, nije lako izvući korisne informacije iz njega niti je jednostavno pratiti evidenciju o svim patentima koji mogu biti relevantni. Ovaj rad opisuje strukturu, module, performanse i funkcionalnost PSALM-a, nedavno razvijenog softverskog alata za analizu konkurenata temeljenog na patentnim podacima. PSALM omogućuje preobrazbu sirovih patentnih podataka u smislene i korisne informacije za donošenje poslovnih odluka. Alat se temelji na MySQL bazi podataka i web robotu, oba su podržana potprogramima razvijenima u Java i PHP-u. Patent Search and Analysis for Landscaping and Management (PSALM) alat sakuplja patentne podatke iz javno dostupnih baza podataka, prikuplja i analizira bibliografske parametre patenata, ali i radi dubinsku analizu teksta. Cilj ovog rada je opisati strukturu i funkcije razvijenog softvera, pokazati učinkovitost i točnost njegovih modula (za procesuiranje teksta, grupiranje, vizualizaciju), ali i pokazati njegovu upotrebljivosti kroz dubinski studij slučaja.

Ključne riječi: *donošenje odluka; grupiranje; konkurenti; patenti; PSALM; rudarenje teksta; softver; vizualizacija*

1 Introduction

Patent is a complex legal instrument and a powerful business tool. Based on patents, companies can gain monopoly position in the market, block and disadvantage competitors, attract investors or make additional profits through licensing. At the same time, patents are a unique and valuable source of information. WIPO and EPO estimate that approximately 80 % of the scientific and technical information disclosed in patents is never published in any other form [1, 2]. In addition to technical data, patent document provides a lot of information relevant for legal, business and public policy usage (Tab. 1 gives summary of the format and information contents of patent documents). The validity of this information is amplified by the fact that all data found in a patent document is collected, verified and presented in a systematic manner according to internationally agreed standards. Based on this, patents offer a full spectrum of possibilities for using them in key areas of competitive intelligence and technology management, including [3]: competitors monitoring, technology trends observation, the identification and assessment of potential partners and R&D portfolio management. Researchers and inventors, R&D managers, patent professionals, entrepreneurs and policy-makers are interested in using this information in strategic decision making, as well as in everyday operations. However, it is neither easy nor simple. There were 9,45 million patents in force in 2013 [4] with an increasing number of pages and claims per patent, difficult language used and unclear relations between patents. To overcome the barriers, various software tools have been developed in the patent field [5, 6]. They could analyse patent portfolios, make basic statistics, visualize, map and landscape the patent data. Most of these tools use statistical methods to analyse patent data and represent patent trends by various graphs and tables. They provide various features and representations for researchers, managers and R&D specialists. However, most of the patent databases and tools available today are expensive, complex or ask for a strong expertise in the field of intellectual property. Therefore, SMEs and academic institutions, especially in developing countries, do not take a full advantage of using patents as a source of information for their own research, market and innovative activities [2]. Responding to this challenge, our research group has developed a tool for patent data analysis and management.

The software tool is named Patent Search and Analysis for Landscaping and Management (PSALM). It allows fast access to free patent databases and provides an easy way to automatically analyse patents. The PSALM is designed to collect and analyse both structured (bibliographic parameters) and unstructured (free text) patent data and to visualize the results of both analyses.

The objective of this paper is to describe the structure and functions of developed software, to show its clustering efficiency and accuracy, as well as to demonstrate its usability through case study. In several conference papers [7, 8, 9, 10] different features of the tool were presented and tested. However, this is first holistic presentation of PSALM's structure, functions and an in-depth application in real life case study.

The rest of the paper is organized as follows. Section 2 describes structure and functional modules of PSALM, while Section 3 assesses their performances. In Section 4, user interface is presented. Implications for practice are discussed in Section 5. Finally, in Section 6, a conclusion with a summary of results and further research are outlined.

Table 1 Format and information contents of patent documents

Format	Where can be found?	Contents	Type	Use
Title and abstract	Front page	Concise summary of the technology of the invention.	Unstructured – free text	Archiving
Bibliographic data	Front page	Bibliographic information, i.e. the document number, filing and publication dates, name of the inventors, assignees, etc.	Structured – strict format	Business and public policy
Description	Main body	Discloses clearly the technical details, illustrated by working examples showing how to carry out the invention into practice.	Unstructured – free text	Technical and legal
Claims	Main body	Define the scope of protection for the invention under consideration, hence satisfying the legal aspect of the patent document.	Unstructured – free text	Legal

2 Software structure and modules

The PSALM is a software tool developed to analyse patent portfolios with larger number of patents. It is designed to search for patents based on given information, to automatically download them, analyse their structured and unstructured data, and to visualize the results. The most important parts of the tool are the functional modules for patents collection, text processing, clustering and visualization (Fig. 1). These modules will be further described in more details. In addition to these complex modules, there are modules for analysing IPC codes, extracting and displaying citing and cited patents, progress report module and module for recording data in the CSV file. All modules are developed in programming languages Java and PHP, while the database is developed in MySQL.

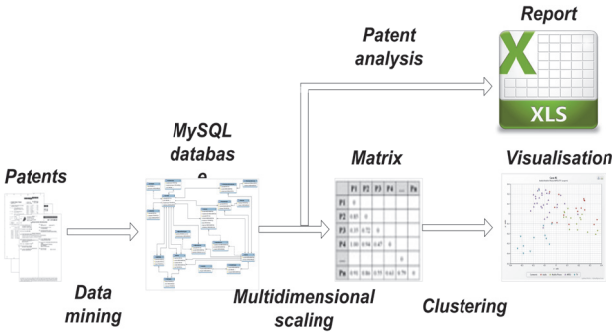

Figure 1 Structure and functional modules of the PSALM tool

2.1 Web Robot

The front end of the software is a so-called "web robot". It collects data on patents from trustful public database, namely USPTO (US patent office) or Espacenet (European Patent Office) database and parses them. Web robot [7] extracts patents' bibliographic parameters (structured part of patent data), like: title, inventor(s), applicant, priority date, country of publication, priority number, priority country, references cited by the patent, patents citing the patent, IPC classes, etc., but also translates free text to structured data form. Bibliographic data are collected to allow the development of various statistics, correlations and histograms. On the other hand, unstructured text is processed to enable identification and extraction of undetected or unexpected patterns of concentrations embedded in a set of patents and visualization with respect to various contexts, such as technological similarities or the scope of claims. All collected information is archived in a local MySQL database for further analysis. The database has the following tables (Fig. 2): mainData, searchSetup, caseData, citationData, claims, ipc, and apa. All tables are InnoDB type, which allows us to have the true relational database structure. These relationships are of the one-to-many type in which one instance of the parent entity relates to many instances of the child entity. In our case each row of mainData table contains a unique key which connects it with one or more rows in child tables: caseData, citationData, claims, ipc and apa.

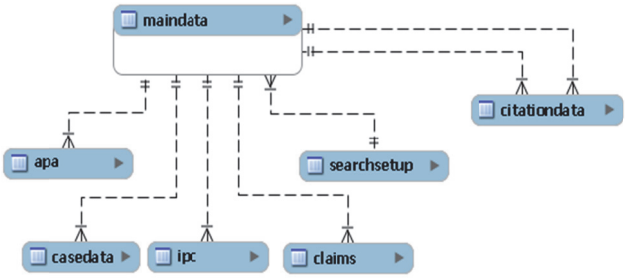

Figure 2 Enhanced entity relationship model of MySQL database

2.2 Text Processing

The second functional module of the tool is text processing module. It is based on Term Frequency - Inverse Document Frequency (*TF-IDF*) weighting scheme [11] for keyword extraction and on Multi-Dimensional Scaling (*MDS*) scheme [12] for data dimensionality reduction. This module performs text mining and text analysis. Its main goal is to extract important attributes and keywords from a patent data structure and to analyse different parts of patent text (abstract, description, claims or other data).

TF-IDF is a statistical method for determining the importance of words within a document, which belongs to a larger set of documents [11]. For calculating the *TF-IDF* weight of a term (t) in a particular document (d), it is necessary to know how often it occurs in the document. This is named as the "term frequency" (TF), while the frequency of appearing in documents (d) of the collection (D) is known as the "document frequency" (*DF*). If we consider what is inverse from the document frequency (*IDF*), we can calculate the weight by multiplying *TF* by *IDF*. In practice, the inverse document frequency is calculated as the logarithm of the proportion of the total

number of documents (N) and the document frequency in order to scale the values. The calculation of *TF-IDF* is shown in Eqs. (1), (2) and (3).

$$TF(t,d) = 0{,}5 + \frac{0{,}5 * f(t,d)}{\max\{f(w,d) : w \in d\}} \quad (1)$$

$$IDF(t,D) = \log \frac{|D|}{|\{d \in D : t \in d\}|} \quad (2)$$

$$TF - IDF(t,d,D) = TF(t,d) * IDF(t,D) \quad (3)$$

The importance of words within each patent document, derived by the *TF-IDF* method, is used for creation of dissimilarity matrix. The dissimilarity matrix given by (4) is square distance matrix where $\delta_{i,j}$ is distance function that measures level of dissimilarity between pair of patent documents calculated using cosine of vectors assigned to the each patent document, and term I represents number of patent documents in the considered case.

$$\Delta := \begin{pmatrix} \delta_{1,1} & \cdots & \delta_{1,I} \\ \vdots & \ddots & \vdots \\ \delta_{I,1} & \cdots & \delta_{I,I} \end{pmatrix} \quad (4)$$

This high dimensional matrix is transformed into much lower dimensionality space, maintaining the most similar structure to the original, using the Multi-Dimensional Scaling (*MDS*) scheme [12]. The goal of *MDS* is to find vectors $x_1, \ldots, x_I$ such that:

$$x_1, \ldots, x_I \in R^N \quad (5)$$

$$\|x_i - x_j\| \approx \delta_{i,j}; \quad i,j \in I \quad (6)$$

The output of the *MDS* module is a 2-dimensional matrix that can be easily used to visually present patents similarities.

2.3 Clustering

The output of the text processing module is a 2-dimensional matrix that is further processed by the clustering module. Clustering is partitioning a set $O = \{O_1, O_2, \ldots, O_n\}$ of objects into homogenous clusters maximizing intra-cluster similarity while minimizing inter-cluster similarity. Clusters are formed without any prior information about objects that are grouped. Any labels associated with objects are obtained solely from the data.

Clustering helps identifying meaningful patterns, undetected or unexpected groups from a set of unlabelled objects [13]. Unsupervised clustering technique groups the given unlabelled collection of patent documents into meaningful clusters without any prior information of patent documents. As the number of patents increases and volume of data grows, it is impossible to successfully analyse any set of patents without clustering. Therefore, clustering is the essential function any patent analysis tool should provide.

Due to importance of four unsupervised learning algorithms: *k*-means, fuzzy *c*-means, neural gas and re-organizing neural network (ronn) that will be analysed, their main features are shortly explained in the following text and later tested in an experimental setting.

2.3.1 *k*-means clustering

The *k*-partitioning attempts to detect k optimal clusters by an iterative relocation method based on an optimization function. The most popular *k*-clustering is *k*-means clustering [14]. *k*-means clustering segments the n observations into k clusters, where each observation belongs to the cluster with the nearest mean. It uses the mean (centroid) as the representative of a cluster. One of main strengths of *k*-means clustering is its scalability. *k*-means clustering problem regards to the complexity of NP-hard. However, there are efficient approximate solutions.

2.3.2 Fuzzy *c*-means clustering (fcm)

Fuzzy *c*-means clustering is very similar to *k*-means. The difference is that each object has a degree of belonging to clusters, rather than belonging to just one cluster. Thus, objects on the edges of the cluster are in the cluster in somewhat lesser degree than objects inside the cluster. Fuzzy *c*-means is an important algorithm for image processing used for clustering of objects inside the image [15]. Complete overview and comparison of fuzzy clustering is presented in [16].

2.3.3 Re-Organising Neural Network (ronn)

Re-Organising Neural Network (ronn) algorithm is an iterative learning procedure. It performs iterative adjustments of node-coordinates in the manner of k-means algorithm until the nodes stabilize relatively in their current positions. Simultaneously it shifts nodes that turn out to be dead-nodes into better positions. Complete overview of ronn algorithm is presented in [17].

2.3.4 Neural Gas

The Neural Gas is an unsupervised learning technique that allows the uniform placement of representative prototypes in the vector space. This algorithm determines prototypes in such a way that the Euclidean distance between data vectors and prototype vectors is minimal. The neural gas is a relatively simple algorithm for finding optimal data representations and is a robust alternative for k-means clustering. It is widely used where compression is a problem, like image processing or pattern recognition [18]. Algorithm's detailed description is given in [19].

2.4 Visualization

This module is responsible for data visualization or/and exporting the results. Patent data is processed with data and text mining techniques in order to present the data in some visual form which will allow its better understanding and interaction with the data [20]. The

clustered patent data space can be presented and visualized with respect to various contexts.

The tool enables visualizations of high and low-dimensional data. High-dimensional data are visualized by mapping patents and clusters in proportion to each other in 2D space. This makes it very easy to locate the most developed areas in certain technologies. It also shows outliers in the data from patents that do not have much in common with the subject. Low-dimensional (structured) data, presented as bar charts and pie charts of bibliographic data, could also help in better understanding of the technology areas, changes in the technology development, company competiveness etc.

3 PSALM performance assessment

3.1 Web robot performance assessment

In the first phase the web robot performances were assessed using several patents with different data available on "front page". The following patents were tested: US7962846 (patent with standard front page data), US7919816 (patent with additional field: Foreign Application Priority Data), US7962825 (patent with additional fields: Related U.S. Patent Documents and Parent Case Text), D503691 (US design patent with nonstandard data), and D254200 (US design patent with standard data).

Program execution and time performances of acquiring, parsing and writing data into the database were analysed, as well as statistical data on the amount of data that is written to the database. The speed of the Internet connection is an important factor in assessing program performances because it affects the most speed of program operation. In the test case download speed was 6 Mbps and upload speed was 0,36 Mbps. The amount of data that should be processed and the time needed for processing are shown in Tab. 2 and Tab. 3.

Table 2 Amount of data that is written to the database in the test case

	Database tables (as in Fig. 2)					
PID	Search Setup	Main Data	Case Data	Citation Data	Claims	IPC
7962846	1	17	1	16	516	64
7919816	1	7	1	6	140	26
7962825	1	11	1	10	182	39
D503691	1	18	1	17	317	76
D254200	1	15	1	17	7	16

Table 3 Processing time in the test case

PID	Total processing time	Number of processed patents (Main Data)	Time per patent
7962846	23 s 827 ms	17	1,40 s
7919816	12 s 359 ms	7	1,76 s
7962825	21 s 671 ms	11	1,97 s
D503691	23 s 235 ms	18	1,29 s
D254200	16 s 155 ms	15	1,07 s

In the second phase the web robot performances were assessed using portfolio of 1820 selected US patents. Statistical data on downloading the patent portfolio is presented in Tab. 4 while the average download time for the same portfolio is shown in Tab. 5.

The average time is calculated using unique patents only, i.e. patents that were actually downloaded and written into the database. Patents that already existed in the database (duplicates) were not downloaded but just linked with the current search.

Table 4 Statistical data on processing patent portfolio with 1820 patents

Number of patents in the patent portfolio (A)	Number of all citations (B)	Number of unique patents which are cited (C)	Average number of citations (B/A)	Average number of unique patent citations (C/A)
1820	55754	29121	30,63	16,00

Table 5 Processing times for patent portfolio with 1820 patents

Total processing time	Total number of patents	Total number of unique patents	Average time per unique patent
17 h 15 m 42 s 567 ms	59829	30225	2,06 s

3.2 Clustering performance assessment

To test clustering accuracy dataset with 72 US patents is used. All patents have been invented by the same company and cover the field of consumer electronics. Five experienced engineers clustered these patents in the following four groups: Audio, MPEG, Mobile phone and TV. The Audio group consists of 16 patents related to audio coding, audio signal transmission, audio processing, wide-band audio coding and techniques related to audio editing and trick play features. These inventions could be implemented in audio home entertainment equipment, mobile devices and portable gadgets like MP3 players. The MPEG group consists of 29 patents which relate to various optimizations techniques for hardware and software video decoding, creating multi-streams of compressed video data, increasing image compression efficiency, improving error concealment, extracting coding parameters and quality improvement of scalable coding techniques. The Mobile Phone group consists of 15 patents related to call re-establishment and call transfer in telecommunication networks, software defined radio, signal filtering and equalization. The last group is the TV group that consists of 12 patents related to image sharpness enhancement.

In order to test accuracy of clustering algorithms and select the best performing for patent data, four described (*k*-means, the neural-gas, fuzzy-c-means and ronn) clustering algorithms are compared. Although artificial intelligence and machine learning have significantly improved over the last few years, patent analysis and clustering by human experts has remained the safest and the most accurate method. Therefore, the results of clustering techniques have been compared to expert's results as well. In order to do that, the following methodology was adopted:

- for the selected dataset the text processing is performed on different subsets of patent data: abstract, claims, patent description and IPC codes;
- after text processing is finished, clustering is performed (four different functions in Matlab have been developed, one for each clustering algorithm);

- after that, visualization part of the algorithm is performed in order to show the patents in 2D space;
- the graphical user interface is used to mark classified patents to defined 4 clusters;
- the graphical user interface is used to export the coordinates of patents and corresponding groups in CSV table, and
- the obtained table is used to evaluate and calculate cluster accuracy.

For evaluation of obtained clusters in 2D space, the Davies-Bouldin (*DB*) Index has been used [21]. It is a measure which indicates the similarity of clusters which are assumed to have a data density as a decreasing function of distance from a vector characteristic of the cluster. The measure can be used to infer the appropriateness of data partitions and can therefore be used to compare relative appropriateness of various divisions of the data.

If C_i is a cluster of vectors, X_j is a n-dimensional vector assigned to cluster C_i, A_i is the centroid of C_i and T_i is the size of the cluster i, then S_i is a measure of scatter within the cluster and it is calculated as follows:

$$Si = \sqrt[q]{\frac{1}{Ti}\sum_{j=1}^{Ti}\left|X_j - A_i\right|^q} \tag{7}$$

If q is equal to 2, which is usually the case, this is a Euclidean distance function between the centroid of the cluster and the individual vectors. Of course, other distance metric can be used instead.

M_{ij} is a measure of separation between cluster C_i and C_j and it is defined as follows:

$$M_{ij} = \left\|A_i - A_j\right\|p = \sqrt[p]{\sum_{k=1}^{n}\left|a_{k,i} - a_{k,j}\right|^p} \tag{8}$$

where k_i is the k^{th} element of A_i and there are n such elements in A. Let $R_{i,j}$ be a measure of how good the clustering scheme is. Then $R_{i,j}$ is calculated as:

$$R_{i,j} = \frac{S_i + S_j}{M_{i,j}} \tag{9}$$

The lower the value $R_{i,j}$ is, the better the separation of clusters and the 'tightness' inside the clusters. If D_i chooses the worst case scenario, for example:

$$D_i = \max_{j:i\neq j} R_{i,j} \tag{10}$$

then, if N is the number of the clusters, Davies Bouldin index can be defined as in (11). In this way defined, Davies Bouldin index is a non-negative value.

$$DB = \frac{1}{N}\sum_{i=1}^{N} D_i \tag{11}$$

The classification accuracy of the four selected algorithms: *k*-means, the neural-gas, fuzzy c-means and ronn, for each of the four patent subparts: abstract, claims, descriptions and IPC is calculated by Matlab. Tab. 6 illustrates the results, the values of Davies–Bouldin (DB) indices. The results demonstrate similar performances for all four tested clustering techniques. However, fuzzy c-means and ronn algorithms give slightly better results, while *k*-means algorithm shows less accurate results than other algorithms.

At the same time, the best results were achieved by processing description part of the patent documents, while processing only IPC codes gives slightly poorer results. Based on these results fazzy *c*-means algorithm was selected and implemented in PSALM. Also, the tool enables processing either description or patent full text.

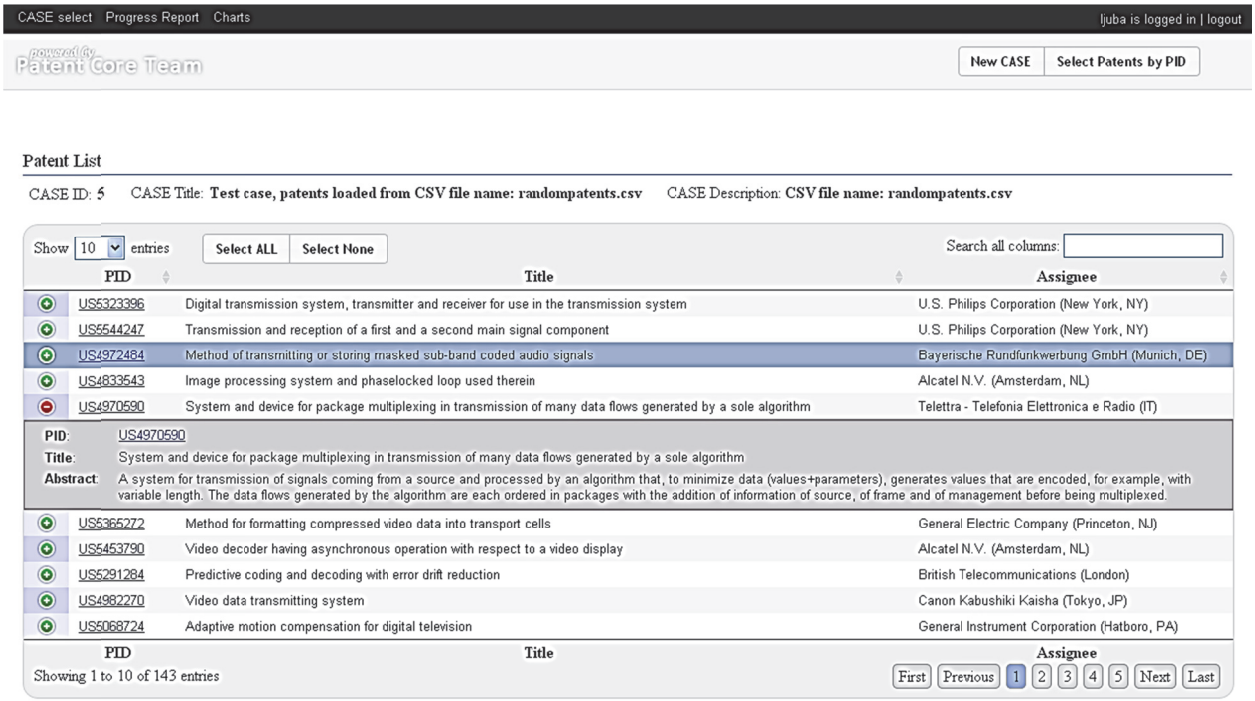

Figure 3 An example of PSALM interface – List of patents in the case

Table 6 DB Index

	DB index			
	k-means	Neural-gas	fcm	ronn
description	0,79	0,79	0,85	0,82
claims	0,70	0,74	0,76	0,76
abstract	0,73	0,75	0,75	0,75
IPC	0,66	0,66	0,70	0,64

4 User interface

The PSALM is case-based tool developed to analyse a larger number of patents and to serve multiple networked users at the same time in server-client manner. Each case is made of a group of patents selected on the basis of users' defined criteria. Criteria for creating a new case can be based on: assignee, IPC codes and cited and citing patents. In addition to these criteria, the user can create unlimited number of criteria for selecting patents based on keywords and bibliographic attributes. Each case is unchangeable after creation. However, it is possible to create a new case with a different set of patents combining existing cases. Patents should be entered directly number-by-number (PID) or as list in .csv form. The user interface (Fig. 3) is built using PHP, HTML and JavaScript programming languages as well as JQuery JavaScript library, DataTables and HighCharts library for displaying the results of data processing.

5 Implications for practice

In this section, the PSALM functionality will be demonstrated shading more light on Android litigation cases and Google's decision to buy Motorola Mobility. Google's Android is the most popular mobile operating system (OS) in the world. According to the recent statistics, approximately 75 % of the estimated 2.2 billion mobile devices worldwide are based on it [22]. Its huge success and popularity with technology companies has made it an attractive target for patent litigations. Android OS is one of key battlefields in the so-called "smartphone wars" between Apple, Microsoft, Oracle on one side and Google and companies that are using the Android software (e.g. Samsung and HTC) on the other.

On the day when Google bought Motorola Mobility, popular tech web site www.cnet.com published this information under the title: *Google just bought itself patent protection* [23]. The question is: Is this (completely) true? To find the answer to the question it is needed to explore and analyse patent litigations related to Android OS, to study relevant patent portfolios, and from that perspective reflect on Google's decision to buy Motorola Mobility.

The analysis was done in two steps. The first step was to search for all active litigations against products that are based on the Android OS between 2009 and 2012, and to analyse them using PSALM. 55 patents were detected [24]. From technology side, approximately half of these patents are related to broader functionality of smartphones, while others were spread across several categories, including operating system, user interface and power saving (Fig. 4).

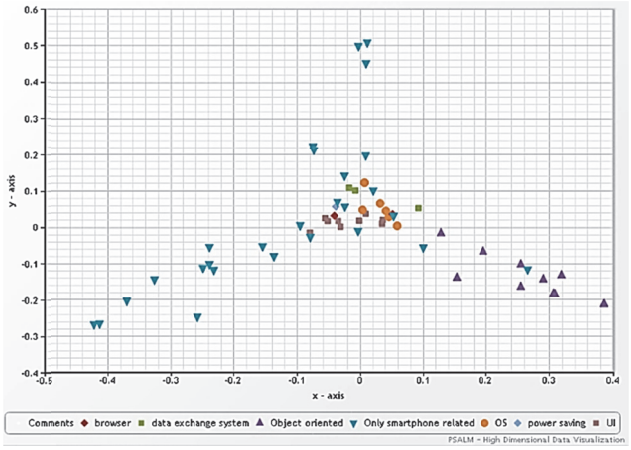

Figure 4 Technology groups of Android (litigated) patents

From the ownership perspective, the litigated patents were assigned to Apple, Microsoft, British Telecom and Sun Microsystems (Fig. 5).

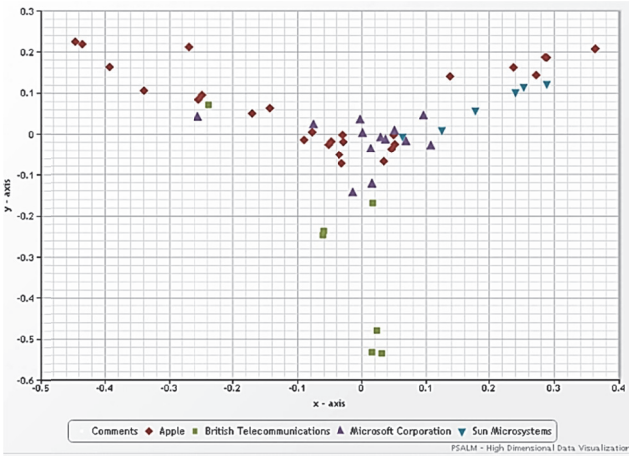

Figure 5 Owners of Android (litigated) patents

Step two was to identify and analyse the patents which belong to Motorola Mobility and are cited by 55 litigated patents. This was done because it is expected that citation analysis can lead to patents related to the same or very similar areas of technology. Analyses done by the tool indicated that 55 litigated patents cited 22 Motorola Mobility patents. Fig. 6 shows how these two groups of patents are distributed in 2D space and how Motorola Mobility patents match to litigated patents.

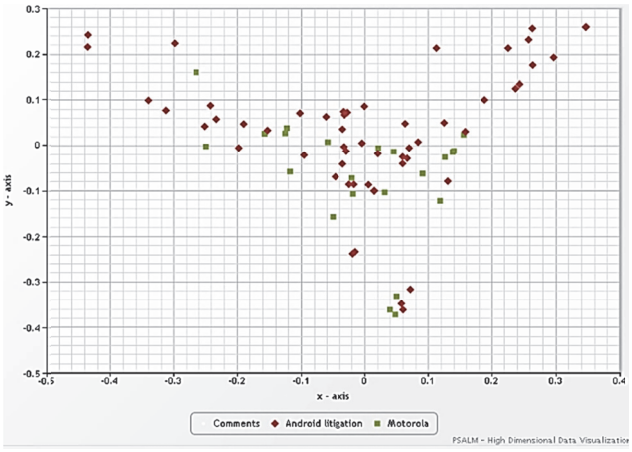

Figure 6 Android (litigated) patents vs. Motorola Mobility patents

Closeness of detected (Motorola Mobility patents) and litigated (Android related) patents revealed that Motorola's patents are relatively well distributed and related to patents which can harm Google. From that point, those who argued that Google's decision to buy Motorola Mobility was rooted in its patent portfolio were right. Full support for this understanding of the acquisition came directly from Google's CEO Larry Page in January 2014, after Google sold the device maker to Lenovo. He said [25]:

"We acquired Motorola in 2012 to help supercharge the Android ecosystem by creating a stronger patent portfolio for Google […] Motorola's patents have helped create a level playing field, which is good news for all Android's users and partners."

6 Conclusion

This paper described the structure and functions of developed software. It demonstrated efficiency and accuracy of two crucial functional modules – for patent collection and for clustering, and demonstrated PSALM usability through Android case study.

PSALM is a software tool for competitive intelligence based on patent data. It enables transformation of raw patent data into meaningful and useful information for business decision making. PSALM is a simple tool with good ergonomics. It enables easy patent search over a selected database on the Internet, automatic download and saving of selected patents in a local database. The tool provides its users with automatic analysis, enabling them to visualize low- and high-dimensional data from the patent, and to save and print out the analysis and reports. The real power of the tool is in analysing portfolios with a larger number of patents.

Patent data analyses will still be difficult, time and manpower consuming of experts' work, but PSALM could help in improving the correctness and timeliness of decision-making in competitive environment providing useful information and focusing experts' time and efforts on the most interesting and most promising patents. For example, based on PSALM results, it is easier to target technology weak areas and to group and select patents which could be interesting for the company.

Results presented in this paper are results of the current version of PSALM and improvements are expected in the next period. Beta version of PSALM is currently available to practitioners from RT-RK Computer Based Systems company, and the next version of the tool will be publicly available on the commercial basis. Further research will be directed towards tool improvement in text processing, using WordNET for comparing words in the text and SAO structures for text analysis. Also, future work will be concentrated on extending the test data set in order to further verify the results and improve data mining techniques, clustering and visualization modules. Main drawbacks of the tool, at the moment, are time needed for download of required data from USPTO web site into a local database, as well as fact that current version of the tool is using only US patents for analysis.

Acknowledgements

An earlier version of this paper was presented at the 14th International IEEE Symposium on Computational Intelligence and Informatics – CINTI 2013 in Budapest. Current version is significantly improved and changed based on comments from conference participants and progresses in the tool development.

This work was partially supported by the Ministry of Education, Science and Technology Development of the Republic Serbia under Grant number TR-32034, III-44009; and by the Provincial secretary of Science and Technology Development of Vojvodina Province under Grant number 114-451-2434/2011-03.

7 References

[1] EC; EPO, Why Researchers Should Care about Patents. European Commission and European Patent Office, 2007

[2] WIPO, Patents as a Source of Technological Information in the Technology Transfer Process (submitted by the Delegation of Spain). Intergovernmental Committee on Intellectual Property and Genetic Resources, Traditional Knowledge and Folklore, Sixth Session Geneva, March 15 to 19, 2004.

[3] Ernst, H. Patent Information for Strategic Technology Management. // World Patent Information. 25, 3(2003), pp. 233-242. DOI: 10.1016/S0172-2190(03)00077-2

[4] WIPO, World Intellectual Property Indicators 2014. WIPO Publication No. 941E/14, Geneva, 2014.

[5] Dou, H.; Leveillé, V.; Manullang, S.; Dou, J. M. Patent Analysis for Competitive Technical Intelligence and Innovative Thinking. // Data Science Journal. 4, 31(2005), pp. 209-237. DOI: 10.2481/dsj.4.209

[6] Ruotsalainen, L. Data Mining Tools for Technology and Competitive Intelligence. VTT Tiedotteita – Research Notes 2451, Espoo, 2008.

[7] Nikolic, LJ.; Kukolj, D.; Pokric, M.; Drazic, M.; Vuckovic, M.; Vitas, M. Web Robot – Patent Data Acquisition Software (in Serbian). // Proceedings of 56th conference for electronics, telecommunications, computers, automation, and nuclear engineering – ETRAN (RT 5.5) / Zlatibor, 2012, pp. 1-4.

[8] Tekic, Z.; Kukolj, D.; Nikolic, LJ.; Drazic, M.; Pokric, M.; Vitas, M.; Panjkov, Z.; Nemet, D. PSALM – Tool for business intelligence. // Proceedings of 35th Int. convention on information and communication technology, electronics and microelectronics – MIPRO / Opatija, (2012), pp. 1975-1980.

[9] Drazic, M.; Kukolj, D.; Vitas, M.; Pokric, M.; Manojlovic, S; Tekic, Z. Technology matching of the patent documents using clustering algorithms. // Proceedings of 14th Int. IEEE Symposium on Computational Intelligence and Informatics – CINTI 2013 / Budapest, (2013), pp. 405-409. DOI: 10.1109/cinti.2013.6705231

[10] Tekic, Z.; Drazic, M.; Kukolj, D.; Vitas, M. From patent data to business intelligence – PSALM case studies. // Procedia Engineering. 69 (2014), pp. 296-303. DOI: 10.1016/j.proeng.2014.02.235

[11] Wu, H. C.; Luk, R. W. P.; Wong, K. F.; Kwok, K. L. Interpreting TF–IDF Term Weights as Making Relevance Decisions. // ACM Transactions on Information Systems. 26, 3(2008), pp. 1-37. DOI: 10.1145/1361684.1361686

[12] Cox, T.; Cox, M. Multidimensional Scaling. // Chapman & Hall, London, 1994.

[13] Nock, R.; Nielsen, F. On Weighting Clustering. // IEEE Transactions on Pattern Analysis and Machine Intelligence. 28, 8(2006), pp. 1-13. DOI: 10.1109/TPAMI.2006.168

[14] MacQueen, J. B. Some methods for classification and analysis of multivariate observations. // Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability. University of California Press. (1967) pp. 281-297.
[15] Aloise, D.; Deshpande, A.; Hansen, P.; Popat, P. NP-hardness of Euclidean sum-of-squares clustering. // Machine Learning. 75, (2009), pp. 245-249. DOI: 10.1007/s10994-009-5103-0
[16] Du, Q.; Emelianenko, M.; Ju, L. Convergence of the Lloyd algorithm for computing centroidalVoronoi tessellations. // SIAM Journal on Numerical Analysis. 44, (2006), pp. 102-119. DOI: 10.1137/040617364
[17] Kukolj, D.; Atlagic, B.; Petrov, M. Unlabeled data clustering using are-organizing neural network. // Cybernetics and Systems, an Int. Journal. 37, 7(2006), pp. 779-790. DOI: 10.1080/01969720600887152
[18] Angelopoulou, A.; Psarrou, A.; Rodriguez, J.G.; Revett, K. Automatic landmarking of 2D medical shapes using the growing neural gas network. // Computer Vision for Biomedical Image Applications. (2005), pp. 210-219. DOI: 10.1007/11569541_22
[19] Martinetz, T.; Schulten, K. Topology representing networks. // Neural Networks. 7, 3 (1994) pp. 507-522. DOI: 10.1016/0893-6080(94)90109-0
[20] Keim, D. A. Information Visualization and Visual Data Mining. // IEEE Transactions on Visualization and Computer Graphics. 8, 1(2002), pp. 100-107. DOI: 10.1109/2945.981847
[21] Davies, D. L.; Bouldin, D. W. A Cluster Separation Measure // IEEE Transactions on Pattern Analysis and Machine Intelligence. PAMI-1 2 (1979), pp. 224-227. DOI: 10.1109/TPAMI.1979.4766909
[22] The Statistics Portal. Smartphone OS Worldwide by Installed Base in 2014. URL:www.statista.com/statistics/385001/smartphone-worldwide-installed-base-operating-systems/. (20.02.2015).
[23] Reardon, M. Google just bought itself patent protection CNET.com.URL:www.cnet.com/news/google-just-bought-itself-patent-protection/. (20.02.2015).
[24] Drazic, M. Contribution to The Solution of Automatic Processing of Patent Documents (in Serbian) // Master thesis, University of Novi Sad, 2012
[25] Page, L. Lenovo to Acquire Motorola Mobility. // Google Official Blog URL: http://googleblog.blogspot.ru/2014/01/lenovo-to-acquire-motorola-mobility.html. (20.02.2015).

Authors' addresses

Željko Tekić, PhD, Assistant Professor
Skolkovo Institute of Science and Technology
Nobel Street 3, 143026 Moscow, Russia
and
University of Novi Sad
Trg Dositeja Obradovića 6, 21000 Novi Sad, Serbia
E-mail: z.tekic@skoltech.ru

Miroslava Dražić, MSc, Senior Engineer
RT-RK Institute for Computer Based Systems
Narodnog fronta 23a, 21000 Novi Sad, Serbia
E-mail: miroslava.drazic@rt-rk.com

Dragan Kukolj, PhD, Professor
University of Novi Sad
Trg Dositeja Obradovića 6, 21000 Novi Sad, Serbia
E-mail: dragan.kukolj@rt-rk.uns.ac.rs

Ljubiša Nikolić, MSc, Engineer
RT-RK Institute for Computer Based Systems
Narodnog fronta 23a, 21000 Novi Sad, Serbia
E-mail: ljubisa.nikolic@rt-rk.com

Sandra Kukolj, MSc, Engineer
RT-RK Institute for Computer Based Systems
Narodnog fronta 23a, 21000 Novi Sad, Serbia
E-mail: sandra.kukolj@rt-rk.com

Milana Vitas, MSc, Engineer
RT-RK Institute for Computer Based Systems
Narodnog fronta 23a, 21000 Novi Sad, Serbia
E-mail: milana.vitas@rt-rk.com