Enhancing Biomedical Schema Matching with LLM-Based Training Data Generation

Yurong Liu, Aécio Santos, Eduardo H. M. Pena, Roque Lopez, Eden Wu, Juliana Freire
New York University
{yurong.liu,aecio.santos,eduardo.pena,rlopez,eden.wu,juliana.freire}@nyu.edu

Abstract

Data harmonization is the process of combining disparate datasets while ensuring that the information is compatible, consistent, and can be accurately compared. In this context, schema matching is an essential task that identifies correspondences between attributes coming from different data sources. In this paper, we empirically show that existing schema-matching methods are not effective at aligning complex schemas in the biomedical domain. We introduce a new approach for schema matching that leverages LLMs for (1) generating semantically coherent training data pairs that can be used to train effective column embedding models using the contrastive learning framework, and (2) refining final column match selections. This brings two important benefits: it overcomes the scarcity of in-domain and semantically diverse training data, which in turn enables the creation of simpler and cost-effective models for filtering candidate matches; and by making use of the broad knowledge of LLMs, it attains high accuracy in selecting the correct matches. We discuss the results of an experimental evaluation using real-world biomedical datasets which shows that our approach leads to significant improvements compared to existing state-of-the-art schema matching methods.

1 Introduction

In recent years, the growing volumes of biomedical data have substantially increased our ability to obtain data-driven insights. But harmonizing data from different sources to answer a research question or to create a predictive model is difficult [Adhikari et al., 2021]. Researchers often have to collect and integrate data from various sources, such as supplementary materials from related studies, and then resolve differences in their schemas, representations, data formats, units of measurement, terminologies. This ensures that the information can be accurately combined and used for analytics and machine learning to drive insights and inform decision-making. Harmonization is also applied to produce datasets that adhere to established standards, facilitating data sharing and interoperability.

Schema matching is a key task in data harmonization. It aims to determine correspondences between attributes of different schemas and ensure that data from various sources can be integrated despite differences in naming conventions or formats [Bellahsene et al., 2011]. Formally, the schema-matching problem can be defined as follows:

Definition 1 (Schema Matching) Let $S = \{s_1, \ldots, s_n\}$ be a source table and $G = \{g_1, \ldots, g_m\}$ be a global table, where $s_i \in S$ and $g_i \in G$ are attributes that define the table schemas. Schema matching consists of aligning the table schemas by establishing correspondences between attributes that represent the same real-world concept or entity.

Challenges in data harmonization. In typical "enterprise" schema matching, both the source S and global (or target) schema G are commonly structured as relational tables. However, G can also be a standard schema, a set of candidate column names with their associated descriptions, without any associated values—similar to ontologies. Standard schemas, such as the Genomic Data Commons

Table Representation Learning Workshop at NeurIPS 2024

(GDC) model [National Cancer Institute, 2024], can include hundreds of columns. Existing schema matching methods, discussed in Appendix B, do not scale effectively to large schemas and can yield low accuracy. These approaches struggle with diverse schema elements, ambiguous semantics, and cross-domain differences. Large Language Models (LLMs) can be used for schema matching [Tu et al., 2023]. Large models have shown to be effective in some scenarios, however they are resource-intensive. Smaller models, although more cost-effective, tend to have lower accuracy. This trade-off hinders their wide adoption. Finally, since no automated schema matching approach is foolproof, it is crucial to include the user in the loop to review the matching results. Therefore, solutions should incorporate ranking mechanisms or confidence scores to help users validate and refine matches, especially for ambiguous results.

Our solution and contributions. We propose *SSM* (Scalable Schema Matcher), a new approach to schema matching that addresses specific challenges in data harmonization, namely: the ability to handle large schemas and to create small (low-cost) yet accurate models. We introduce a novel contrastive learning framework to train an encoder that generates column embeddings. These embeddings capture the semantic distinctions between columns, enabling accurate schema matching, even in scenarios where the number of columns can reach the hundreds. Our solution incorporates an LLM for column match selection to refine matching results and improve accuracy. We also address the challenge of limited in-domain training data: we propose a data augmentation technique that leverages the capabilities of LLMs to generate semantically coherent positive pairs for training. We demonstrate the effectiveness of our method through an experimental evaluation using real-world biomedical datasets, curated by domain experts. Our results show that traditional methods attain poor performance for these datasets, and that our approach outperforms baseline models in both embedding similarity search and and schema matching tasks, highlighting its suitability for large-scale biomedical data harmonization.

2 The *SSM* Schema Matcher

Our approach tackles two key challenges discussed in Section 1: (1) achieving scalability and accuracy for schema matching, and (2) the lack of in-domain training data to train small retrieval models. It does so in two main stages. First, in an offline stage, we train a column embedding model (Section 2.1) to encode table columns as vectors. This allows us to efficiently retrieve semantically similar columns using vector similarity search in the second (online) stage. To overcome the lack of in-domain training data, we leverage a generalist LLM to generate synthetic data (Section 2.2) that can be used to train the embedding model. Finally, to improve the performance of the online matching phase, we introduce a final selection step based on an LLM that chooses the single best column out of the top k columns retrieved using the vector similarity search and our proposed embedding model (Section 2.3). In the remainder of this section, we describe each of these steps in detail.

2.1 Contrastive Learning Framework

We aim to generate column embeddings that spatially cluster similar schemas. This spatial arrangement facilitates the retrieval of semantically related schema representations through cosine similarity. The core of our training methodology is to minimize the distance between embeddings of identical or semantically related columns while maximizing the separation between those of distinct schemas.

To achieve this, we employ contrastive learning techniques to train a RoBERTa model which contains approximately 125 million parameters [Zhuang et al., 2021], specifically leveraging the framework established by the SimCLR algorithm from Chen et al. [2020]. The foundation of contrastive learning involves creating *positive pairs*—data points that should be close in the embedding space, typically representing identical or closely related columns. In contrast, *negative pairs*—those that should be distant in the embedding space—are formed from unrelated columns. Details on the tokenization and serialization processes used in our model are provided in the Appendix C.

When ground-truth matches are unavailable, a practical strategy to generate positive pairs, as explored by Fan et al. [2023], involves row shuffling or sampling within columns to synthesize data variations. However, this method falls short in generating semantically diverse pairs, thus limiting the efficacy of the learned embeddings for our specific application in semantic schema matching, especially in the biomedical domain. We propose to use a LLM to address this gap for more nuanced and context-aware data augmentation, as discussed in Section 2.2. This approach enhances our training dataset with synthetic yet plausible variations, better capturing the semantic nuances necessary for effective schema embedding.

2.2 Generating Positive Training Data Pairs

We employ an LLM for data augmentation. Specifically, we use GPT-4 [Achiam et al., 2023], which has demonstrated substantial capability in understanding and generating domain-specific knowledge across various fields [Bonifacio et al., 2022]. This allows us to generate semantically rich and diverse positive pairs. Given an original column x, we generate its augmented version x_i using GPT-4 with the prompt shown in Appendix D.

To enhance the accuracy and relevance of the synthetic data generated by GPT-4, we include additional context in the prompt based on the dataset. This context encompasses descriptions of columns and profiling statistics for numerical features, such as mean, median, and standard deviation. By providing these details, GPT-4 can generate synthetic matches that are not only structurally varied but also semantically coherent with the original data (see examples in Appendix E).

This approach enhances the semantic alignment of data points within the embedding space, which is critical for training robust models. The generated pairs capture a broader range of real-world variations and semantic nuances, thereby improving the generalization capability of the embedding. By leveraging the advanced capabilities of GPT-4, we move beyond simple augmentation techniques and achieve a higher level of data enrichment and variability in the training process.

2.3 Precise Match Selection

The contrastive learning approach allows us to efficiently retrieve a ranked list of the top potential relevant column matches. However, given that we use a small language model (RoBERTa) to generate the embeddings, the raking accuracy may be limited by its small number of parameters. To address this issue, we employ an LLM-based matcher to narrow these down to the most accurate single match.

To implement this idea, we introduce an LLM-based matcher that extends the methodology proposed by Feuer et al. [2023] (originally proposed for the task of a column type annotation) to our problem. To construct the prompt, we use as input the potential matches retrieved by the contrastive-learning method and request the model to select the match that best describes the source column which we want to find a match for. The full prompt is available in Appendix D. In addition to the top-k results from the contrastive learning model, we also provide additional context in the prompt by sampling rows from our source column The enhanced context is then used to prompt the LLM, which identifies and returns the most suitable match among the candidate columns.

Note that our contrastive learning model plays a crucial role in reducing the problem complexity for the LLM. The contrastive learning model simplifies the input fed into the LLM by pre-processing and condensing the data into a more manageable form, effectively addressing the well-known issue of limited context size. This enables the LLM to operate within its context constraints while benefiting from a rich, compressed dataset. In our experimental evaluation, we demonstrate that this approach not only reduces the prompt size but also significantly improves the accuracy of the LLM. By integrating the broad initial candidate generation capabilities of contrastive learning with the context-aware analysis afforded by LLMs, our method significantly improves accuracy in match selection by reducing the potential for errors from LLMs due to the smaller number of candidates.

3 Experimental Evaluation

Dataset. The data used in our evaluation was obtained from a study carried out by the Clinical Proteomic Tumor Analysis Consortium (CPTAC). It consists of source datasets produced by eight studies [Clark et al., 2019, Krug et al., 2020, Vasaikar et al., 2019, Wang et al., 2021, Huang et al., 2021, Satpathy et al., 2021, Cao et al., 2021, Dou et al., 2020], which were harmonized by Li et al. [2023]: each dataset was aligned with the (target) Genomics Data Commons (GDC) standard [National Cancer Institute, 2024]. The GDC standard consists of 730 data variables spanning various data types, such as categorical and numerical, and more complex types, such as proteins. The source datasets include demographic, diagnostic, and clinical variables for patients with proteogenomic tumor samples; each dataset contains over 50 columns. The gold data used to evaluate *SSM*, i.e., the set of matches between each dataset and the GDC, was manually created by biomedical researchers to ensure accuracy and reliability.

The schema matching process involves standardizing variable names and value types to align with GDC format specifications. Included variables are case submitter ID, age at diagnosis, gender, race, ethnicity, and various pathological specifics as shown in Appendix F. Matching is difficult due to the large number of columns and the nuance in the semantics of some of the biomedical data types.

Tables	Clark et al.	Satpathy et al.	Wang et al.	Cao et al.	Krug et al.	Huang et al.	Vasaikar et al.	Dou et al.
Coma	0.357	0.400	0.500	0.381	0.500	0.111	0.364	0.353
Similarity Flooding	0.286	0.500	0.611	0.381	0.500	0.167	0.455	0.294
Cupid	0.286	0.400	0.389	0.381	0.500	0.111	0.455	0.353
Distribution-based	0.143	0.000	0.111	0.000	0.000	0.056	0.000	0.059
Jaccard Distance	0.214	0.200	0.167	0.143	0.250	0.111	0.091	0.235
Vanilla RoBERTa	0.000	0.000	0.000	0.048	0.000	0.056	0.000	0.000
Starmie	0.286	0.400	0.333	0.238	0.250	0.167	0.091	0.294
Ours	0.429	0.500	0.611	0.381	0.500	0.278	0.636	0.294
Starmie + LLM	0.643	0.500	0.333	0.429	0.750	0.389	0.636	**0.647**
Ours + LLM	**0.786**	**0.800**	**0.778**	**0.762**	**1.000**	**0.556**	**0.909**	0.589

Table 1: Precision of the top-1 match results for baseline methods versus our approach, both with and without the integration of an LLM-based precise matcher. The results demonstrate that our method consistently outperforms the baseline approaches under both conditions.

Model	Contrastive Learning-based Matchers					+ LLM Precise Match Selection		
	R@3	R@5	R@10	R@20	R@50	P@1 (k=10)	P@1 (k=20)	P@1 (k=50)
Starmie	0.421	0.540	0.635	0.730	0.825	0.429	0.460	0.405
Ours	**0.722**	**0.833**	**0.897**	**0.944**	**0.992**	**0.802**	**0.619**	**0.722**

Table 2: Comparison of Recall at different cutoffs (R@k) for Starmie and our training data generation method on the CPTAC task. "Contrastive Learning-based Matchers" refers to using embeddings from a pre-trained language model fine-tuned using contrastive learning for similarity searches to retrieve the top-k entries. "+ LLM Precise Match Selection" involves employing the top-k entries retrieved by the language model to serve as input for an LLM-based top-1 matcher, as detailed in Section 2.3, comparing precision at top 1 match for label set sizes 10, 20, and 50.

Experiments. We implemented our method using two configurations: 1) a RoBERTa-base model with around 125 million parameters trained using contrastive learning employing LLM to generate positive training data, and 2) an enhanced version incorporating LLMs for precise matching ("Ours + LLM"). Table 1 shows the results of a comparison of our methods against popular schema matching algorithms including Coma [Do and Rahm, 2002], Similarity Flooding [Melnik et al., 2002], Cupid [Madhavan et al., 2001], Distribution-based [Zhang et al., 2011], Jaccard Distance, and a single-column version of Starmie [Fan et al., 2023] also using the RoBERTa-base model. Additionally, we evaluated a vanilla RoBERTa model without contrastive-learning based fine-tuning, as well as Starmie with the LLM precise matcher, to highlight our model's performance improvements using LLM for training data generation over simple generation techniques.

We employed precision at the top-1 match (Precision@1) and recall at multiple cutoff points (Recall@k for k = 3, 5, 10, 20, 50) as our primary metrics. These metrics were chosen to evaluate both the accuracy of the top match and the completeness of the returned matches up to the 50th rank.

The integration of LLMs into our schema matching process ("Ours + LLM") demonstrated superior match accuracy, achieving the highest top-1 precision in 7 out of 8 tables as presented in Table 1, highlighting the effectiveness of LLMs in deciphering complex schema relationships.

Table 2 shows the performance of our model compared to Starmie, both with and without LLM augmentation. Our LLM-enhanced approach showed significant improvements, which suggests a robust capability to capture a wider array of relevant matches. We also assessed the matching performance across different label set sizes. While our model maintained high recall at label sizes of 20 and 50 compared to label size of 10, the LLM performance declined, indicating challenges in achieving precise matches with larger label sets. This underscores the importance of using our model as a scope reduction tool that narrows down the label set to a smaller, more manageable size before integrating LLMs, thus optimizing the precision of the matching process.

4 Conclusion

In this work, we propose *SSM*, a schema matching approach that employs large language models (LLMs) and contrastive learning to address the challenges of schema matching in biomedical data harmonization. *SSM* outperforms traditional methods by leveraging LLMs to generate semantically rich training data and refine match selections. By creating a smaller specialized model, it provides a cost-effective solution that balances the trade-off of model size and accuracy. Future work will focus on extending our approach into a general foundation schema-matching framework applicable across various domains.

Acknowledgments We would like to thank DARPA for funding this work through the Automating Scientific Knowledge Extraction and Modeling (ASKEM) program, Agreement No. HR0011262087. This work was also partially supported by NSF awards IIS-2106888. The views, opinions, and findings expressed are those of the authors and should not be interpreted as representing the official views or policies of the Department of Defense, the U.S. Government, or NSF.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

K. Adhikari, S. B. Patten, A. B. Patel, S. Premji, S. Tough, N. Letourneau, G. Giesbrecht, and A. Metcalfe. Data harmonization and data pooling from cohort studies: a practical approach for data management. *International Journal of Population Data Science*, 6(1):1680, November 2021. doi: 10.23889/ijpds.v6i1.1680.

Zohra Bellahsene, Angela Bonifati, and Erhard Rahm. *Schema Matching and Mapping*. Springer Publishing Company, Incorporated, 1st edition, 2011. ISBN 9783642165177.

Luiz Bonifacio, Hugo Abonizio, Marzieh Fadaee, and Rodrigo Nogueira. Inpars: Unsupervised dataset generation for information retrieval. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2387–2392, 2022.

Liwei Cao, Chen Huang, Daniel Cui Zhou, Yingwei Hu, T Mamie Lih, Sara R Savage, Karsten Krug, David J Clark, Michael Schnaubelt, Lijun Chen, et al. Proteogenomic characterization of pancreatic ductal adenocarcinoma. *Cell*, 184(19):5031–5052, 2021.

Riccardo Cappuzzo, Paolo Papotti, and Saravanan Thirumuruganathan. Creating embeddings of heterogeneous relational datasets for data integration tasks. In *Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data*, pages 1335–1349, 2020.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.

David J Clark, Saravana M Dhanasekaran, Francesca Petralia, Jianbo Pan, Xiaoyu Song, Yingwei Hu, Felipe da Veiga Leprevost, Boris Reva, Tung-Shing M Lih, Hui-Yin Chang, et al. Integrated proteogenomic characterization of clear cell renal cell carcinoma. *Cell*, 179(4):964–983, 2019.

Tianji Cong, Fatemeh Nargesian, and HV Jagadish. Pylon: Semantic table union search in data lakes. *arXiv preprint arXiv:2301.04901*, 2023.

Hong-Hai Do and Erhard Rahm. Coma—a system for flexible combination of schema matching approaches. In *VLDB'02: Proceedings of the 28th International Conference on Very Large Databases*, pages 610–621. Elsevier, 2002.

Yongchao Dou, Emily A Kawaler, Daniel Cui Zhou, Marina A Gritsenko, Chen Huang, Lili Blumenberg, Alla Karpova, Vladislav A Petyuk, Sara R Savage, Shankha Satpathy, et al. Proteogenomic characterization of endometrial carcinoma. *Cell*, 180(4):729–748, 2020.

Grace Fan, Jin Wang, Yuliang Li, Dan Zhang, and Renée J Miller. Semantics-aware dataset discovery from data lakes with contextualized column-based representation learning. *Proceedings of the VLDB Endowment*, 16(7):1726–1739, 2023.

Benjamin Feuer, Yurong Liu, Chinmay Hegde, and Juliana Freire. Archetype: A novel framework for open-source column type annotation using large language models. *arXiv preprint arXiv:2310.18208*, 2023.

Chen Huang, Lijun Chen, Sara R Savage, Rodrigo Vargas Eguez, Yongchao Dou, Yize Li, Felipe da Veiga Leprevost, Eric J Jaehnig, Jonathan T Lei, Bo Wen, et al. Proteogenomic insights into the biology and treatment of hpv-negative head and neck squamous cell carcinoma. *Cancer cell*, 39 (3):361–379, 2021.

Aamod Khatiwada, Grace Fan, Roee Shraga, Zixuan Chen, Wolfgang Gatterbauer, Renée J Miller, and Mirek Riedewald. Santos: Relationship-based semantic table union search. *Proceedings of the ACM on Management of Data*, 1(1):1–25, 2023.

Christos Koutras, George Siachamis, Andra Ionescu, Kyriakos Psarakis, Jerry Brons, Marios Fragkoulis, Christoph Lofi, Angela Bonifati, and Asterios Katsifodimos. Valentine: Evaluating matching techniques for dataset discovery. In *2021 IEEE 37th International Conference on Data Engineering (ICDE)*, pages 468–479. IEEE, 2021.

Karsten Krug, Eric J Jaehnig, Shankha Satpathy, Lili Blumenberg, Alla Karpova, Meenakshi Anurag, George Miles, Philipp Mertins, Yifat Geffen, Lauren C Tang, et al. Proteogenomic landscape of breast cancer tumorigenesis and targeted therapy. *Cell*, 183(5):1436–1456, 2020.

Yize Li, Yongchao Dou, Felipe Da Veiga Leprevost, Yifat Geffen, Anna P Calinawan, François Aguet, Yo Akiyama, Shankara Anand, Chet Birger, Song Cao, et al. Proteogenomic data and resources for pan-cancer analysis. *Cancer cell*, 41(8):1397–1406, 2023.

Jayant Madhavan, Philip A Bernstein, and Erhard Rahm. Generic schema matching with cupid. In *vldb*, volume 1, pages 49–58, 2001.

Sergey Melnik, Hector Garcia-Molina, and Erhard Rahm. Similarity flooding: A versatile graph matching algorithm and its application to schema matching. In *Proceedings 18th international conference on data engineering*, pages 117–128. IEEE, 2002.

National Cancer Institute. Gdc data model. `https://gdc.cancer.gov/developers/gdc-data-model`, 2024. Accessed: 2024-09-20.

Shankha Satpathy, Karsten Krug, Pierre M Jean Beltran, Sara R Savage, Francesca Petralia, Chandan Kumar-Sinha, Yongchao Dou, Boris Reva, M Harry Kane, Shayan C Avanessian, et al. A proteogenomic portrait of lung squamous cell carcinoma. *Cell*, 184(16):4348–4371, 2021.

Roee Shraga, Avigdor Gal, and Haggai Roitman. Adnev: cross-domain schema matching using deep similarity matrix adjustment and evaluation. *Proc. VLDB Endow.*, 13(9):1401–1415, may 2020. ISSN 2150-8097. doi: 10.14778/3397230.3397237. URL `https://doi.org/10.14778/3397230.3397237`.

Jianhong Tu, Ju Fan, Nan Tang, Peng Wang, Guoliang Li, Xiaoyong Du, Xiaofeng Jia, and Song Gao. Unicorn: A unified multi-tasking model for supporting matching tasks in data integration. *Proceedings of the ACM on Management of Data*, 1(1):1–26, 2023.

Suhas Vasaikar, Chen Huang, Xiaojing Wang, Vladislav A Petyuk, Sara R Savage, Bo Wen, Yongchao Dou, Yun Zhang, Zhiao Shi, Osama A Arshad, et al. Proteogenomic analysis of human colon cancer reveals new therapeutic opportunities. *Cell*, 177(4):1035–1049, 2019.

Liang-Bo Wang, Alla Karpova, Marina A Gritsenko, Jennifer E Kyle, Song Cao, Yize Li, Dmitry Rykunov, Antonio Colaprico, Joseph H Rothstein, Runyu Hong, et al. Proteogenomic and metabolomic characterization of human glioblastoma. *Cancer cell*, 39(4):509–528, 2021.

Meihui Zhang, Marios Hadjieleftheriou, Beng Chin Ooi, Cecilia M Procopiuc, and Divesh Srivastava. Automatic discovery of attributes in relational databases. In *Proceedings of the 2011 ACM SIGMOD International Conference on Management of data*, pages 109–120, 2011.

Liu Zhuang, Lin Wayne, Shi Ya, and Zhao Jun. A robustly optimized BERT pre-training approach with post-training. In Sheng Li, Maosong Sun, Yang Liu, Hua Wu, Kang Liu, Wanxiang Che, Shizhu He, and Gaoqi Rao, editors, *Proceedings of the 20th Chinese National Conference on Computational Linguistics*, pages 1218–1227, Huhhot, China, August 2021. Chinese Information Processing Society of China. URL `https://aclanthology.org/2021.ccl-1.108`.

Appendix

A Formal Statement of Schema Matching Algorithms

A matching algorithm (or matcher) aims to identify pairs (s_i, g_j) where $s_i \in S$ and $g_j \in G$ likely represent the same attribute. Thus, a matcher $\mathcal{M}$ can be seen as a function that generates a schema mapping $M \subseteq S \times G$ where each element $(s, g) \in M$ represents a correspondence between a source attribute s and a global attribute g.

B Prior Works on Schema Matching

Schema matching algorithms have traditionally relied on functions that compute the similarity between attribute pairs (s, g). This similarity can be determined by analyzing various characteristics, including attribute names, data types, values, and domain constraints. Early approaches relied on basic string-based metrics, such as Jaccard similarity between attribute names and values, to find exact matches between schema elements [Madhavan et al., 2001, Melnik et al., 2002]. The COMA algorithm improved upon these methods by employing a composite matching strategy, combining multiple features (e.g., name, structure, and data type) to provide a more comprehensive similarity assessment [Do and Rahm, 2002]. More recently, Koutras et al. [2021] introduced an extensible experimentation suite for evaluating different schema-matching algorithms, integrating these early techniques. These approaches often struggle to capture complex relationships and deeper semantics within data sets, resulting in poor accuracy when identifying semantically valid matches [Khatiwada et al., 2023]. To address this limitation, approaches using supervised learning have been proposed [Shraga et al., 2020] and language representation techniques by using fine-tuned language models to generate contextual embeddings that represent schema information [Cappuzzo et al., 2020, Khatiwada et al., 2023, Fan et al., 2023, Tu et al., 2023, Cong et al., 2023].

C Tokenization and Serialization Process

To facilitate the contrastive learning approach, columns are serialized in a format that enhances their discriminative features within the embedding space. This process is crucial for the success of the model when distinguishing between similar and dissimilar column pairs.

Following the standard tokenization process for the contrastive learning model, given a column $C = \{v_1, \ldots, v_m\}$ with header C_h, we serialize C as follows:

$$\text{serialize}(C) = \text{[CLS]} \oplus \text{Encode}(C_h) \oplus \text{[SEP]} \oplus \bigoplus_{i=1}^{m} \text{Encode}(v_i) \oplus \text{[SEP]},$$

This serialization ensures that both the header and values within the column are adequately represented and distinguishable within the model's embedding space.

D Prompts

The full prompt used as context to the LLM for augmenting a given column is:

```
Given a table with the header {column_name} and its values
{column_values}, use your expertise to identify one alternative
name for this column as found in other datasets.  Ensure this name
follows common database formatting conventions like underscores
and abbreviations.  Also, provide distinct possible synonyms or
alternative forms for the values that are technically correct.
Output in format:  "alternative_name, value1, value2, value3, ..."
Do not include any other information or use quotes in your response.
```

For the Precise Match Selection phase, the LLM uses the following prompt (where $top\_k = 1$):

```
You are an assistant for column matching. Please select the top
{top_k} class from {labels} which best describes the context. The
context is defined by the column name followed by its respective
values. Please respond only with the name of the classes separated
by semicolon. CONTEXT: {context}
```

E Training Data Generation Examples

See below some examples of the provided data to GPT-4 and the generated data.

Original Data	**Generated Data**
Column: country_of_residence_at_enrollment **Values:** *France, Germany, etc.*	**Column:** enrollment_country **Values:** *FR, GE, etc.*
Column: tissue_source_sites **Values:** *Thyroid, Ovary, etc.*	**Column:** tumor_site **Values:** *Thyroidal, Ovarian, etc.*
Column: exon **Values:** *exon11, exon15, etc.*	**Column:** gene_segments **Values:** *segment11, segment15, etc.*
Column: masked_somatic_mutations **Values:** *MET_D1010N, FLT3_ITD, etc.*	**Column:** genetic_variants **Values:** *D1010N_MET, ITD_FLT3, etc.*
Column: max_tumor_bulk_site **Values:** *Maxilla, Splenic lymph nodes, etc.*	**Column:** primary_tumor_location **Values:** *Maxillary, Splenic_nodes, etc.*

F Schema Matching Ground Truth Examples

See below for example columns from Li et al. [2023] and corresponding matches from the GDC Schema.

Raw Data	**GDC-formatted Data**
Column: Tumor_Site **Values:** *Lower pole, Upper pole, etc.*	**Column:** site_of_resection_or_biopsy **Values:** *Supraglottis, Thymus, etc*
Column: tumor_reoccur_after_treatment **Values:** *1.0 or 0.0*	**Column:** progression_or_recurrence **Values:** *Yes, No, or Not Reported*
Column: Histologic_Grade_FIGO **Values:** *FIGO grade 1, FIGO grade 2, etc*	**Column:** tumor_grade **Values:** *G1, G2, etc*
Column: Path_Stage_Reg_Lymph_Nodes-pN **Values:** *pN0, pN1, etc*	**Column:** ajcc_pathologic_n **Values:** *N0, N1, etc*
Column: Path_Stage_Primary_Tumor-pT **Values:** *pT1a (FIGO IA), pT3b (FIGO IIIB), etc*	**Column:** ajcc_pathologic_t **Values:** *T1a, T3b, etc*