# Discovery of Semantic Non-Syntactic Joins

Marc Maynou[1,*], Sergi Nadal[1]

[1] *Universitat Politècnica de Catalunya, BarcelonaTech, Barcelona, Spain*

### Abstract

Data discovery is an essential step in the data integration pipeline involving finding datasets whose combined information provides relevant insights. Discovering joinable attributes requires assessing the closeness of the semantic concepts that two attributes represent, which is highly sensitive and dependent on the chosen similarity metric. The state of the art commonly approaches this task from a syntactic perspective, this is, performing comparisons based on the data values or on direct transformations (e.g., via hash functions). These approaches suffice when the two sets of instances share the same syntactic representation, but fail to detect cases in which the same semantic concept is represented by different sets of values, which we refer as *semantic non-syntactic* joins. This is a relevant problem in data lake scenarios, when the underlying datasets present high heterogeneity and lack of standardization. To that end, in this paper, we propose an empirical approach to detect semantic non-syntactic joins, which leverages, simultaneously, syntactic and semantic measurements of the data. We demonstrate that our approach is effective in detecting such kind of joins.

### Keywords

Data discovery, semantic similarity, syntactic similarity, profile comparison, distribution comparison,

## 1. Introduction

Data discovery is the exploratory task of navigating through numerous data sources to find relevant datasets for a given downstream task. With the advent of large-scale and highly heterogeneous repositories (e.g., data lakes [1], and open data repositories [2]), manual data discovery is an unfeasible task that demands automated and scalable methods [3]. In this paper, we address the problem of discovering joinable tables in a data lake. This is a problem that differs from the classical challenge of discovering *inclusion dependencies* in relational databases, which requires scalable and approximate methods [4], and has been the subject of extensive research [5].

The customary approaches for discovering joinable datasets are based on approximating or predicting metrics that quantify the degree of overlapping among sets of values (e.g., containment, Jaccard or cosine). Yet, a more challenging setting arises in the presence of syntactic or semantic ambiguity. Indeed, the recently coined *data lake disambiguation problem* [6], focuses on mapping *homographs* (i.e., data values that have the same representation but different meanings). Conversely, in this paper, we focus on the discovery of joinable tables with *synonyms* (i.e., when data values have different representations but have the same meaning). We refer to these cases as semantic non-syntactic joins (*sns*). Clearly, traditional syntactically-oriented methods fail to detect *sns* relationships, causing *pairs of attributes with shared semantics whose syntactic representation differs* to not be labelled as similar, which is a source of false negatives.

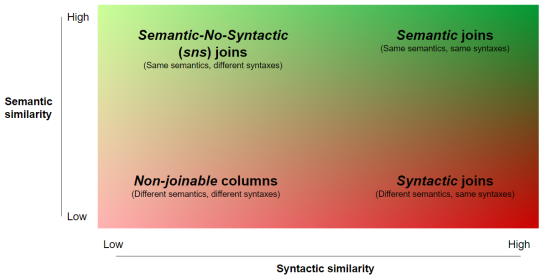

High
Semantic similarity
Low
*Semantic-No-Syntactic* (*sns*) joins
(Same semantics, different syntaxes)
*Semantic* joins
(Same semantics, same syntaxes)
*Non-joinable* columns
(Different semantics, different syntaxes)
*Syntactic* joins
(Different semantics, same syntaxes)
Low
High
Syntactic similarity

**Figure 1:** Join typology classification. Color indicates the semantic similarity, green being semantically similar concepts (i.e., interesting), and red uninteresting cases. The color intensity highlights the degree of syntactic similarity

One of the main limitations of the related work to discover *sns* joins is the usage of a binary label to determine joinability (i.e., joinable or non-joinable). This is commonly measured using a single syntactically-related metric, and, hence, *sns* joins tend to be misclassified as non-joinable columns, given that, in both cases, the syntactic similarity is low. To overcome this limitation, in this paper we advocate for a finer-grained distinction of joinability categories, taking into account **both syntactic and semantic similarities**. As shown in Figure 1, based on these two dimensions, besides *sns* joins, we can further introduce the following categories: *a)* **Semantic joins**: high degree of both syntactic and semantic similarity. That is, the same conceptual idea represented via the same values; *b)* **Syntactic joins**: high degree of

*DOLAP 2024: 26th International Workshop on Design, Optimization, Languages and Analytical Processing of Big Data, co-located with EDBT/ICDT 2024, March 25, 2024, Paestum, Italy*

[*] Corresponding author.

✉ marc.maynou@upc.edu (M. Maynou); sergi.nadal@upc.edu (S. Nadal)
iD 0009-0004-3908-4392 (M. Maynou); 0000-0002-8565-952X (S. Nadal)

syntactic similarity without a semantic relation. That is, the same set of values representing different semantic entities, which implies that, regardless of syntactic similarity, the join is not useful; and *c)* **Non-joinable pairs**: neither semantic nor syntactic relation.

The challenge of discovering *sns* joins has been studied from other perspectives, which can be classified in three major categories: entity resolution (ER) methods, embedding-based discovery, and comparison of statistics. In ER, the community has proposed methods for soft matching criteria based on fuzzy set matching or string similarity joins [7]. Embedding-based discovery utilizes high-dimensional vector representations of the data to capture their underlying semantics [8]. Methods based on the comparison of statistics rely on statistical properties of the data to capture semantic relationships. Yet, all such approaches combine all semantically-similar joins under a same label (i.e. joinable). Hence, their applicability for the *sns* detection problem is unknown, as their capacity to identify same-semantics, different-syntax joins is untested. Moreover, they rely on computationally expensive pairwise comparison, presenting prohibitive costs in large-scale environments.

To address the *sns* discovery problem, we propose a novel method for the discovery of *sns* joins. As depicted in Figure 1, we depart from the hypothesis that syntactic and semantic similarities can be measured separately to avoid misclassifying relevant *sns* pairs. To that end, we consider both set-based metrics (to determine syntactic similarity) and probability distributions (to determine semantic similarity). We study the validity of our hypothesis and experimentally show that our proposed method identifies *sns* joins with high accuracy. This new approach relies on descriptive metrics at the schema level, which ensures its scalability on large-scale scenarios.

## 2. Related work

A vast literature on discovering joinable datasets relies on value comparison to assess the similarity of two columns. Such syntactically-oriented approaches commonly use similarity metrics such as containment [9], Jaccard [10], or cosine [11]. As previously discussed, these methods are unable to detect those cases in which semantically similar columns present different instances of values (i.e., *sns* joins). We, hence, study related work on methods that present notions of similarity that do not leverage the intersection of values, at least not as a unique factor to determine the joinability of two columns. We classify them in three categories, which we review as follows.

**Entity Resolution methods.** Filtering is a technique in entity resolution that, after blocking, aims to identify all pairs of similar records to enable similarity joins. We refer the reader to [12] for an extensive survey on systems for large-scale entity resolution, which can be distinguished in learning-based methods [13], and non-learning ones [14, 15].

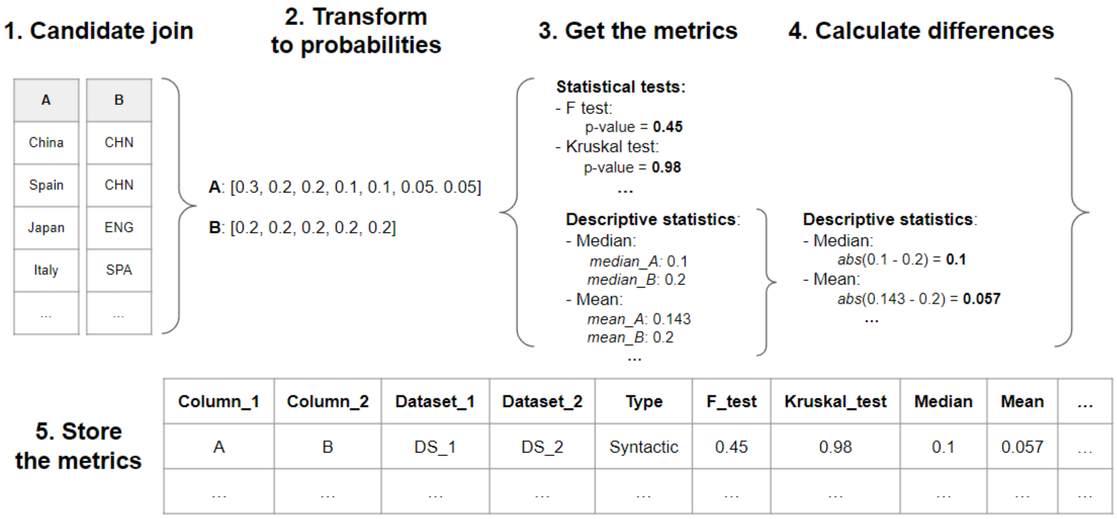
**Embedding-based discovery.** Embeddings are high-dimensional representations of values, an implicit method to capture the underlying semantics of the data. SEMPROP [16] uses embeddings on word names to find attributes in a data lake that respond to a given semantic type. A more complex implementation of this idea involves the computation of embeddings for every value of the attribute. PEXESO [17] directly defines the similarity of two columns as the proximity of the embeddings of the instances of both columns. WarpGate [18] incorporates several optimizations to the comparison of embeddings, such as the use of LSH indexes. DeepJoin [19] uses pre-trained models to generate the embeddings.

**Comparison of statistics.** Statistical procedures and measures are used to assess the similarity of two columns. Statistical properties of the data highlight the relationships that are hidden underneath the values that can not be detected by a pure syntactical comparison. Some examples are the comparison of distributions to create clusters of columns followed by the execution of syntactic-based filtering [20], leveraging big table corpora to look-up and detect correlations between attributes (SEMA-JOIN [21]) or executing post-statistical-analysis data transformations to produce the joins (Auto-Join [22]).

**Research gap.** The approaches above present non-syntactic measures of semantic similarity that could address the *sns* detection problem. However, they are limited specially in large-scale environments. Embedding-based systems require pairwise comparisons among sets of embeddings, the usage of statistics is mostly designed to operate within a table and Entity Resolution techniques present efficiency issues when handling large datasets [23]. Moreover, their lack a finer-grained categorization of same-semantics, different-syntax joins, so their applicability for the described task is untested.

## 3. Non-syntactic measures for semantic similarity

The starting hypothesis of this exploration is the following: *the semantic similarity of two columns can be defined by comparing their probability distributions* [20]. Yet, this hypothesis is meant to define a general trend in the behavior of column pairs and is hardly the case that two columns that share the same semantics are going to present *exactly* the same probability distribution. Therefore, a more general hypothesis needs to be stated: *two columns represent a similar semantic concept if their distributions resemble each other.* The opposite statement



| Column_1 | Column_2 | Dataset_1 | Dataset_2 | Type | F_test | Kruskal_test | Median | Mean | ... |
|---|---|---|---|---|---|---|---|---|---|
| A | B | DS_1 | DS_2 | Syntactic | 0.45 | 0.98 | 0.1 | 0.057 | ... |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |

**Figure 2:** Generation of the metrics

might be more intuitive: *two columns that do not present any kind of semantic relationship will likely have different distributions of values*. In order to assess whether these claims are valid or not, a fully-fledged experimentation needs to be conducted, as, oppositely to the set-intersection problems, the comparison of distributions has not been thoroughly explored as a method to assess the similarity of two columns, and less so for the detection of *sns* joins. Before, however, we will define how the comparison of distributions will be performed.

## 3.1. Comparing distributions

In [20] the selected algorithm to compare distributions is a modified version of the Earth Mover's Distance algorithm, which is difficult to employ and time-consuming to execute. A more direct and efficient approach, that still follows the same comparison-distribution principle, could be defined by employing the **usage of statistical tests to determine if the distributions are significantly different**. These are tools to mathematically assess whether two sets of data are significantly different from each other, leveraging certain statistical measures to do so, such as the mean, median or the standard deviation. This work focuses on non-numerical columns, given that assessing the semantic resemblance of two sets of numbers is significantly harder. This implies that the string-based values will be converted to sets of probabilities and ingested by the tests to determine if these same sets of probabilities are different, thus indicating whether the underlying distributions of the values align. If several of the tests are used and they all determine that the groups of probabilities are not different, then we can assume that the distribution is the same.

Nonetheless, the arbitrary nature of statistical tests implies that relying on them as the only predictor might generate too restrictive of an approach [24]. In order to rectify this issue, statistical tests can be combined with a more abstract measurement: the **comparison of metrics that describe general properties of the distributions**. This includes calculating the differences of several descriptive statistics obtained from the two distribution of the data, such as means, standard deviations, entropies, etc. This second procedure gives a higher-level intuition of the closeness of the sets of probabilities. Including an entire set of descriptive statistics about the distributions can present a less constraining approach that can generalize better in real-life scenarios.

Given the reasons stated, the comparison of distributions to assess semantic similarity will be performed by combining two types of evaluation metrics: statistical tests as a direct comparison and descriptive statistics as an indirect comparison. On paper, this is the desired compromise between correctly assessing the closeness of the sets of data while allowing some leniency to develop a more generalized approach. Figure 2 illustrates the process of generating the metrics.

## 3.2. Defining the model

In Section 3.1 we have defined a novel approach to measure semantic similarity, which will be implemented by the list of metrics defined in Table 1, following the proposed categories. Syntactic similarity will be defined following a similar, metric-based approach [4]. Our main objective is to ascertain whether a combined consider-

**Table 1**
Considered metrics

| Category | Sub-category | Metrics |
|---|---|---|
| Descriptive statistics | ”Basic” descriptive statistics | Mean, mode, median, standard deviation, min value, max value |
| | Distribution statistics | Skewness, entropy, kurtosis |
| | ”Advanced” descriptive statistics | Geometric mean, harmonic mean, variation, mean absolute deviation, inter-quantile range, standard error of the mean |
| | Confidence intervals | Mean of CI, min and max value of CI, standard deviation of CI |
| Statistical tests | Compare means | F test, Alexander-Govern test, T test, Kruskal test |
| | Compare distributions | Mann-Whitney test, Kolmogorov-Smirnov test, Cramer-Von Mises test, Anderson-Darling test |
| | Compare scale parameters | Ansari test, Mood test |
| | Compare variances | Bartlett test, Levene test, Fligner test |
| | Others | Wasserstein distance |

**Table 2**
Predictive models results

| Model | F1-score (macro) | Accuracy | F1-score (*sns*) | Recall (*sns*) | Precision (*sns*) |
|---|---|---|---|---|---|
| Combined metrics model | **90.39%** | **91.03%** | **88.08%** | **86.65%** | **90.04%** |
| Semantic metrics model | 77.16% | 77.51% | 76.03% | 82.84% | 70.34% |
| Syntactic metrics model | 81.61% | 83.98% | 73.95% | 68.50% | 81.44% |

ation of both semantic and syntactic similarities is able to correctly characterize *sns* joins. To that end, we have trained a classification model that employs boths sets of metrics with the goal of accurately isolating *sns* joins. In order to explore the behavior of the two sets of semantic and syntactic assessment metrics, three different models were developed: (i) only using semantic similarity assessment metrics, (ii) only using syntactic similarity assessment metrics and (iii) combining both groups.

Table 2, depicts the evaluation results of the classifier. We have used five different metrics to evaluate the models. The first two are the F1-score and the accuracy rate for the entire model, that is, taking into account the predictions for all labels. This highlights the potential of this predictive model in correctly classifying all join typologies. The three final metrics measure the behavior of the *sns* detection. First, we conclude that both sets of metrics perform considerably well on their own (74.54% and 73.95% in the *sns* F1-score), but combining the two groups dramatically improves the capabilities of the system (88.08% in the *sns* F1-score). The separated good behavior can be explained by Figure 1, as leveraging only semantic or syntactic aspects already separates *sns* joins from two other typologies of joins, whilst making it mostly indistinguishable to another category. This complementary behavior is supported by the inverse relationship between the recall and precision metrics. The semantic-metrics-only model detects more *sns* joins correctly, but has a higher tendency of classifying other typologies of joins as *sns*. On the other hand, the syntactic-metrics-only model presents more false positives but reduces the rate of false negatives. By combining the two approaches we retain and improve on the best characteristics of both methods. The results of the combined-metrics model seem to indicate that combining both types of metrics does provide the best environment for *sns* join detection, as theorized in the introduction of this work.

## 4. Conclusions and future work

We have proposed a new approach to data discovery that focuses on the detection of *sns* joins. This new methodology leverages, simultaneously, both syntactic and semantic similarity measurements, developing a more nuanced definition of similarity that could accurately characterize the semantic closeness of two sets of values without requiring the same value-representation. This work is a first step towards the definition of a model to identify *sns* joins, yet, since we have followed an empirical approach driven by labeled data gathered from external data lakes, further work is required to ensure its generalizability.

## Acknowledgments

Marc Maynou is supported by the EU’s Horizon Programme call, under Grant Agreements No. 101093164 (ExtremeXP), and Sergi Nadal is partially supported by the DOGO4ML project, funded by the Spanish Ministerio de Ciencia e Innovación under the funding scheme PID2020-117191RB-I00 / AEI / 10.13039/501100011033.

# References

[1] F. Nargesian, E. Zhu, R. J. Miller, K. Q. Pu, P. C. Arocena, Data lake management: Challenges and opportunities, Proc. VLDB Endow. 12 (2019) 1986–1989.

[2] R. J. Miller, F. Nargesian, E. Zhu, C. Christodoulakis, K. Q. Pu, P. Andritsos, Making open data transparent: Data discovery on open data, IEEE Data Eng. Bull. 41 (2018) 59–70.

[3] B. Golshan, A. Y. Halevy, G. A. Mihaila, W. Tan, Data integration: After the teenage years, in: PODS 2017, ACM, 2017, pp. 101–106.

[4] J. Flores, S. Nadal, O. Romero, Towards scalable data discovery, in: EDBT 2021, OpenProceedings.org, 2021, pp. 433–438.

[5] G. Fan, J. Wang, Y. Li, R. J. Miller, Table discovery in data lakes: State-of-the-art and future directions, in: SIGMOD 2023, Association for Computing Machinery, 2023, p. 69–75.

[6] A. Leventidis, L. D. Rocco, W. Gatterbauer, R. J. Miller, M. Riedewald, Domainnet: Homograph detection and understanding in data lake disambiguation, ACM Trans. Database Syst. 48 (2023) 9:1–9:40.

[7] G. Papadakis, D. Skoutas, E. Thanos, T. Palpanas, Blocking and filtering techniques for entity resolution: A survey, ACM Comput. Surv. 53 (2021) 31:1–31:42.

[8] T. Cong, M. Hulsebos, Z. Sun, P. Groth, H. V. Jagadish, Observatory: Characterizing embeddings of relational tables, CoRR abs/2310.07736 (2023).

[9] E. Zhu, D. Deng, F. Nargesian, R. J. Miller, JOSIE: overlap set similarity search for finding joinable tables in data lakes, in: SIGMOD 2019, ACM, 2019, pp. 847–864.

[10] R. C. Fernandez, Z. Abedjan, F. Koko, G. Yuan, S. Madden, M. Stonebraker, Aurum: A data discovery system, in: ICDE 2018, IEEE Computer Society, 2018, pp. 1001–1012.

[11] M. H. Franciscatto, M. D. D. Fabro, L. C. E. D. Bona, C. Trois, H. Tissot, Blending topic-based embeddings and cosine similarity for open data discovery, in: ICEIS 2022, SCITEPRESS, 2022, pp. 163–170.

[12] V. Christophides, V. Efthymiou, T. Palpanas, G. Papadakis, K. Stefanidis, An overview of end-to-end entity resolution for big data, ACM Comput. Surv. 53 (2021) 127:1–127:42.

[13] M. Kejriwal, D. P. Miranker, A two-step blocking scheme learner for scalable link discovery, in: Proceedings of the 9th International Workshop on Ontology Matching, volume 1317 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2014, pp. 49–60.

[14] J. Nin, V. Muntés-Mulero, N. Martínez-Bazan, J. L. Larriba-Pey, On the use of semantic blocking techniques for data cleansing and integration, in: IDEAS 2007, IEEE Computer Society, 2007, pp. 190–198.

[15] G. Papadakis, E. Ioannou, T. Palpanas, C. Niederée, W. Nejdl, A blocking framework for entity resolution in highly heterogeneous information spaces, IEEE Trans. Knowl. Data Eng. 25 (2013) 2665–2682.

[16] R. C. F. et al., Seeping semantics: Linking datasets using word embeddings for data discovery, in: ICDE 2018, IEEE, 2018, pp. 989–1000.

[17] Y. Dong, K. Takeoka, C. Xiao, M. Oyamada, Efficient joinable table discovery in data lakes: A high-dimensional similarity based approach, in: ICDE 2021, IEEE, 2021, p. 456–467.

[18] T. Cong, J. Gale, J. Frantz, H. V. Jagadish, Ç. Demiralp, Warpgate: A semantic join discovery system for cloud data warehouses, in: CIDR, www.cidrdb.org, 2023.

[19] Y. Dong, C. Xiao, T. Nozawa, M. Enomoto, M. Oyamada, Deepjoin: Joinable table discovery with pre-trained language models, in: Proc. VLDB Endowment 2023, Association for Computing Machinery, 2023, p. 2458–2470.

[20] M. Zhang, M. Hadjieleftheriou, B. C. Ooi, C. M. Procopiuc, D. Srivastava, Automatic discovery of attributes in relational databases, in: SIGMOD 2011, Association for Computing Machinery, 2011, p. 109–120.

[21] Y. He, K. Ganjam, X. Chu, Sema-join: joining semantically-related tables using big table corpora, in: Proc. VLDB Endowment 2015, VLDB, 2015, p. 1358–1369.

[22] E. Zhu, Y. He, , S. Chaudhuri, Auto-join: Joining tables by leveraging transformations, in: Proc. VLDB Endowment 2017, VLDB, 2017, p. 1034–1045.

[23] A. Zeakis, G. Papadakis, D. Skoutas, M. Koubarakis, Pre-trained embeddings for entity resolution: An experimental analysis, Proc. VLDB Endow. 16 (2023) 2225–2238.

[24] G. K. Kanji, 100 statistical tests, Sage, 1995.