Exploitation and Merge of Information Sources for Public Procurement Improvement

Exploitation and Merge of Information Sources for Public Procurement Improvement

Roberto Nai[0000−0003−4031−5376], Emilio Sulis[0000−0003−1746−3733], Paolo Pasteris[0000−0001−9638−9273], Mirko Giunta, and Rosa Meo[1 [0000−0002−0434−4850]

Computer Science Department, University of Turin
roberto.nai,emilio.sulis,paolo.pasteris,mirko.giunta,rosa.meo@unito.it
https://www.cs.unito.it/do/home.pl

Abstract. The analysis of big data on public procurements can improve the process of carrying out public tenders. The goal is to increase the quality and the correctness of the process, the efficiency of administrations, and reduce the time spent by economic operators and the costs of the public administrations. As a consequence, being able to recognize as early as possible if a public tender might contain some flaws, can enable a better relationship between the public organizations and the privates, and improve the economic conditions through the correct use of public funds. With the proliferation of e-procurement systems in the public sector, valuable and open information sources are available and can be accessed jointly. In particular, we consider the sentences published on the Italian Administrative Justice website and the Italian Anti-Corruption Authority database on public procurement. In this paper, we describe how to find connections between the procurement data and the appeals and how to exploit the resulting data for the measurement of litigation and clustering into communities the nodes representing entities having similar interests.

Keywords: Public Procurement · Open Data · Information Retrieval · Government Transparency.

1 Introduction

In the Internet age, the extraction of information from texts is of concern [15], as well as the use of search-based applications for the information sources integration [14,25]. This work investigates the automatic knowledge extraction from a set of public law archives. In particular, we focus on two legal datasets: first, the complete archive of the (Italian) National Anti-Corruption Authority (ANAC), which includes public procurement; and second, the vast online dataset of the appeals submitted to the Italian Administrative Justice (IAJ). Our work aims to find connections between the data on procurement in ANAC, and the judges' sentences on appeals in IAJ. The goal is to integrate the two information sources and gather the cases of a procurement whose execution leads to appeal to the administrative justice and to controversies between a public authority and a private company. Being able to extract this information automatically makes it possible to suggest possible ameliorative solutions to decision-makers and anticipate or prevent problematic cases for the governance. Our research question is:

(RQ) How can we automatically extract information from legal archives to identify the entities involved in a public procurement? To address the issue, we propose a methodological framework that employs both information retrieval and graph analysis. Graph analysis allows us to connect the related business entities in a graph and then identify the communities or "clusters" as the graph components that share something. The remainder of the paper is organised as follows: section 2 introduces the background with related works. In section 3 we describe the case study, in section 4 we describe the proposed methodology while section 5 provides insights about the results of the research. Finally, section 6 concludes the paper.

2 Related work

Web-based archives are growing steadily, following the Internet expansion in recent times, as evidenced by the importance of the non-profit online libraries[1]. Online archives facilitate the dissemination of information for professionals, citizens, and researchers [5]. Digital documents, as in the case of legal texts, allow ample opportunity to apply automatic information extraction techniques [23]. The Information Retrieval (IR) community has developed many systems to support research [22]. Some recent examples include the cases of knowledge extraction from a collection of legal documents [9], the summarization algorithms applied to legal case judgments [6], the co-occurrence network on European Directives [24], the shift towards Open Science [17]. Recently, researchers benefited from new tools, such as the IR software Apache Lucene [3]. Lucene is nowadays a well-known platform for building and deploying search-based applications [26]. In [8], starting from the Portuguese Public Procurement portal, a graph-oriented user interface is proposed to support decision-making, using Cypher queries [10]. Besides this, supervised machine learning methods are used to find suspicious procurement. The authors of [20] propose the SALER software prototype. Inside SALER, several internal and external data sources are analyzed and assessed to explore possible irregularities in budget and cash management, public service accounts, salaries, disbursement, grants, subsidies, etc. SALER employs graph databases, too. Unlike the previous works, our research is based on the merge of two separate legal datasets. Their joint use enables some key elements useful to solve some tasks. One is the determination of possible exemplars of inefficiency or irregularity in tenders through the application of machine learning models trained on the labels provided by the presence of judges' sentences on tenders. The second task is the recognition of sets of agents, like the public entities or the economic operators that show strong connections that make them a community that forms a "cluster".

3 Case study

The National Anti-Corruption Authority, abbreviated to ANAC, is an independent Italian administrative authority whose task is to prevent corruption in the Italian public administration, implement transparency and supervise public contracts. ANAC collects data on calls for procurement from the public contract authority and provides a catalog

of Open Data describing public procurement, contract authority, and contractors (interchangeably named economic operators). Currently, the ANAC website provides data on approximately 7.2 million of public tenders for procurement collected from the first of January 2007 to the end of March 2022 within a dataset collecting procurement whose cost is above 40 thousand euros. The Open Data is available on the ANAC website[1]. On the other side, the Italian Administrative Justice (IAJ) collects the judges' sentences related to the public procurement appeals. Currently, about 80,360 sentences are available on the website[2].

3.1 Data overview

Following 3, the ANAC dataset contains a table `Procurement` of 7,189,462 rows, a table `Contractors` of 42,393 rows that stores the public authorities, a table `Economic operators` of 265,039 rows about the successful bidders (also named economic operators), and finally a table `Awards` of 1.635.609 rows that reports the winner for the tenders. Unfortunately there are data quality problems in this table because it does not contain the winners for all the procurement. Each procurement is identified by an alphanumerical value called CIG (the key value). A procurement can be of three types: "supplies of goods", "public works", "services'". Figure 1 shows that about 50% of contract types are for goods/supplies, followed by services (35.872%) and lastly for public works (13,806%).

4 Methodology

4.1 Problem definition

Our methodological framework is grounded both in IR and in structured databases. They are complementary because the first one allows the efficient search in large corpora and the second to store large amounts of data and perform analytic. We applied IR to combine the two sources: procurement dataset (ANAC) and court rulings (IAJ). Following [27], the process of the full-text search is: build a texts database, create indexing, search and filter the results. Figure 2 resumes the applied workflow.

4.2 Data gathering

Regarding the ANAC dataset, we imported the Open Data on the procurement (in CSV format) into an InnoDB table of a MySQL database (whose size is 5.5 GB). We chose a relational database to maintain the relationship between a procurement, the contractor, and the successful bidders via the shared key of the CIG (the ID of each procurement). We obtained the IAJ judgments via web scraping. Since these are text files in HTML, Doc/Docx, and PDF format, they were indexed using Lucene. In addition, we imported

[1] https://www.anticorruzione.it
[2] https://www.giustizia-amministrativa.it

Table 1. Quantitative description of ANAC (tables `Procurement`, `Contractors`, `Economic operators` and `Awards`) and IAJ datasets

Topic	Value
Total number of procurement	7,189,462
Temporal range of procurement	January 2007 - March 2022
Identifier (key) value for every procurement	CIG (alphanumerical value)
Total number of contractors (public authority)	42,393
Total number of successful bidders (economic operators)	265,039
Total number of awarded procurements	1.635.577
Total number of bids for all the procurement	8.199.059
Average number of bids received per procurement	4.113
Procurement contract type	goods/supplies: 50.322% services: 35.872% public works: 13,806%
Number of procurement by area	ordinary: 86.202% special: 13.798%
Total number of appeals in IAJ	80,360
Identifier (key) value for every appeal	ECLI
Text file types in IAJ	html: 60,284 doc/docx: 20050/26 pdf: 12

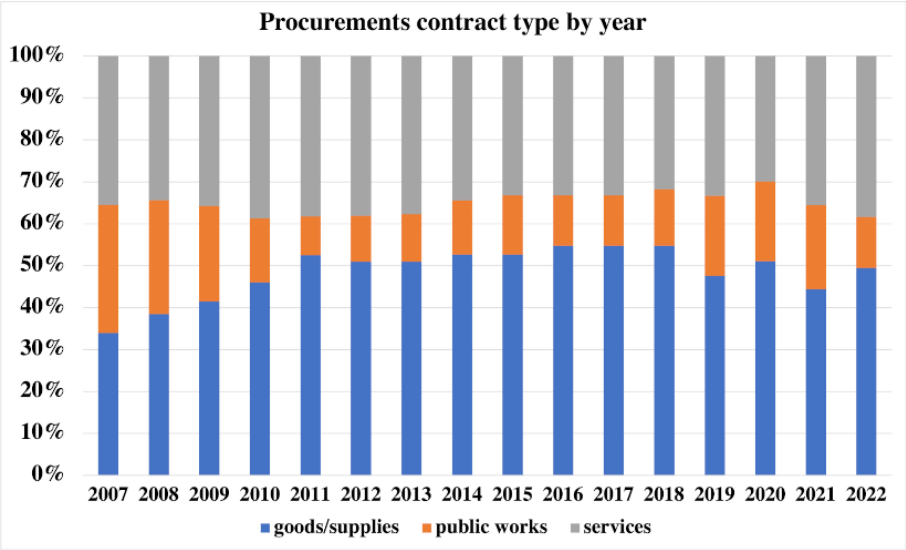

Fig. 1. Distribution of contract type from 2007 to 2022: from 2011 the main contract type concerns goods/supplies (blue bars), followed by services (grey) and public works (orange)

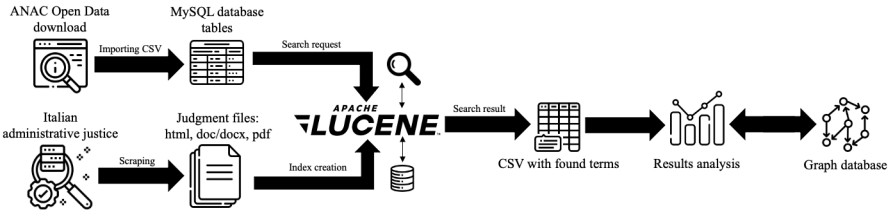

Fig. 2. Workflow of the research approach from data gathering to results analysis

the ANAC Open Data on the procurement between a public entity and a private company into the Neo4j[3] graph NoSQL database [2]. It permits the efficient navigation of the graphs formed by the related business entities and the application of analysis algorithms.

4.3 Technologies

We employed web scraping of the online archives from ANAC and the IAJ websites: we used the Python libraries with MechanicalSoup[4] and BeautifulSoup[5]. We opted for a relational database (MySQL) for the storage of the ANAC database, as the downloaded Open Data is organized in tabular form and referenced by key values. For IAJ, we developed a Java application for textual search with Lucene (being a native library in Java), since the downloaded files are textual. We analyzed the results with the Pandas libraries in Python and plotted them with Matplotlib. We performed the computations on a 2.8 GHz Intel Core i5 quad-core with 8 GB of RAM with SSD drive, without GPU support. The source code in Java of this work is publicly available on GitHub[6].

5 Results

5.1 Data indexing

Table 5.1 describes the results for each type of file of the 80,360 documents from the IAJ archive. Interestingly, Lucene indexed all the terms (614,696) in about 5 minutes. We excluded from indexing PDF files as well as image scans of old judgments (a negligible subset of only 12 files out of 80,360).

5.2 Search by procurement

The connection between the two information sources occurred by searching for the procurement ID (CIG) in the IAJ sentences archive; it yielded the results shown in

[3] https://neo4j.com
[4] https://mechanicalsoup.readthedocs.io/en/stable
[5] https://beautiful-soup-4.readthedocs.io/en/latest
[6] https://github.com/roberto-nai-unito/ANACLucene

Table 2. Indexing performance of Lucene

File type	Number of indexed files	Index size	Index time
HTML	60284	407 MB	2.577 min
DOC/DOCX	20050/26	135 MB	1.551 min

Table 3. It is worth mentioning that the search for $7,189,462$ terms in about $80,000$ files took 24 hours. The total number of CIG found is $8,062$: this means that the probability that a sentence in our archive refers to a CIG is 10% only. We continued the integration of the two datasets by performing a search via Lucene: we obtained further information on the procurement with sentences. As a result we computed the bar chart shown in Figure 3 and show which courts deliberated most on procurement. In preparation for this bar chart, we exploited the ECLI code (the key of an appeal). Moreover, following the procurement types of Section 3.1, the highest number of judgments is related to the "services" procurement type (about 61%). Instead, the "supplies of goods" and "public works" have a lower ratio. The "ordinary" area has the highest percent of procurement (about 89%). Finally, the procurement in the "special" area rarely has an appeal. This result leads to a transparency gain in the search for courts with a high number of appeals and the identification of the most problematic kind of procurement.

Table 3. Quantitative description of ANAC procurement by application of Lucene on the procurement ID (CIG) inside sentences

Topic	Value
Procurement type	services: 61.634% public works: 22.163% goods/supplies: 16.248%
Procurement area	ordinary: 88.714% special: 11.331%

5.3 Search by contract authority and economic operators' denomination

The search by contract authority and economic operators' denominations yielded the results shown in the first row of Table 4. The search for $42,393$ contracting authorities' names in about $80,000$ files took 23 minutes. The search for $265,039$ economic operators' names in about $80,000$ files took about 2 hours and 15 minutes. This second result may be useful in bringing transparency to the contracting authority most affected by appeals from the economic operators: a higher presence may indicate greater "aggressiveness" toward an administration resulting in inefficiency in the implementation of the intended public tenders (Section 6).

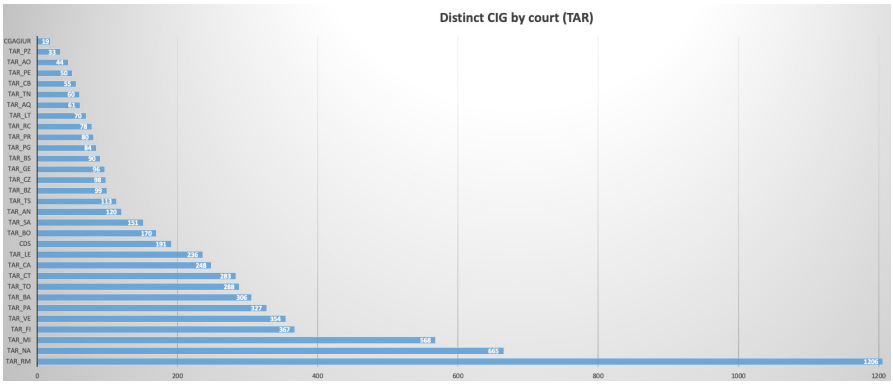

Fig. 3. Bar chart on the distribution by court of procurement with an ID (CIG): at the top there are the courts with the lowest number of judgements; in the bottom, the courts with a higher number

Table 4. Search result of the contracting authorities and economic operators in sentences by Lucene

Type	Names found (total)	Names found (percent)	Time
contracting authorities	37,890	6.164%	23 min
economic operators	152,934	24.880%	2h:15m

5.4 Definition of the litigation measure with estimation of participation in public tenders

In both ANAC and IAJ datasets, it is not easy to infer the identity of the companies that participated in tenders in case they did not win. Table *Awards* contains only the total number of participants for each tender and includes only the winner's identity. However, being able to infer the participation for an economic operator could be useful to estimate the amount of administrative litigation with reference to the number of participation which is of particular interest because much litigation is caused when participants did not win.

One of the goals of our work is to define a way to measure the tendency to litigation of the economic operators. With reference to the generic economic operator i the first measure is:

$$\frac{Number\_appeals\_generated\_by\_i}{Number\_of\_bids\_won\_by\_i} \tag{1}$$

where the numerator is obtained by collecting the sentences in which the economic operator i is the generator of the appeal and the denominator is obtained by table *Awards*. We computed the number at the numerator by means of the identification of the economic operator who started the appeal. This is not an immediate task. To this aim we first applied BeautifulSoup in the identification of the initial section of the judge's sentence that contains the denomination of the economic operator who started the appeal. In the second step, we applied Lucene to unify the multiple possible denominations for

each economic operator into a single one stored in an internal dictionary. Unfortunately this equation is not suitable to estimate the litigation for those companies that did not win any tender or won just a few because the little number at the denomination inflates the evaluation.

The second measure is:

$$\frac{Number\_appeals\_generated\_by\_i}{Number\_of\_bids\_attempted\_by\_i} \tag{2}$$

that differs from equation 1 on the denominator that is the number of tenders in which the economic operator participated by not necessarily won. We would prefer to use equation 2 because appeals are often generated by participants who only attempted but did not win the tender. We show how we estimate the denominator, i.e., the number of economic operators' attempts to award a contract.

We assumed the number of attempts done by economic operators to win a public contract is similarly distributed as the number of awarded tenders per company, i.e., as a power-law distribution function, but differing from it by a constant factor that corresponds to the probability that a company, participating a tender awards the contract. We now try to estimate this probability that we call *p_awd* (probability of award).

We start from the table *Awards* of the awarded procurement. We calculated the probability a company awards a tender as the proportion of the number of success cases over the total number of cases. The number of success cases is the number of tenders for which we know there is a winner economic operator. The total number of cases is the total number of received bids in those tenders. In some cases (corresponding to $628,703$ tenders, approximately 8.3% of the cases) this number was not specified, and we assumed it was one (presuming the public authority deemed it useless to communicate in case there was a single participant). Even if *p_awd* might differ from company to company (the most successful ones will have a higher probability of winning a contract than others) we assumed this probability is approximately constant over the population of the economic operators. As said, *p_awd* was computed as:

$$p\_awd = \frac{Total\_number\_of\_tenders\_awarded}{Total\_number\_of\_bids} = 0.199 \tag{3}$$

that corresponds to estimating that an economic operator wins a contract for every five participation to tenders. Equation 3 is useful to determine the litigation measure of equation 2 and in particular the denominator by multiplication of the scaling factor obtained by equation 3. Thus, we obtain the litigation measure of equation 2 from the litigation measure of equation 1 multiplied by the scaling factor of 0.199. This result is important in the analysis of tenders participants to determine the ones with a high probability of litigation - an essential issue for the reduction of the overload on the justice.

As it is possible to see in Figure 4, the plot of the cumulative distribution of the litigation measure in the logarithmic scale computed by equation 1 assumes an approximate linear form that corresponds to the power-law distribution. We fitted it by application of the Maximum Likelihood Estimation [12] and obtained the parameter of the exponential distribution equal to 6, corresponding to a strongly skewed distribution.

On this distribution, we can rank the companies according to the litigation measure and find a threshold *x_crit*. *x_crit* is a critical value above which the probability to find a company with a litigation measure higher than this value is bounded by the confidence level α. α can be set to an arbitrarily low value (customarily set to 5% or 1%). In the case of Figure 4 the threshold of the litigation measure is 19 for an $\alpha = 5\%$ (and of 85 for an $\alpha = 1\%$). This corresponds to saying that for every contract awarded to the companies with an extremely high tendency to litigation, the administrative justice expects to receive as many (and more) than 85 appeals. If we consider equation 2, the threshold is 16.9 for an $\alpha = 5\%$. It corresponds to saying that for every company with an extremely high tendency to litigation that participates in tenders, the administrative justice expects to receive almost 17 appeals. This occurs with a probability of $\alpha = 5\%$.

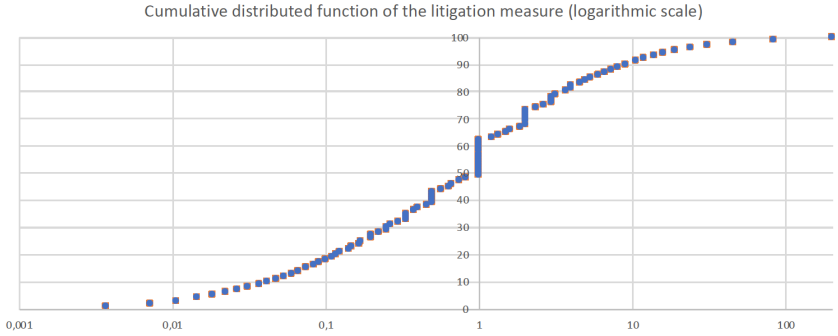

Fig. 4. Cumulative distribution function of the litigation measure obtained using equation 1

5.5 Analysis on the Graph

Following the Section 4.2, the graph database in Neo4j was constructed using a contract authority and an economic operator as nodes, while an edge represents a CIG that identifies the procurement won by the economic operator with the specific contract authority. An exemplary part of the obtained graph database is shown in Figure 5.

Following [4], we decided to use Neo4j due it's powerful visualization tools; we also exploited the fact that Neo4j contains the software library "Graph Data Science" (GDS) [13]; GDS was used because the algorithms of interest for this research (community detection and betweenness centrality) are built into the tool, thus avoiding the need to use other external applications. We applied two main graph algorithms to analyze the graph: community detection and betweenness centrality detection. Although we have extracted the communities at the structural level, we have extracted the community data noting that the community is homogeneous in supplies and contracts.

Community detection The Neo4j GDS library contains the Louvain method: it is an algorithm to detect communities in large networks [19]. It maximizes the modularity

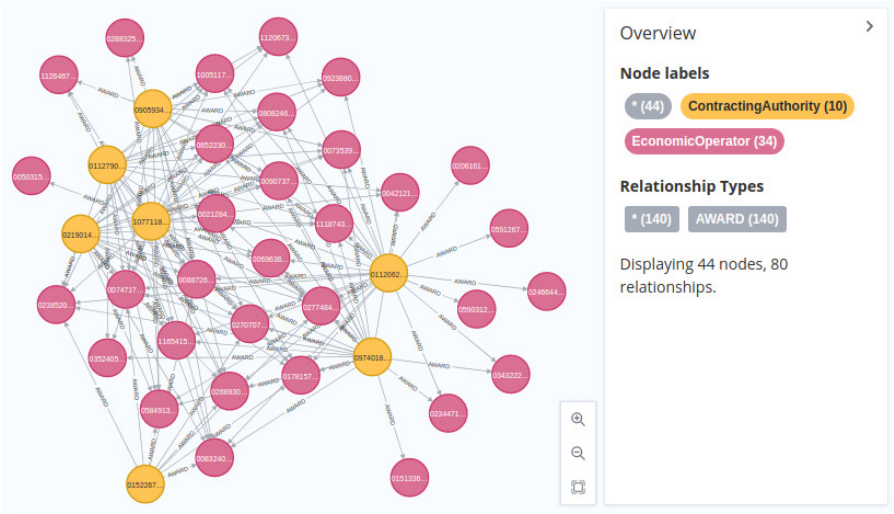

Fig. 5. Part of the graph obtained in Neo4j containing the relationship AWARD (edges) between the contracting authorities (yellow nodes) and the economic operators (pink nodes)

score for each community, where the modularity quantifies the quality of a node assignment to communities. This means evaluating how much more densely connected the nodes within a community are, compared to how connected they would be in a random network [16]. The Louvain algorithm is a hierarchical clustering algorithm, that recursively merges communities into a single node and executes the modularity clustering on the condensed graphs [11]. Figure 6 shows the first five communities detected by the algorithm. It is possible to see that the first community is composed by two subcommunities and contains 38 nodes (4 contracting authorities and 34 economic operators) while the second community is composed by 12 nodes (1 contracting authority and 11 economic operators). These clusters or communities can be used for various transparency analysis. For example, anomaly detection can be carried out by asking how likely a particular entity (contracting authority or economic operator) in a given cluster is likely to make a transaction (winning a procurement) with an arbitrarily selected cluster; the less likely that transaction takes place, the higher the assigned anomaly score.

Betweenness centrality Betweenness centrality is often used to find the nodes that serve as a bridge from one part of a graph to another. The algorithm calculates unweighted shortest paths between all pairs of nodes in a graph. Each node receives a score, based on the number of shortest paths that pass through the node. Nodes that more frequently lie on shortest paths between other nodes will have higher betweenness centrality scores [19]. The Neo4j GDS library implementation is based on Brandes' approximate algorithm for unweighted graphs [7]. Figure 8 shows the first ten nodes with higher centrality score; these measures can help shed light on the accounts (economic

Community	Communities	Size	Nodes
242	238,242	38	10771180014 01127900049 01120620057 02190140067 09238800156 08082461008 09012850153 02774840595 00832400154 05903120631 11206730159 00488410010 05912670964 02385200122 04411460639 11264670156 10172190018 02006400960 07123400157 11654150157 06032681006 00747170157 01781570591 11187430159 00735390155 10051170156 02061610792 00421210485 06522300968 02466440167 00503151201 00807970157 05849130157 00887261006 02707070963 00696360155 02689300123 09284460962
307	307,307	12	09740180014 01542210222 00674840152 00212840235 03524050238 01423300183 00907371009 02344710484 00803890151 03432221202 01513360345 06209390969
242	242,242	5	01522670031 09059340019 00076670595 09699320017 02883250017
261	261,261	5	00514490010 07820120017 03747970014 03717670016 06484280018
343	343,343	4	02078000037 00967720285 01481120697 00468910070

Fig. 6. First five communities detected by the Louvain algorithm: green nodes are contracting authorities; orange nodes are economic operators. The communities are in descending order according to their "Size" (number of nodes)

Fig. 7. Graphical representation of the community detection: light blue nodes are the contracting authorities; economic operators are the blue nodes

operators) which are the most central to the entire transaction network and help to identify suspiciously well-connected accounts.

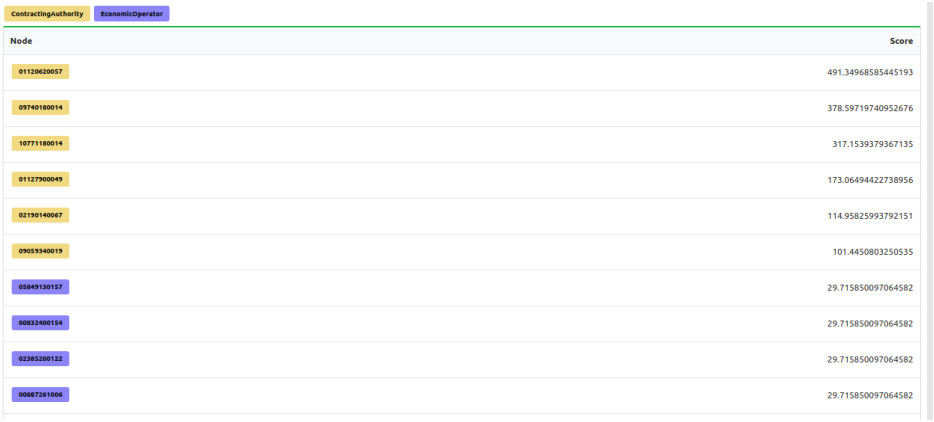

Fig. 8. First ten nodes with higher centrality score: yellow nodes are contracting authorities; orange nodes are economic operators. The "Score" on the right side indicates how central the node is; nodes are sorted in descending order according to this value

Fig. 9. Graphical representation of the betweenness centrality: light blue are contracting authorities; blue nodes are economic operators

6 Conclusions and future work

In this paper, we explored the possibility of the integration by IR of two information sources (ANAC and IAJ) about procurement using common data in both datasets. By

fitting models on observed data applying the principle of MLE, we estimated the probability that a company awards a tender and the number of participation. These are the ingredients for the identification of the companies that cause the highest number of litigation whose elimination could drastically improve the justice overload. We applied also graph analytic to identify the communities formed by the public contractors and economic operators with recurrent procurement. As future work, we plan to study the use of Legal BERT [21] to search within the judgments for named entities [18] such as the names of the economic operators that were excluded from the tender selection (thus not tracked in the ANAC dataset) in order to create a graph database of the economic operators that may appear in the appeals despite an unsuccessful bid.

References

1. AlNoamany, Y., Alsum, A., Weigle, M.C., Nelson, M.L.: Who and what links to the internet archive. Int. J. Digit. Libr. **14**(3-4), 101–115 (2014). https://doi.org/10.1007/s00799-014-0111-5, https://doi.org/10.1007/s00799-014-0111-5
2. Angles, R., Gutierrez, C.: Survey of graph database models. ACM Computing Surveys (CSUR) **40**(1), 1–39 (2008)
3. Azzopardi, L., Moshfeghi, Y., Halvey, M., Alkhawaldeh, R.S., Balog, K., Di Buccio, E., Ceccarelli, D., Fernández-Luna, J.M., Hull, C., Mannix, J., et al.: Lucene4IR: Developing information retrieval evaluation resources using Lucene. In: ACM SIGIR Forum. vol. 50, pp. 58–75. ACM New York, NY, USA (2017)
4. Baton, J., Van Bruggen, R.: Learning Neo4j 3. x: Effective data modeling, performance tuning and data visualization techniques in Neo4j. Packt Publishing Ltd (2017)
5. Berget, G., Hall, M.M., Brenn, D., Kumpulainen, S. (eds.): Linking Theory and Practice of Digital Libraries - 25th International Conference on Theory and Practice of Digital Libraries, TPDL 2021, Virtual Event, September 13-17, 2021, Proceedings, Lecture Notes in Computer Science, vol. 12866. Springer (2021). https://doi.org/10.1007/978-3-030-86324-1, https://doi.org/10.1007/978-3-030-86324-1
6. Bhattacharya, P., Hiware, K., Rajgaria, S., Pochhi, N., Ghosh, K., Ghosh, S.: A comparative study of summarization algorithms applied to legal case judgments. In: Azzopardi, L., Stein, B., Fuhr, N., Mayr, P., Hauff, C., Hiemstra, D. (eds.) Advances in Information Retrieval - 41st European Conference on IR Research, ECIR 2019, Cologne, Germany, April 14-18, 2019, Proceedings, Part I. Lecture Notes in Computer Science, vol. 11437, pp. 413–428. Springer (2019). https://doi.org/10.1007/978-3-030-15712-8_27, https://doi.org/10.1007/978-3-030-15712-8_27
7. Brandes, U., Pich, C.: Centrality estimation in large networks. International Journal of Bifurcation and Chaos **17**(07), 2303–2318 (2007)
8. Carneiro, D., Veloso, P., Ventura, A., Palumbo, G., Costa, J.: Network Analysis for Fraud Detection in Portuguese Public Procurement, pp. 390–401 (10 2020). https://doi.org/10.1007/978-3-030-62365-4_37
9. Castano, S., Falduti, M., Ferrara, A., Montanelli, S.: A knowledge-centered framework for exploration and retrieval of legal documents. Inf. Syst. **106**, 101842 (2022). https://doi.org/10.1016/j.is.2021.101842, https://doi.org/10.1016/j.is.2021.101842
10. Francis, N., Green, A., Guagliardo, P., Libkin, L., Lindaaker, T., Marsault, V., Plantikow, S., Rydberg, M., Selmer, P., Taylor, A.: Cypher: An evolving query language for property graphs. In: Proceedings of the 2018 International Conference on Management of Data. pp. 1433–1445 (2018)

11. Ghosh, S., Halappanavar, M., Tumeo, A., Kalyanaraman, A., Lu, H., Chavarria-Miranda, D., Khan, A., Gebremedhin, A.: Distributed Louvain algorithm for graph community detection. In: 2018 IEEE international parallel and distributed processing symposium (IPDPS). pp. 885–895. IEEE (2018)
12. Goldstein, M.L., Morris, S.A., Yen, G.G.: Problems with fitting to the power-law distribution. The European Physical Journal B-Condensed Matter and Complex Systems **41**(2), 255–258 (2004)
13. Hodler, A.E., Needham, M.: Graph Data Science Using Neo4j. In: Massive Graph Analytics, pp. 433–457. Chapman and Hall/CRC
14. Konchady, M.: Building Search Applications: Lucene, LingPipe, and Gate. Lulu.com (2008)
15. Lakhara, S., Mishra, N.: Desktop full-text searching based on Lucene: A review. In: 2017 IEEE International Conference on Power, Control, Signals and Instrumentation Engineering (ICPCSI). pp. 2434–2438 (2017). https://doi.org/10.1109/ICPCSI.2017.8392154
16. Lu, H., Halappanavar, M., Kalyanaraman, A.: Parallel heuristics for scalable community detection. Parallel Computing **47**, 19–37 (2015)
17. Manghi, P., Candela, L., Lazzeri, E., Silvello, G.: Digital libraries: Supporting open science. SIGMOD Rec. **48**(4), 54–57 (2019). https://doi.org/10.1145/3385658.3385669, https://doi.org/10.1145/3385658.3385669
18. Nadeau, D., Sekine, S.: A survey of named entity recognition and classification. Lingvisticae Investigationes **30**(1), 3–26 (2007)
19. Needham, M., Hodler, A.E.: A comprehensive guide to graph algorithms in neo4j. Neo4j.com (2018)
20. Plumed, F., Casamayor, J., Ferri, C., Gómez, J., Vendrell Vidal, E.: SALER: A Data Science Solution to Detect and Prevent Corruption in Public Administration, pp. 103–117 (01 2019). https://doi.org/10.1007/978-3-030-13453-2_9
21. Ravichandiran, S.: Getting Started with Google BERT: Build and train state-of-the-art natural language processing models using BERT. Packt Publishing Ltd (2021)
22. Sansone, C., Sperlí, G.: Legal information retrieval systems: State-of-the-art and open issues. Inf. Syst. **106**, 101967 (2022). https://doi.org/10.1016/j.is.2021.101967, https://doi.org/10.1016/j.is.2021.101967
23. Solihin, F., Budi, I., Aji, R.F., Makarim, E.: Advancement of information extraction use in legal documents. International Review of Law, Computers & Technology **35**(3), 322–351 (2021). https://doi.org/10.1080/13600869.2021.1964225, https://doi.org/10.1080/13600869.2021.1964225
24. Sulis, E., Humphreys, L., Vernero, F., Amantea, I.A., Audrito, D., Caro, L.D.: Exploiting co-occurrence networks for classification of implicit inter-relationships in legal texts. Inf. Syst. **106**, 101821 (2022). https://doi.org/10.1016/j.is.2021.101821, https://doi.org/10.1016/j.is.2021.101821
25. Wikipedia: Search-based application (June 2022), https://en.wikipedia.org/wiki/Search-based_application
26. Yang, P., Fang, H., Lin, J.: Anserini: Enabling the Use of Lucene for Information Retrieval Research. In: Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval. p. 1253–1256. SIGIR '17, Association for Computing Machinery, New York, NY, USA (2017). https://doi.org/10.1145/3077136.3080721, https://doi.org/10.1145/3077136.3080721
27. Zhang, Y., Li, J.l.: Research and improvement of search engine based on lucene. In: 2009 International Conference on Intelligent Human-Machine Systems and Cybernetics. vol. 2, pp. 270–273. IEEE (2009)