To Join or Not to Join: An Analysis on the Usefulness of Joining Tables in Open Government Data Portals

Arif Usta[1], Semih Salihoğlu[1]

[1]University of Waterloo, Waterloo, Ontario, Canada

Abstract

Governments have initiated national programs to make government data publicly available with the purpose of improving transparency and making it easier for general public to access information of interest about many aspects of their countries. Thousands of datasets are being continuously published in Open Government Data Portals (OGDP) for public use, which makes them an attractive data repository for researchers to study data integration problem. One common application of data integration is join operation to expand a table with additional columns, for which many studies have been proposed in the literature. However, usefulness of end result after joining of potential table pairs is under-explored, especially considering heterogeneous nature of OGDPs. To this end, we analyze joinability of tables based on high value overlap in several, English-speaking OGDPs; Canada, Singapore, UK, and US. Our analysis reveals that mainly due to high value repetition and nonexistence of key columns, vast majority of the joinable table pairs are accidental, resulting in uninterpretable tables.

Keywords

tabular data analysis, open data, data integration, joinability

1. Introduction

The launch of OGDPs, such as data.gov, open.canada.ca, or data.gov.in, has popularized the open data movement of the last decade. The overarching vision of OGDPs is to make governments transparent so that journalists, policy analysts, researchers, and the general public can easily monitor how their societies are functioning. Achieving this vision requires developing additional tools and applications over these datasets to discover, understand, link, and integrate them. Excitingly, these are some of the core research problems that interest the database and information retrieval communities, and as such OGDPs have become some of the most popular data repositories (aka *data lakes*) to study [1, 2, 3, 4, 5, 6, 7, 8, 9].

An important prerequisite for building better data tools around open datasets is to understand the properties of these datasets. Previous empirical studies have focused on analyzing either the metadata on the webpages that publish these datasets [10, 11], or detailed technical aspects of the files that store these datasets, such as the delimiters used and the location of the headers of CSV files [12].

In this paper, we focus on tabular datasets, and study properties related to their contents. In particular, we analyze them from the perspective of data design and examine how these characteristics impact finding joinable table candidates and resulting table after the join. We examine tabular datasets both programmatically and manually from four OGDPs that publish in English and follow the same publishing structure (CKAN[1]); Canada (CA) [13], Singapore(SG) [14], UK(UK) [15], and USA(US) [16]. In our large scale analysis, our main findings are as follows:

- Tables exhibit high-degree of denormalization regarding the perspective of data design; high value repetition, plethora of functional dependencies (FD), absence of key and more importantly identifying columns, and prevalence of multi-attribute composite keys.
- We found out that the denormalized nature of tables has significant implications on data integration operations such as join. Overwhelming majority of joinable pairs, even with a conservative, value-overlapping-based approach, are accidental, i.e., useless. The joins happen mostly between non-key columns, resulting in significantly bigger tables in rows which is contrary to most common join case exercised in relational databases, that is to expand a table with a column. Even the joins that occur in the presence of a key

Joint Workshops at 49th International Conference on Very Large Data Bases (VLDBW'23) — TaDA'23: Tabular Data Analysis Workshop, August 28 - September 1, 2023, Vancouver, Canada

arif.usta@uwaterloo.ca (A. Usta);
semih.salihoglu@uwaterloo.ca (S. Salihoğlu)

[1]https://ckan.org/

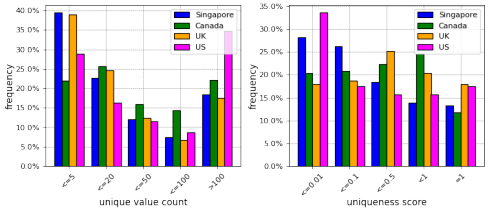

Figure 1: Unique value count and uniqueness score distributions for columns across portals.

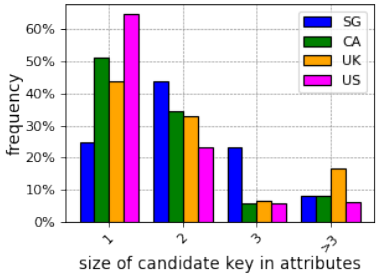

Figure 2: Distribution of candidate key sizes.

column are mostly accidental due to either columns having the same domain from tables with irrelevant context or key but non-identifying columns.

2. Analysis

2.1. Normalization Analysis

2.1.1. Uniqueness and Key Column Analysis

For a column c, let c's *uniqueness score* be $\frac{|set(c)|}{|c|}$, which is the ratio of the number of unique values vs number of values in c (the latter is equivalent to the number of rows in the table c belongs to). Distributions of number of unique value counts and uniqueness scores of columns are depicted in Figure 1. We also recorded the median unique value counts for all portals, which are 10, 23, 10, and 30 for SG, CA, UK, and US, respectively.

There is a very high degree of value repetition across all portals. Almost half of the columns for all the portals have equal to or less than 20 unique value, which consequently leads into non-descriptive columns with significantly low uniqueness scores. For instance, 51% and 41% of the columns in US and CA, respectively, have smaller than 0.1 uniqueness score, i.e., each value in these columns are on average repeated more than 10 times.

We next analyze the distributions of key columns. A column c with uniqueness score of 1.0 is a key column. Key columns are desirable as they help identify a table's records. Furthermore, in data integration, joins of two tables on two key columns lead to non-growing joins, which are desirable as they effectively extend these tables with additional columns. For those tables that do not have a key column, we searched for all possible 2-size and 3-size candidate keys. The distribution of the minimum candidate key columns of the tables are depicted in Figure 2.

A very large number of tables, 58%, 53%, 50%, and 33% in SG, CA, UK, and US, respectively, do not have any single key columns. Therefore data systems, such as search engines that index records, may need to find composite keys to identify majority of the records in some portals. Furthermore, 10% of the tables across all portals do not have a candidate key of size 1, 2, or 3, which indicates the extent of denormalization and value repetitions in these portals.

2.1.2. Functional Dependency (FD) Analysis

Next, we analyze the prevalence of non-trivial FDs in OGDPs. Recall that an FD [17] in a table T is an expression $X \rightarrow A$ where $X \subseteq attr(T)$ and $A \in attr(T)$, which informally indicates that a specific set of X values imply the same A values in T. Formally, $X \rightarrow A$ holds iff for any pairs of tuples $t_u, t_v \in T$ if $t_u[X] = t_v[X]$, then $t_u[A] = t_v[A]$. $X \rightarrow A$ is *trivial* if $A \subseteq X$ or if X forms candidate key. It is well known that existence of non-trivial FDs indicates poor relation design and leads to value repetitions that can be avoided by decomposing the relation into Boyce Codd normal form (BCNF). In the rest of this section, LHS and RHS stand for the left- and right-hand side of an FD, respectively. While all our previous analyses used all datasets in each OGDP, our next analyses on composite keys and FDs require super-linear computations and for these we used tables with $10 \leq t \leq 10000$ tuples and $5 \leq c \leq 20$ columns. The final number of tables along with some other statistics from the sample are provided in Table 1.

To find FDs in tables, we implemented the *FUN* algorithm for finding FDs [18] and limited the algorithm to find FDs whose LHS contain at most 4 attributes. Table 1 shows the percentages of the tables for which we found at least 1 FD across all portals.

Majority of tables in each portal, and overwhelming majority in UK (84.05%) and US (79.86%), have non-trivial FDs. These percentages indicate that most of the table published by OGDPs are not in Boyce Codd normal form, so up to the common normalization standards of relational tables in practice.

Finally, we note that in most of the tables, the FDs have a simple structure where a single attribute on the LHS implies columns on the RHS. Such FDs indicate a direct dependency between two columns in a table. A classic example of such FD is *City* $\rightarrow$ *Province*, which is prevalent in the Canadian portal. As shown in Table 1 ("tables with a non-trivial FD s.t |LHS|=1" lines), the majority of the tables that have a non-trivial FD has a non-trivial FD in this simple form.

Table 1
FD statistics of the tables.

	Portal			
	SG	CA	UK	US
total # tables	701	7492	18864	9770
total # columns	4142	76976	189930	102118
avg # columns per table	5.91	10.27	10.07	10.45
# tables with a non-trivial FD	381 (54.35%)	5500 (73.41%)	15855 (84.05%)	7802 (79.86%)
# tables with a non-trivial FD s.t |LHS|=1	318 (45.36%)	3659 (48.83%)	12998 (68.90%)	5944 (60.84%)

Table 2
Main statistics of the joinable pairs for each portal.

	Portal			
	SG	CA	UK	US
total # joinable pairs	28770	268103	616956	3786199
total # tables	2376	14707	33359	25857
# joinable tables	1578 (66.4%)	8286 (56.3%)	16157 (48.4%)	14208 (54.9%)
total # columns	12428	194022	405093	374400
# joinable columns	1962 (15.8%)	25975 (13.4%)	48221 (11.9%)	66493 (17.8%)
# key joinable columns	410(20.9%)	5311(20.4%)	11722(24.3%)	11918(17.9%)
# non-key joinable columns	1552(79.1%)	20664(79.6%)	36499(75.7%)	54575(82.1%)

2.2. Joinability Analysis

Throughout our analyses, we define *joinable pairs* as quadruplets (t_i, c_k^i, t_j, c_l^j), where (t_i, t_j) are found to be *joinable tables* through the pair of *joinable columns* (c_k^i, c_l^j). Similar to many prior studies [1, 19, 20, 4, 21, 22], we consider an equi-join operation. We use *Jaccard* similarity as a metric of joinability, since in terms of precision it is regarded as highly effective [21, 22]. We used all available tables from each portal and picked all joinable pairs within the same portal, i.e., pairs of tables and joinable columns (t_i, c_k^i, t_j, c_l^j) and filtered out joinable pairs based on two criteria:

- *High Jaccard similarity:* Since our overall goal is to analyze useful joinable pairs, we wanted the resulting joins to not filter many tuples from the tables and picked pairs only if their join columns had very high, at least 0.9, Jaccard similarity value.
- *High unique values:* We selected pairs only if their columns had at least 10 unique values. Very small domains tend to have high value repetitions and lead to very large join outputs, which we assumed are not useful in data integration operations.

2.2.1. General Characteristics of Joinable Pairs

Table 2 reports the general statistics of the joinable pairs that we analyzed. Between 48.4% (UK) and 66.4% (SG) of the total tables in each portal have at least one other joinable table on some column. In contrast, only between 11.9% (UK) to 17.8% (US) of the columns have another column they are joinable with, of which only 17.9% (US) to 24.3% (UK) were key columns. We manually analyzed some tables and columns with high-degree joinability and observed three patterns:

- Tables with the same schema: There are large sets of tables that have the same or almost the same schema, e.g., because these are *periodically published tables*, which refers to publication style prevalent in OGDPs to store information weekly, monthly, annually. These tables tend to have many columns that have exactly the same domain and tend to be all pairwise joinable.
- Tables in the same dataset: Another common publication style in OGDPs is to have multiple tables storing information about different aspects about an entity, which we refer to as *semi-normalized tables*. The schemas of these tables tend to be different but still have common columns with significant value overlaps. These tables can be seen as normalized versions of a larger table yet can still exhibit FDs.
- Common non-descriptive columns: Some columns, such as state or year, exist in many tables and lead to high joinability degrees.

We next analyzed the sizes of the outputs of the joins, i.e., *expansion ratio* of the joins, which we define as: output size of the join / the size of the larger table. Expansion ratio distributions for all portals are depicted as letter-value plots in Figure 3. The biggest box in each distribution represents values between the 1st and 3rd quartiles. Vertical line in the biggest box represents median expansion ratios, which we

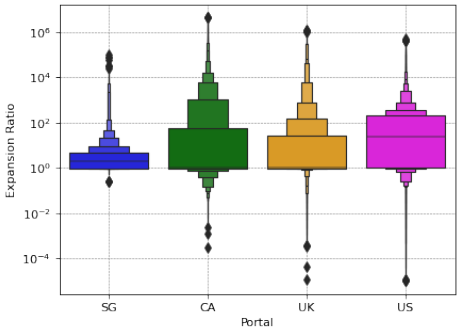

Figure 3: Distribution of expansion ratios of joinable pairs.

found as 1 for CA and UK, 2 for SG and 24 for US. As shown in the plot, except in SG, very large fractions of joinable pairs grow significantly, often beyond 10. For example in the US, the majority grows beyond 24 and there are at least 25% of the pairs that have an expansion ratio of above 100.

Although it is not possible to infer whether a particular join is useful only by inspecting its expansion ratio, perhaps the most common motivating case for joins is to extend one table with a new column, without growing the table at all, e.g., to add a new property of an entity in a table as a new column. If the expansion rate of a join is very high it is safe to assume that the joins are accidental.

2.2.2. Useful vs Accidental Pair Analysis

We sampled a large set of 450 pairs of tables (excluding SG due to common publication practice skewing the sample) from all of the pairs, and manually labeled them as accidental vs useful [2]. For the sample, we omit pairs with the same schema (i.e., list of column names), since these dominate the joinable space and are better candidates for union operation. We categorized the tables into 3 as follows:

- Unrelated Tables and Accidental (U-Acc): These are the clear false positive pairs of tables that come from completely different domains (e.g., crime vs health) and happen to have columns with high value overlaps.
- Related Tables and Accidental (R-Acc): These are pairs that originate from the tables storing same or similar information in a same context (e.g., health), but the join is accidental because the join's output does not have a clear interpretation. Often, this happens because the join is on columns that do not represent the main entities but some other property of these entities.
- Useful: These are the pairs where the output of the table has a clear interpretation.

[2]For reference, the final pairs we used along with their annotated labels can be found at https://github.com/arifusta/ogdpAnalysis

Table 3
Distribution of useful vs Accidental labels.

Portal	Join Result			
	accidental			useful
	U-Acc	R-Acc	total	
CA	35.95%	50.33%	86.28%	13.72%
UK	31.79%	49.01%	80.80%	19.20%
US	62.67%	24.00%	86.67%	13.33%

Table 3 shows the overall frequencies of the labels we gave across portals. As we hypothesized, overwhelming majority of the joinable pairs we sampled, whose columns had close-to-perfect value overlaps are accidental, i.e., false positives. The frequency ranges between 80.8% and 86.7% across portals (and 100% in SG). Our results indicate that value overlap alone can be a weak signal of useful joins and applications offering join feature need to be more selective in the tables they suggest to users.

In what follows, we list possible remediation strategies for avoiding U-Acc and R-Acc pairs, respectively:

- In order to avoid U-Acc pairs, tools must take the context of the tables forming the join into account as well instead of solely relying on column similarity, for which some metadata properties of datasets such as description or subject can be utilized, if available. Another alternative is to limit candidate joinable tables to be within the same dataset given the query table to ensure the same context, as exercised in a recent work [9].
- We argue that avoiding R-Acc pairs is what makes the problem of finding useful joinable pairs more challenging, which can be a future research direction to explore. Besides, uniqueness score of the columns and expansion ratio of a prospective join can be leveraged as heuristics. However, such heuristics standalone may not accurately predict usefulness for certain cases, since they do not address the problem of finding joins through identifying columns, which is yet another research direction that can be delved into.

3. Conclusion

We studied 4 OGDPs with the goal of informing researchers and developers that develop data systems over OGDPs about normalization properties of the tabular datasets and their impact in join operation in these portals. Our analysis reveals that tabular datasets published by OGDPs have unique characteristics that play role leading into false positive joins, which should be remedied by embracing more selective join candidates.

References

[1] E. Zhu, F. Nargesian, K. Q. Pu, R. J. Miller, Lsh ensemble: Internet-scale domain search, arXiv preprint arXiv:1603.07410 (2016).

[2] F. Nargesian, E. Zhu, K. Q. Pu, R. J. Miller, Table union search on open data, Proceedings of the VLDB Endowment 11 (2018) 813–825.

[3] E. Zhu, D. Deng, F. Nargesian, R. J. Miller, Josie: Overlap set similarity search for finding joinable tables in data lakes, in: Proceedings of the 2019 International Conference on Management of Data, 2019, pp. 847–864.

[4] A. Bogatu, A. A. Fernandes, N. W. Paton, N. Konstantinou, Dataset discovery in data lakes, in: 2020 IEEE 36th International Conference on Data Engineering (ICDE), IEEE, 2020, pp. 709–720.

[5] O. Benjelloun, S. Chen, N. Noy, Google dataset search by the numbers, in: The Semantic Web – ISWC 2020: 19th International Semantic Web Conference, Athens, Greece, November 2–6, 2020, Proceedings, Part II, Springer-Verlag, Berlin, Heidelberg, 2020, p. 667–682. URL: https://doi.org/10.1007/978-3-030-62466-8_41. doi:10.1007/978-3-030-62466-8_41.

[6] S. Castelo, R. Rampin, A. Santos, A. Bessa, F. Chirigati, J. Freire, Auctus: A dataset search engine for data discovery and augmentation, Proc. VLDB Endow. 14 (2021) 2791–2794. URL: https://doi.org/10.14778/3476311.3476346. doi:10.14778/3476311.3476346.

[7] A. Khatiwada, G. Fan, R. Shraga, Z. Chen, W. Gatterbauer, R. J. Miller, M. Riedewald, Santos: Relationship-based semantic table union search, Proc. ACM Manag. Data 1 (2023). URL: https://doi.org/10.1145/3588689. doi:10.1145/3588689.

[8] A. Khatiwada, R. Shraga, W. Gatterbauer, R. J. Miller, Integrating data lake tables, Proc. VLDB Endow. 16 (2022) 932–945. URL: https://doi.org/10.14778/3574245.3574274. doi:10.14778/3574245.3574274.

[9] C. Liu, A. Usta, J. Zhao, S. Salihoglu, Governor: Turning open government data portals into interactive databases, in: Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems, Association for Computing Machinery, New York, NY, USA, 2023, pp. 468–479.

[10] D. Brickley, M. Burgess, N. Noy, Google dataset search: Building a search engine for datasets in an open web ecosystem, in: The World Wide Web Conference, WWW '19, Association for Computing Machinery, New York, NY, USA, 2019, p. 1365–1375. URL: https://doi.org/10.1145/3308558.3313685. doi:10.1145/3308558.3313685.

[11] S. Neumaier, J. Umbrich, A. Polleres, Automated quality assessment of metadata across open data portals, J. Data and Information Quality 8 (2016). URL: https://doi.org/10.1145/2964909. doi:10.1145/2964909.

[12] J. Mitlöhner, S. Neumaier, J. Umbrich, A. Polleres, Characteristics of Open Data CSV Files, in: 2016 2nd International Conference on Open and Big Data (OBD), IEEE, 2016, pp. 72–79.

[13] Canada's open government data portal, https://open.canada.ca/en/open-data, 2023.

[14] Singapore's open government data portal, https://data.gov.sg/, 2023.

[15] Uk's open government data portal, https://www.data.gov.uk/, 2023.

[16] Usa's open government data portal, https://data.gov/, 2023.

[17] H. Garcia-Molina, J. Widom, J. D. Ullman, Database System Implementation, Prentice-Hall, Inc., USA, 1999.

[18] N. Novelli, R. Cicchetti, Fun: An efficient algorithm for mining functional and embedded dependencies, in: International Conference on Database Theory, Springer, 2001, pp. 189–203.

[19] D. Deng, A. Kim, S. Madden, M. Stonebraker, Silkmoth: An efficient method for finding related sets with maximum matching constraints, Proc. VLDB Endow. 10 (2017) 1082–1093. doi:10.14778/3115404.3115413.

[20] R. C. Fernandez, Z. Abedjan, F. Koko, G. Yuan, S. Madden, M. Stonebraker, Aurum: A data discovery system, in: 2018 IEEE 34th International Conference on Data Engineering (ICDE), IEEE, 2018, pp. 1001–1012.

[21] Y. Dong, K. Takeoka, C. Xiao, M. Oyamada, Efficient joinable table discovery in data lakes: A high-dimensional similarity-based approach, in: 2021 IEEE 37th International Conference on Data Engineering (ICDE), IEEE, 2021, pp. 456–467.

[22] C. Koutras, G. Siachamis, A. Ionescu, K. Psarakis, J. Brons, M. Fragkoulis, C. Lofi, A. Bonifati, A. Katsifodimos, Valentine: Evaluating matching techniques for dataset discovery, in: 2021 IEEE 37th International Conference on Data Engineering (ICDE), IEEE, 2021, pp. 468–479.