SIN-CHN
SCIENTIFIC PRESS PTE.LTD.

AI Insights 2025, 1(1), 1929.
https://doi.org/10.62617/aii1929

Article

SSN filtering method with pre-trained models for entity matching in data washing machine

Bushra Sajid[1,*], Ahmed Abu-Halimeh[2], John R. Talburt[2]

[1] Department of Computer Science, The University of Arkansas at Little Rock, AR 72204, USA
[2] Department of Information Science, The University of Arkansas at Little Rock, AR 72204, USA
*** Corresponding author:** Bushra Sajid, bxsajid@ualr.edu

CITATION

Sajid B, Abu-Halimeh A, Talburt JR. SSN filtering method with pre-trained models for entity matching in data washing machine. AI Insights. 2025; 1(1): 1929. https://doi.org/10.62617/aii1929

ARTICLE INFO

Received: 7 November 2024
Accepted: 17 March 2025
Available online: 25 March 2025

Abstract: Entity Resolution (ER) is a vital process in data integration and quality improvement, aimed at identifying and linking records that refer to the same real-world entity. As data volumes and diversity grow, traditional ER methods face challenges such as scalability, poor data quality, and difficulties in handling sparse or inconsistent records. To address these limitations, this research introduces the Proof-of-Concept Data Washing Machine (DWM), developed under the National Science Foundation, Data Analytics that are Robust and Trusted (NSF DART) Data Life Cycle and Curation research theme, which automates the detection and correction of data quality errors through unsupervised entity resolution. The study focuses on advancing ER by replacing traditional rule-based approaches with machine learning (ML) and deep learning techniques, particularly for the linking process. Deep learning models like Bidirectional Encoder Representations from Transformers (BERT) and its variants are employed to enhance similarity scoring within Cluster ER methods. By integrating these models into the DWM framework, the research leverages attention mechanisms to generate reference embeddings and compute similarity score vectors. Additionally, it addresses optimization in candidate pair reduction during the ER blocking process to improve efficiency. A novel method for managing sensitive data, such as Social Security Numbers (SSNs), is proposed to streamline pair reduction in the linking stage. Comparative analysis between Linking_with_ML and SSN_Filtering_with_ML methods across diverse file types reveals that SSN_Filtering_with_ML achieves higher precision while maintaining a balanced trade-off between precision and recall. These findings highlight its robustness and accuracy in entity matching, significantly enhancing the DWM's capacity for accurate record linkage while reducing unnecessary comparisons. This research contributes to advancing data quality practices, enabling better decision-making across organizations by providing scalable and efficient solutions for complex entity resolution challenges.

Keywords: data quality; machine learning; entity resolution; filtering method

1. Introduction

Artificial Intelligence (AI) in entity resolution (ER) represents a pivotal area of research, particularly as organizations face the challenges posed by large datasets of varying quality. The ability of AI to enhance entity resolution processes is crucial for organizations seeking to extract meaningful insights from their data, thereby facilitating accurate responses to fundamental business inquiries, such as assessing customer counts. Entity resolution is fundamentally concerned with identifying and linking records that correspond to the same real-world entities across diverse data sources. As the reliance on data-driven decision-making grows, the need for precise and efficient ER techniques has become increasingly important [1]. This study

addresses a significant gap in current methodologies by investigating a novel approach that incorporates Social Security Number (SSN) filtering within the framework of a Data Washing Machine (DWM).

The exponential increase in data volume and complexity has necessitated the development of more sophisticated ER techniques. Traditional methods, such as indexed entity matching and similarity evaluations based on predetermined dimensions, have established a strong foundation due to their significance in ensuring data accuracy [2]. Recent advancements in the field have introduced deep learning approaches that promise enhanced accuracy and processing speed when managing large, heterogeneous datasets. By integrating SSN filtering into the DWM architecture, this research aims to create a more efficient and scalable ER solution that not only maintains high accuracy but also minimizes computational resource usage. This study builds on existing advancements by emphasizing SSN filtering as a critical component of the ER process, effectively addressing one of its primary challenges: the quadratic time complexity associated with comparing every possible pair of records [3].

To explore this integration further, our research evaluates pre-trained models, particularly BERT and its variants, alongside newer large language models to assess their potential in enhancing the entity resolution process within the DWM. The application of attention mechanisms aids in deriving reference embeddings based on similarity score vectors essential for comparing entity records. Additionally, machine learning techniques are employed to benchmark results against scores generated by the DWM. By incorporating a deep learning model into an unsupervised DWM process, we aim to enhance clustering accuracy while addressing both syntactic and semantic similarity issues. However, initial implementations of the pre-trained models revealed significant slowdowns, necessitating an alternative approach capable of mitigating these performance issues.

This study extends the capabilities of the DWM by integrating SSN filtering techniques alongside machine learning pre-trained models to facilitate the linking process within the DWM. By addressing potential challenges such as false negatives arising from inaccurately recorded or missing SSNs, this integration seeks to enhance the efficiency of the hybrid linking process. Through demonstrating the value of incorporating SSN filtering within the DWM framework, we aim to contribute to the ongoing evolution of ER methodologies. Our approach directly addresses practical challenges faced in applying ER techniques to large-scale, real-world data applications while improving both precision and effectiveness in entity matching. We ensure compatibility and ease of implementation by making only minor adjustments to align with existing DWM procedures.

The implications of this research are significant across various sectors, including fraud detection, client profile management, and data integration. As we delve deeper into our methodology and results, we aim to provide a comprehensive understanding of how SSN filtering methods combined with machine learning techniques and the DWM framework can transform entity resolution practices.

The remainder of this paper is structured as follows: The related work section reviews recent advancements in filtering methods and performance issues associated with entity resolution while also addressing DWM ER. The Method section details our proposed methodology, illustrating how SSN filtering has improved our hybrid

model's performance and accelerated DWM operations. It further describes our experimental setup and presents findings from applying our method across datasets with varying levels of data quality. The discussion section compares our new approach's performance against previous machine learning methods used for linking processes within the DWM. Finally, we conclude with a summary of contributions and potential future research directions.

2. Related work

In data management, entity matching is a crucial procedure that identifies entries from different data sources referring to the same real-world entities. Recent advancements in this area have significantly improved both accuracy and efficiency by leveraging deep learning alongside traditional methods. These methods often involve consulting a standard name library for potential entity names, calculating similarities using predetermined dimensions, and selecting the most similar candidate as the canonical entity name [1]. Recent research indicates that deep learning approaches consistently outperform traditional ER techniques, particularly in complex scenarios [4,5]. These models utilize sophisticated algorithms to enhance entity matching accuracy; however, they require careful generation of training and assessment datasets to ensure reliable outcomes.

Systems designed for entity matching typically index entities and initiate searches for matches based on these indexed points, thereby facilitating efficient matching processes [6]. Emerging studies emphasize the role of neural networks in entity matching, providing a taxonomy of techniques that automate similarity assessments and enhance processing speed compared to traditional approaches [7]. For instance, the EMAN algorithm exemplifies an unsupervised method that employs locality-sensitive hashing to efficiently match entities across heterogeneous data sources, achieving high accuracy without reliance on ground-truth data. While traditional methods have established a foundational framework for entity matching, the integration of deep learning techniques offers a promising pathway for addressing existing challenges, particularly in managing large and diverse datasets.

Entity resolution (ER) through Social Security Number (SSN) filtering is a critical technique for efficiently identifying records that refer to the same real-world entity. SSN filtering serves as an effective mechanism to narrow down the search space, thereby enhancing the performance of ER systems. The primary features of this strategy are elaborated upon in the subsequent sections.

2.1. Filtering techniques

Entity resolution techniques employing SSN filtering utilize several sophisticated methods to enhance efficiency and accuracy. Blocking workflows, as described by Papadakis et al. [8], group entity profiles with identical or similar SSNs, significantly reducing the number of comparisons needed. This approach acts as a preliminary filter, creating manageable subsets of data for more detailed analysis. Complementing this, string similarity joins quickly identify records with SSNs that exceed a certain similarity threshold, further narrowing down potential matches [1]. This method is beneficial for handling slight variations or errors in SSN entries. Filtering techniques

in entity resolution (ER) play a critical role in managing computational complexity while integrating multidimensional data sources. These methods enable analysts to prioritize high-quality information and reconcile discrepancies across heterogeneous datasets, aligning with the integration of Information Quality (IQ) metrics, including subjective dimensions (SIQ), into unified analytical frameworks [9]. Furthermore, as described by Papadakis et al. [7], nearest-neighbor techniques efficiently identify the closest matches based on SSN closeness by converting entity profiles into vectors. This vectorization technique captures tiny commonalities that more strict matching criteria could overlook, enabling nuanced comparisons. By combining or applying these strategies one after the other, a strong SSN filtering pipeline can be created.

For instance, a typical workflow might start with blocking workflows as an initial step, followed by string similarity joins for refined matching, and finally, nearest-neighbor methods for handling edge cases or particularly complex matches. In addition to improving accuracy, this sequential application of the approaches boosts entity resolution speed by considering many 129 characteristics of similarity in SSN data.

2.2. Performance considerations

SSN filtering significantly reduces the quadratic time complexity associated with traditional Entity Resolution (ER) algorithms, making it a valuable technique for managing large datasets. By leveraging the unique characteristics of Social Security Numbers (SSNs), this method effectively clusters potential matches, thereby minimizing the number of comparisons required. However, while SSN filtering enhances efficiency, it is imperative to balance speed and accuracy; missed matches due to improper filtering can adversely affect the overall quality of the ER process.

The effectiveness of SSN filtering stems from its capacity to create initial groupings based on exact or near-exact SSN matches, which can subsequently be refined using additional attributes. This approach not only accelerates the matching process but also reduces false positives by establishing a robust initial criterion for potential matches. Nonetheless, relying solely on SSN filtering may prove insufficient for comprehensive ER, particularly in datasets where SSN information is incomplete or inaccurate. The potential drawbacks of SSN-only filtering necessitate a hybrid strategy that incorporates additional matching criteria and machine learning techniques to maintain high precision and recall rates. Furthermore, there exists a risk of misleading negative results when SSNs are missing, incorrectly entered, or intentionally altered. These limitations underscore the need for supplementary methods alongside SSN filtering to ensure a thorough and accurate entity resolution process. Additionally, employing probabilistic matching techniques can bolster the robustness of the ER process by addressing possible errors or variations in SSN data.

Continuous Benchmark of Filtering Methods for Entity Resolution [1]: This repository provides code and data for benchmarking various filtering methods used in entity resolution. It highlights how these methods reduce computational costs by focusing on candidate pairs likely to match. While it emphasizes the importance of filtering techniques in improving performance, it does not specifically address the integration of SSN filtering or its implications for accuracy in real-world applications.

An Overview of End-to-End Entity Resolution for Big Data [10]: This paper provides insights into end-to-end ER processes and discusses dynamic indexing/blocking methods to enhance performance. While it mentions various strategies for improving efficiency, it does not specifically address how SSN filtering can be integrated into these workflows or its potential privacy implications.

Benchmarking Filtering Techniques for Entity Resolution [7]: This research presents experimental results demonstrating state-of-the-art performance across different datasets using various filtering techniques. However, it does not delve into the unique aspects of SSN filtering or how it compares to other methods in terms of accuracy and privacy concerns. Sarker et al. [11] builds the approach entity resolution by integrating advanced neural architectures and feature extraction techniques to enhance the alignment and retrieval of entities represented in both text and images. This multimodal strategy aims to improve the accuracy and robustness of entity resolution tasks.

Blocking and Filtering Techniques for Entity Resolution: A Survey [12]: This survey covers a range of blocking and filtering techniques but does not specifically highlight the role of SSNs in enhancing entity resolution processes or discuss the trade-offs involved with their use.

Building Upon Existing Research:

The current research builds upon these studies by specifically focusing on integrating SSN filtering within a hybrid model that combines traditional methods with machine learning approaches. Unlike previous works that may overlook the nuances of privacy concerns associated with SSNs, this study aims to address these issues by proposing a balanced approach that incorporates additional matching criteria alongside SSN filtering.

Moreover, while existing literature emphasizes performance metrics and computational efficiency, our research aims to provide a comprehensive framework that ensures both speed and accuracy in entity resolution processes. By highlighting the importance of addressing potential pitfalls associated with incomplete or erroneous SSNs, we differentiate our work from prior studies that may not fully account for these practical challenges.

In summary, our study seeks to fill critical gaps left unaddressed by previous research by providing a detailed exploration of how SSN filtering can be effectively integrated into broader ER methodologies while maintaining high standards of accuracy and privacy compliance.

2.3. DWM ER

The Data Washing Machine helps cover all the entity resolution steps by going through different processes, which are rule-based methods to solve error correction problems and improve data quality. The DWM (**Figure 1**) uses ER as the first step using unsupervised blocking and stop word schemes 159 based on token frequency. A variant of the Monge-Elkan comparator, a scoring matrix, is used to link unstandardized references in Al Sarkhi and Talburt's research [13,14] for a matrix 161 comparator for linking equivalent references. The scoring matrix comparator underwent further 162 developments, which improved its capabilities and allowed it to do linking using conventional 163 techniques. The DWM ER process is iterative,

and the reference similarity threshold is increased by 164 in each iteration. The prototype was tested on 18 fully annotated test samples of primarily 165 synthetic person data, which varied in two ways: good data quality versus poor data quality and a single record layout versus two different record layouts. The results demonstrated the feasibility of building an unsupervised ER engine to support data integration for good quality references while avoiding the time and effort to standardize reference sources to a standard layout and to design and test matching rules, design blocking keys, or test blocking alignment.

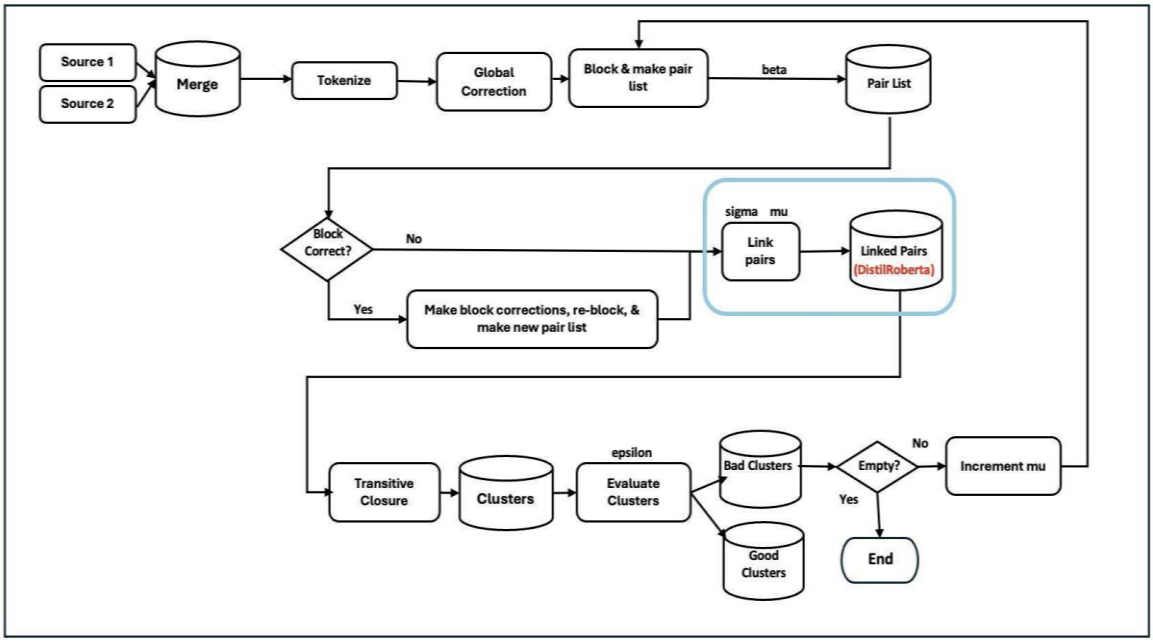

Figure 1. Flowchart of the data washing machine.

To make the DWM move along with the current advancement in the technology, there was a need to assess how the pretrained machine learning model would help with the linking process. By integrating these machine learning models into the unsupervised DWM process, the research aims to improve the clustering accuracy by addressing syntactic and semantic similarity issues, which can be done using advanced techniques.

3. Problem statement

As we wanted to use the pertained model for the linking process in the data washing machine to create the hybrid model, the drawback we were facing was the performance; it was taking several hours to run each data file, especially poor data files because they had more data quality issues, like spelling errors, null values for the SSN's attributes, and deduplication. Matching is complex when references are heterogeneously structured and have low data quality. **Table 1** provides a specific instance of unstructured references. But in a traditional method where a scoring matrix was used while running DWM, taking less time to run overall. To have a scalable AI method for the DWM to have good accuracy, speed, and computational resources. So, even though previous methods, which include machine learning [15], give us closer

results to the traditional method [16,17] we still needed to make the new method efficient to get the desired performance.

4. Dataset

There are 18 sample data files with more than 10,000 samples to test our model. These datasets all have associated truth sets (annotations) that allow the user to check the accuracy of the clustering for a given set of parameter settings. Each dataset came with annotated truth sets, as depicted in **Table 1**, allowing for the validation of clustering accuracy under distinct parameter configurations.

Table 1. Datasets used for data washing machine.

File Name	Size	Characteristics	Quality	Layout	Truth File Name
S1G.txt	50	Person name & address	Good	Single	truthABCgoodDQ.txt
S2G.txt	100	Person name & address	Good	Single	truthABCgoodDQ.txt
S3Rest.txt	868	Business name & address	Good	Single	truthRestaurant.txt
S4G.txt	1912	Person name & address	Good	Single	truthABCgoodDQ.txt
S5G.txt	3004	Person name & address	Good	Single	truthABCgoodDQ.txt
S6GeCo.txt	19,998	Person name & address	Good	Single	truthGeCo.txt
S7GX.txt	2912	Person name & address	Good	Mixed	truthABCgoodDQ.txt
S8P.txt	1000	Person name & address	Poor	Single	truthABCpoorDQ.txt
S9P.txt	1000	Person name & address	Poor	Single	truthABCpoorDQ.txt
S10PX.txt	2000	Person name & address	Poor	Mixed	truthABCpoorDQ.txt
S11PX.txt	3999	Person name & address	Poor	Mixed	truthABCpoorDQ.txt
S12PX.txt	6000	Person name & address	Poor	Mixed	truthABCpoorDQ.txt
S13GX.txt	2000	Person name & address	Good	Mixed	truthABCgoodDQ.txt
S14GX.txt	5000	Person name & address	Good	Mixed	truthABCgoodDQ.txt
S15GX.txt	10,000	Person name & address	Good	Mixed	truthABCgoodDQ.txt
S16PX.txt	2000	Person name & address	Poor	Mixed	truthABCpoorDQ.txt
S17PX.txt	5000	Person name & address	Poor	Mixed	truthABCpoorDQ.txt
S18PX.txt	10,000	Person name & address	Poor	Mixed	truthABCpoorDQ.txt

Table 2. Reference examples in dataset file (S1G.txt).

Reference ID	Reference
C787384	IAN AADLAND, LARS, 7715 ABINGTON DR, KERNERSVILLE, NC 27284, (361)-924-5829,1911/8/25
B996789	IAAN LARS AADLAND, 7715, ABINGTON DR, KERNERSVILLE, NC, 27284,490-46-2048,1911825
C787387	AANAI, HIKARI, F, 2165 MAURINE WAY, WINSTON SALEM, NC 27127, (483)-549-7645,
C787385	Kavassana Aanai, Hikari, F, 2165 MAURINE WAY, WINSTON SALEM, NC 27127, (483)-549-7645, 1906/4/6

Each data file took several hours to process, particularly those with poor data quality, as they contained more issues such as spelling errors, missing values in SSN attributes, and duplicate records. The data to train our model is synthetic data that

mimics real-world people's references. The following **Table 2** is an example of the dataset.

5. Method

After careful consideration, we decided to focus on the attribute that can help us to better match, and there could be none better than the SSN. SSNs is only one important attribute in the record to have unique information about the person's information. Since we need to make the performance better as well, in this method, we added a new module in the DWM for the SSN's filtering to make it work in our favor [18]. The records that have the same SSNs would be separated into a group as shown in **Figure 2** and saved in the form that matches the format of the DWM; this way, it helps to filter the records and does the maximum work itself at this point. Then we take that output and take the records that don't have the SSNs.

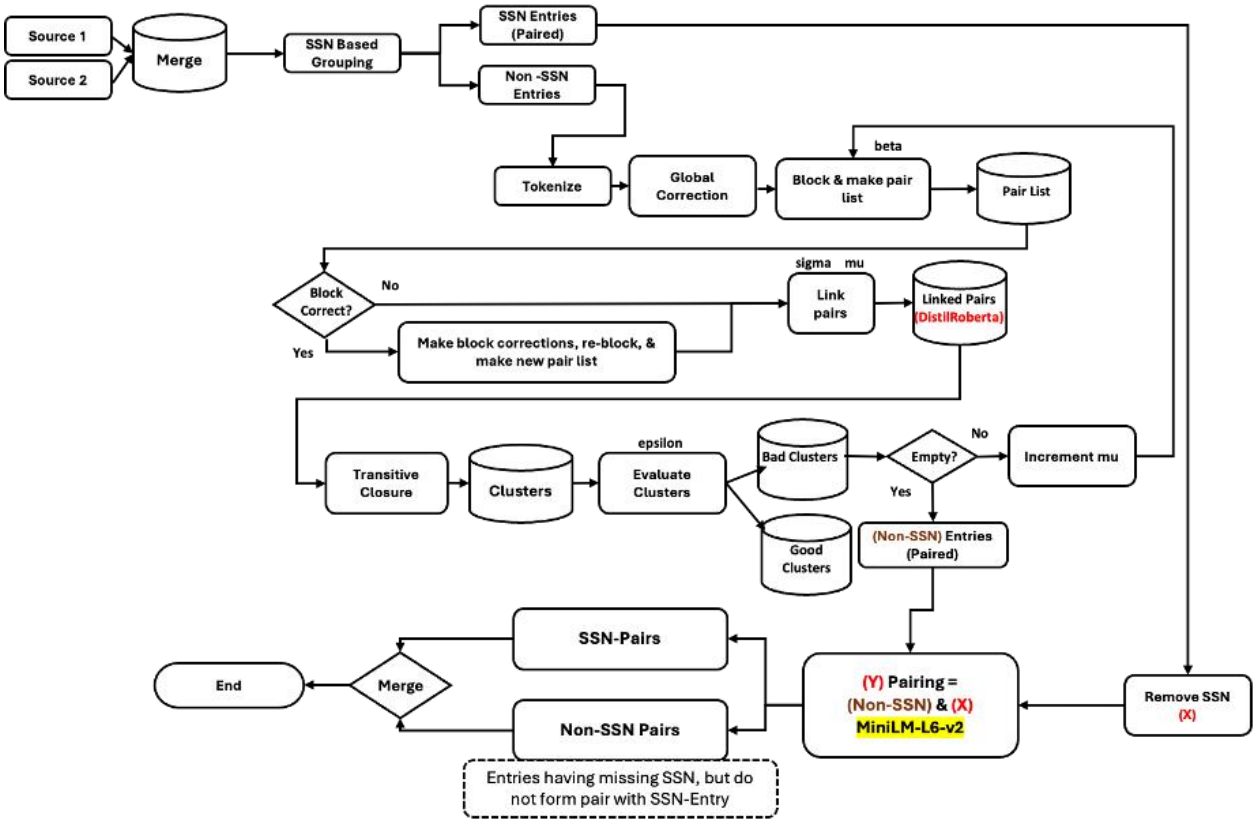

Figure 2. Flowchart of the addition to the data washing machine.

Now, the point is to pass both groups from the DWM whole processes, which include tokenization, blocking, linking, and the clustering processes. Then the non-based SSN records get saved in a new temporary file, and then that file will be given to the same path where we were running the machine learning-based linking comparator. Even if the new changes were made, the format of each pair's file was kept the same. Now, we have two different groups, but we also need to consider that some records might also match with each other in both groups. To put this under consideration, we added a machine learning pre-trained model to compare the records and then save them in the same pair format. After comparing those pairs and saving

them in the same format. The pairs, which belong to each other, get saved together in another file. All of the above steps help to make three kinds of pair files for the filtering in the linking process. Linked index, which passes through the DWM processes, formatted based on pairs that we filtered in the beginning of our process and the pairs we received from the original data. So, in the original dataset file, we compare the other two pair lists and make new pairs. Then we update the same file with those pairs, which will be given to DWM to go through the whole process. We tried our best to make minimal changes in the module to get aligned with the current processes of the DWM.

The entity resolution procedures and data cleansing capabilities were enhanced by the Natural Language Processing (NLP) jobs that were described. Several studies demonstrate how transformer-based models can improve entity matching (EM) linking by using pre-trained models like BERT, RoBERTa, and MiniLM, which are optimized to greatly increase matching quality.

A more detailed depiction of the characteristics of records can be obtained using embeddings. The model's capacity to discriminate between matches and unmatched might be improved, for example, by converting textual features into vector representations that capture semantic meanings. This is consistent with the paper's focus on how crucial similarity vectors are to the entity resolution procedure.

Since deep learning algorithms cannot understand the token sequence, we must translate it into a series of numerical numbers. Several transformation strategies may produce different results when using the same algorithm. We investigated state-of-the-art word embedding to determine the best vectorization methods for our design.

We can run this whole method of the SSN filtering with the traditional method, which is the scoring matrix, and choose to run either of the files to do further analysis as well. In the scoring matrix, the records get processed one by one. So, we give data in batches to make the process faster; we run Distill Roberta [19] to find out the similarity between the records and then return the output to use for the clustering process. It is now apparent that some records should have matched or should have been like each other. So, we have tried to match the records on the basis of their SSN and not for any other attribute. After comparing the results of the older method with the SSN filtering method, we found a huge change in our results.

6. Results

This research aims to explore designs for the matching process that operate effectively and help get better results in linking pairs of heterogeneously structured references. After the addition of the SSN filtering method, we saw that not only were we able to run the whole machine faster, but also it helped us to minimize the pairs as well in the return. Vectorization was done through the DistilRoBERTa model [19], and using those vectors to calculate the similarity, we found the results mentioned in **Table 3**. The results show the precision, recall, and F-measures of the DWM, where Mu represents the match threshold for linking two linked pairs each time in a data file. The mu value must be a decimal value between 0.0 and 1.0.

Table 3. Linking results through ML.

SAMPLE	PRECISION	RECALL	*F*-MEAS	MU
S1G.txt	1	0.963	0.9812	0.87
S2G.txt	0.9333	0.875	0.9032	0.8
S3Rest.txt	0.9074	0.875	0.8909	0.7
S4G.txt	0.9207	0.798	0.855	0.85
S5G.txt	0.8741	0.8421	0.8578	0.8
S6GeCo.txt	0.7018	0.7368	0.7189	0.82
S7GX.txt	0.8134	0.8495	0.8311	0.82
S8P.txt	0.5562	0.5813	0.5685	0.65
S9P.txt	0.7544	0.3671	0.4199	0.78
S10PX.txt	0.5997	0.2856	0.3869	0.74
S11PX.txt	0.8262	0.2163	0.3428	0.8
S12PX.txt	0.7829	0.4196	0.5539	0.73
S13GX.txt	0.9004	0.8861	0.8932	0.81
S14GX.txt	0.6467	0.8744	0.7435	0.81
S15GX.txt	0.8688	0.7893	0.8271	0.83
S16PX.txt	0.6788	0.528	0.4187	0.71
S17PX.txt	0.6986	0.5818	0.4016	0.73
S18PX.txt	0.7549	0.4249	0.5543	0.73

The results in **Table 2** show the results found without the SSN filtering process and are collected using the pre-trained model to find the pair's similarity. DistilRoBERTa was used to convert the pairs, a text, into vectors and then the cosine similarity formula was applied on those vectors, to get the matching scores. Keeping the other steps, which include tokenization, blocking, and clustering, the same but only focused on linking, which is the primary process of the data washing machine.

After taking the linked pairs in three ways and merging them together in a file to find out the similarity between those records, below are scores that were collected from the Data Washing Machine process while using the SSN filtering method. To consider all pairs, we made sure to use the Mini LM pre-trained model for the rest of the pairs as well to get the similarity.

As we can see, after using the filtering method, there has been a change in the accuracy of each file as well. A good dataset like S1G.txt, which has only 50 records, gave us a score for precision, recall, and 286 the F-measures. We further made the comparison, which can be seen in **Figures 3–5**, between both methods, and it can be seen that the SSN filtering method is a lot better with machine learning than the linking only with machine learning models.

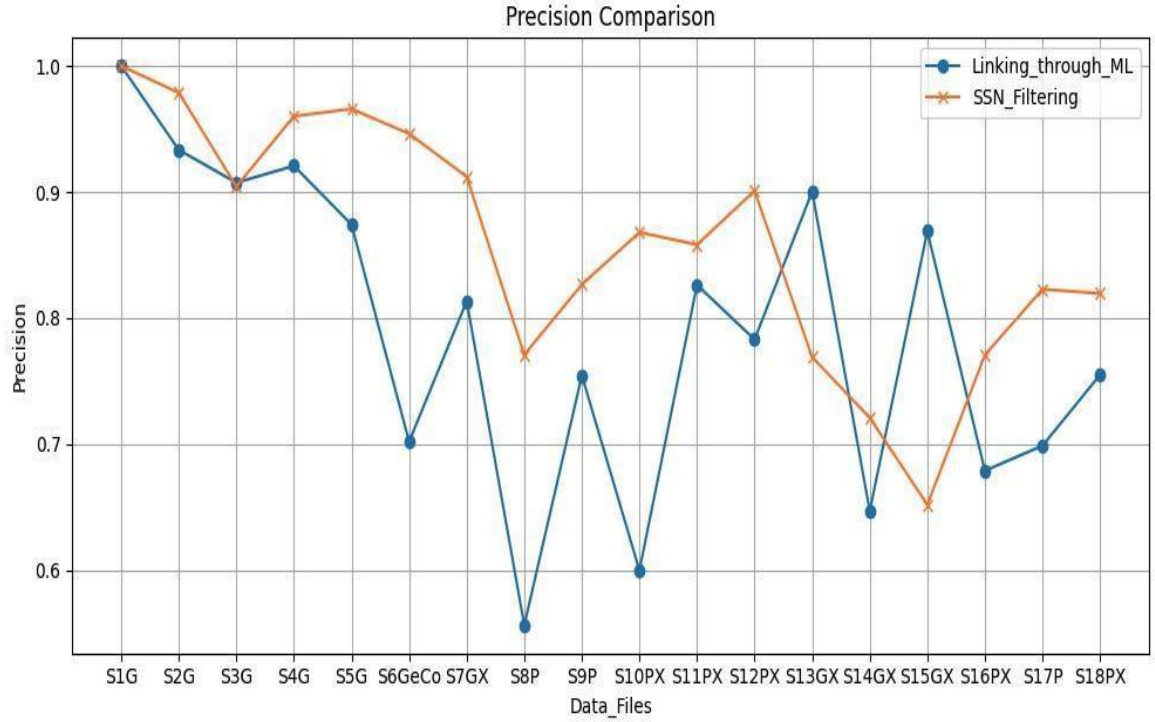

Figure 3. Comparison of precision values between Linking_through_ML vs SSN_Filtering.

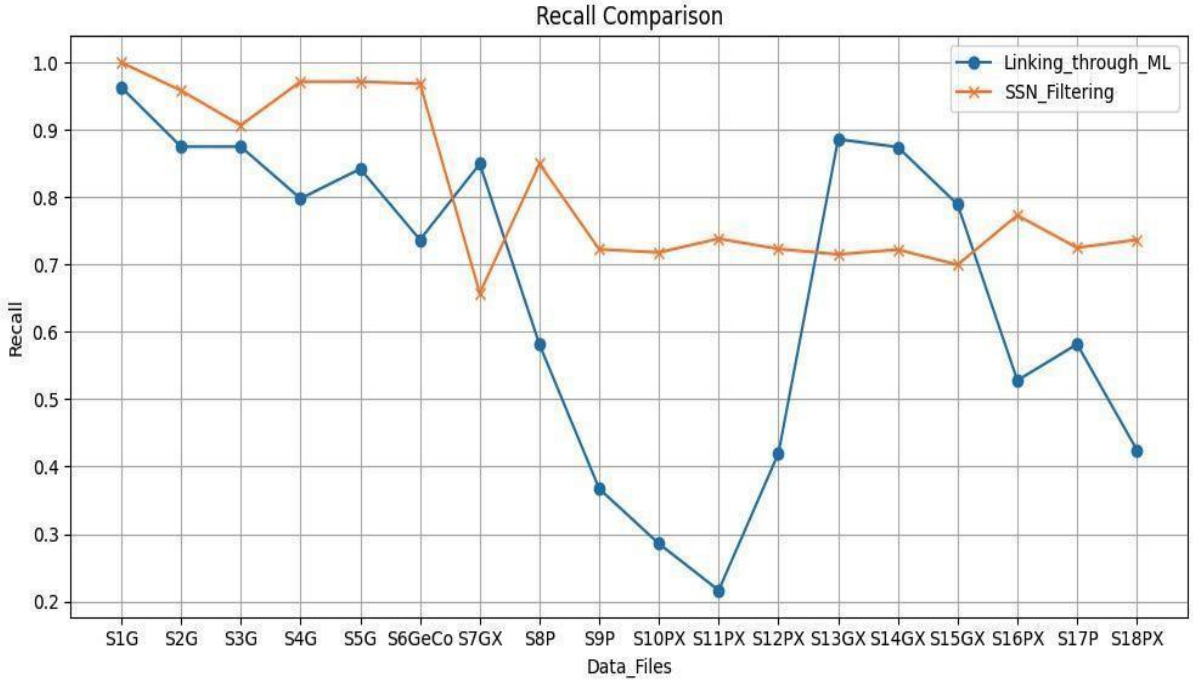

Figure 4. Comparison of recall values between Linking_through_ML vs SSN_Filtering.

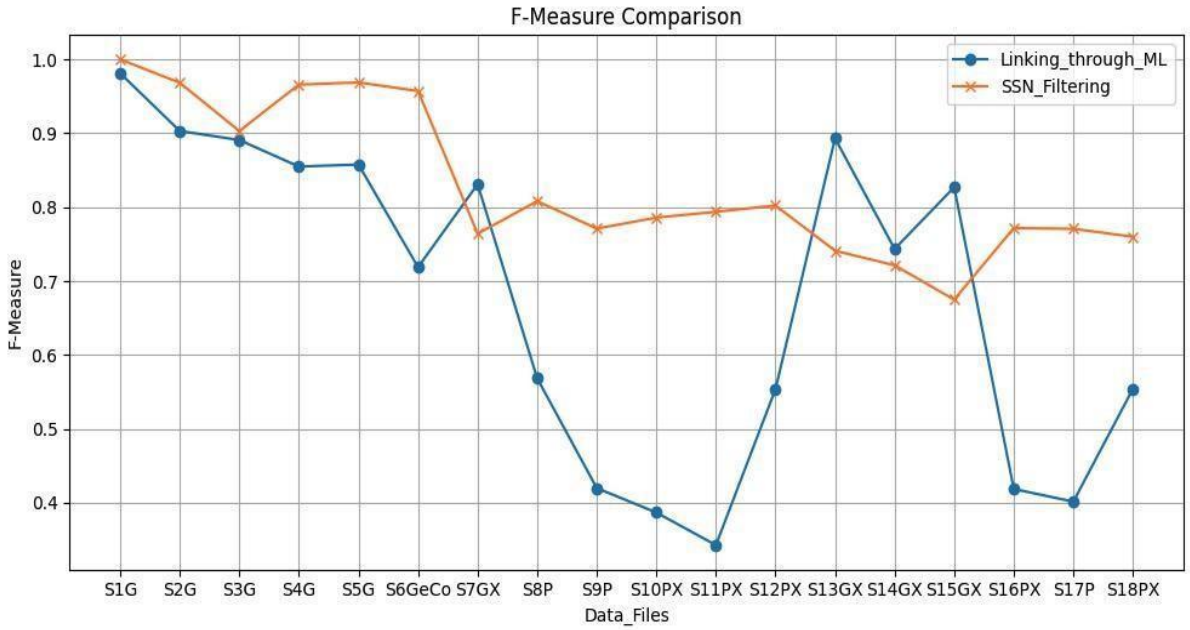

Figure 5. Comparison of F-measure values between Linking_through_ML vs SSN_Filtering.

The graph illustrates a comparative analysis of precision, recall, and F-measure metrics for two results tables we got for both linking processes across files S1G to S18G. The three metrics evaluate how well, completely, and accurately each table captures relevant data from every file. For both tables, precision scores are generally higher than recall, indicating a more critical ability to identify relevant instances correctly.

7. Discussion

The study's methodology introduces a novel approach to improving entity resolution procedures, emphasizing using Social Security Numbers (SSNs) as a crucial factor for entity matching in the DWM. This conversation will examine the virtues and limits of the suggested approach, its effects on data management and privacy, and its compatibility with current entity resolution and machine learning developments.

It makes sense to use SSN as the primary matching attribute because each person's SSN is unique. When matching enormous data sets, when name-based matching alone may not be sufficient, this approach is capable of drastically cutting down on false positives. To ensure compatibility with current processes, the Data Washing Machine (DWM) [20] system was strategically integrated with an SSN filtering module. Implementation is made easier with this modular approach, and future improvements are possible. An inclusive approach to entity resolution is offered by combining machine learning methods for data without SSNs with SSN-based matching. The accuracy and efficiency of the linking process in DWM increased overall with this hybrid approach.

The integration of advanced NLP techniques [21] and the hybrid approach for the linking process in the DWM demonstrate a forward-thinking methodology that aligns well with current trends in the field. Previously, we just wanted to assess if we could include Machine learning into DWM, but now, as we successfully made a hybrid

model, the scalability issues were still there. Machine learning models are highly versatile and can identify patterns or relationships that traditional methods would overlook, especially in datasets with varied topologies. On the other hand, researchers with limited resources can still fine-tune a pre-trained BERT model on a particular job with relative speed and substantially reduced processing demands [19]. The model's capability to distinguish between matches and unmatched can be improved by transforming textual attributes into vector representations that capture semantic meanings. This is consistent with the paper's focus on how important similarity vectors are to the entity resolution procedure [22].

Table 3 represents the results when we apply the machine learning model in the DWM. As you can see, machine learning has performed well in our case, and the results are closer to what traditional methods represent. Our plan would be to improve the linking process evaluation metrics using machine learning models. **Table 4** represents the inclusion of the filtering method and how it increased the accuracy of the clusters in the data washing machine and not only helped increase the accuracy but also gave faster results.

Table 4. SSN filtering linking results.

SAMPLE	PRECISION	RECALL	*F*-MEAS	MU
S1G.txt	1	1	1	0.6
S2G.txt	0.9787	0.9583	0.9684	0.7
S3REs	0.9043	0.9068	0.9031	0.67
S4G.txt	0.9601	0.9717	0.9659	0.73
S5G.txt	0.9659	0.9717	0.9688	0.73
S6GeCo.txt	0.9462	0.9685	0.9572	0.82
S7G.txt	0.9121	0.6574	0.7641	0.67
S8P.txt	0.7708	0.8495	0.8082	0.72
S9P.txt	0.8265	0.7226	0.7711	0.74
S10PX.txt	0.868	0.718	0.7859	0.74
S11PX.txt	0.8581	0.7384	0.7938	0.73
S12PX.txt	0.9009	0.7232	0.8023	0.81
S13G.txt	0.7689	0.7152	0.7411	0.81
S14GX.txt	0.7208	0.7221	0.7214	0.83
S15GX.txt	0.6519	0.6998	0.675	0.71
S16PX.txt	0.7706	0.7731	0.7718	0.73
S17PX.txt	0.8228	0.725	0.7708	0.73
S18PX.txt	0.8194	0.7368	0.7759	0.73

Comparing the Linking_through_ML and SSN_filtering techniques shows notable differences in performance for files S1G via S18PX in terms of precision, recall, and F-measure, as shown in **Figures 3–5**. The benefits and drawbacks of each method for linking or filtering across different file types are covered in this analysis.

In terms of precision, **Figure 3** represents the SSN_filtering generally outperforms Linking_through_ML, especially in files like S6GeCo, S8P, and S16PX

(Poor represents the poor data files), where values of precision for SSN_filtering are higher.

This implies that SSN_filtering is a dependable option when accuracy is crucial since it can more accurately identify pertinent occurrences with fewer false positives. However, both methods demonstrate better precision value in files such as S1G and S4G (G represents the good data files), implying that these files may present less complexity in correctly identifying relevant matches.

Recall values indicate a different layout. While Linking_through_ML shows essential recall losses for particular files (e.g., S9P, S10PX, S11PX, and S18PX), SSN_filtering maintains a generally high recall. Due to poor-quality data files or differences in the data in some files, this pattern suggests that Linking_through_ML would have trouble capturing all the relevant instances in those files. Lower recall results in Linking_through_ML, for example, for files with the prefixes “PX” or “P”, suggest that the matching criteria must be improved or parameters adjusted to capture more relevant instances.

According to the F-measure comparison, which balances recall and precision, SSN_filtering consistently yields better results in most files. SSN_filtering performs better at finding the optimal balance between accuracy and completeness based on this balance. Their notably high F-measure scores demonstrate that both strategies perform well in files such as S1G, S2G, and S4G.

A significant drop in F-measure is shown for specific files, including S9P, S10PX, and S16PX, implying that Linking_through_ML’s performance may not be as trustworthy across a broader range of file types as SSN_filtering.

In summary, these findings indicate that SSN_filtering offers more excellent stability and balanced performance across various kinds of files, either with good or poor-quality data files, even while Linking_through_ML can be valuable in situations where an appropriate level of recall flexibility is acceptable.

The model performance after we included SSN made a massive difference in the results, some similar and some different results overall, and the reason can be that it helped to pick up the essential features and helped with the entity matching process. When SSNs are inconsistently available in datasets with no related unique identifiers, the effectiveness of this approach may be compromised. Computational demands may rise because of the multi-stage procedure, which includes distinct processing for SSN and non-SSN records and cross-group comparisons.

8. Conclusion

In a data washing machine, the linking process has different comparators, scoring matrices, and linking through machine learning, but there was a need to minimize the pairs in this process. We came up with a new approach to do the linking by the filtering process through social security numbers. Through this approach, we were able to minimize the pairs, and it helped us to use the machine learning linking process better. It gave us minimum pairs for each file. As you. Can see in the result section, it did well in the good datasets and even for the poor dataset it gave us better results than what we had when we only used ML on all the records. Through this process that we

are using, we can also decide if we want to apply pertained models on all the records or to the ones with no SSNs in the records.

9. Future work

In Data washing machine, we have a parameters file for each dataset, where we decide to choose the parameters to run each datafile. DWM has a process called Parameter discovery Process (PDP). PDP is an unsupervised process that helps to provide starting parameters for a given dataset processed by the DWM. Now that our process, which is a hybrid method, includes a filtering process and a machine learning model, in future, we want to see how this process working with PDP boosts overall model performance and positively impacts the DWM. Considering how complicated the suggested method is, using explainable AI techniques could improve comprehension and confidence in the matching choices. Future research should resolve the noted drawbacks while preserving this strategy's advantages.

Author contributions: Conceptualization, JRT and BS; methodology, JRT and BS; software, JRT and BS; validation, JRT, AAH and BS; formal analysis, JRT, AAH and BS; investigation, JRT, AAH and BS; resources, JRT and AAH; data curation, JRT and BS; writing—original draft preparation, BS; writing—review and editing, BS and AAH; visualization, BS and AAH; supervision, AAH and JRT; project administration, AAH; funding acquisition, AAH. All authors have read and agreed to the published version of the manuscript.

Funding: This research is part of a broader collaborative effort supported by the National Science Foundation under Award No. OIA-1946391 through the NSF EPSCoR program. We sincerely appreciate the support provided by NSF, which has enabled this work. Additionally, we extend our gratitude to our collaborators and colleagues whose insights and contributions have enriched this study.

Conflict of interest: The authors declare no conflict of interest.

References

1. Hechler E, Weihrauch M, Wu Y. AI for entity resolution. In: Data Fabric and Data Mesh Approaches with AI. Apress; 2023.
2. Yang F, Zhang, C. Entity matching method and device and electronic equipment (Chinese). CN Patent 201811474215.1, 22 April 2022.
3. Barlaug N, Gulla JA. Neural Networks for Entity Matching: A Survey. ACM Transactions on Knowledge Discovery from Data. 2021; 15(3): 1-37. doi: 10.1145/3442200
4. Agarwal A, Singh S, Chaurasiya VK. Assessing Entity Resolution techniques based on deep learning. In: Proceedings of the 2022 IEEE 3rd Global Conference for Advancement in Technology (GCAT); 2022.
5. Carlsson R, Lundström O, Arizmendi GM, Olsson H. System and method for matching entities. WIPO Patent Application No. 2010063311A1, 10 June 2010.
6. Kong C, Gao M, Xu C, et al. Entity Matching Across Multiple Heterogeneous Data Sources. In: Proceedings of the 21st International Conference, DASFAA 2016; April 16-19, 2016; Dallas, TX, USA.
7. Papadakis G, Fisichella M, Schoger F, et al. Benchmarking Filtering Techniques for Entity Resolution. In: Proceedings of the 2023 IEEE 39th International Conference on Data Engineering (ICDE); 2023.
8. Papadakis, G., Palpanas, T., & Koutrika, G. (2020). Entity Resolution Methods for Big Data. ACM Computing Surveys (CSUR), 53(1), 1-42.

9. Halimeh, Ahmed Abu. Integrating information quality in visual analytics. University of Arkansas at Little Rock, 2011.
10. Christophides V, Efthymiou V, Palpanas T, et al. An Overview of End-to-End Entity Resolution for Big Data. ACM Computing Surveys. 2020; 53(6): 1-42.
11. M. I. Sarker and M. Milanova, "Deep Learning-Based Multimodal Image Retrieval Combining Image and Text," 2022 International Conference on Computational Science and Computational Intelligence (CSCI), Las Vegas, NV, USA, 2022, pp. 1543-1546, doi: 10.1109/CSCI58124.2022.00274.
12. Navid P. Deep Dive: What the Heck Is Entity Resolution. Deep Dive: What the Heck Is Entity Resolution. Pedram's Data Based; 2022.
13. Papadakis G, Skoutas D, Thanos E, et al. Blocking and Filtering Techniques for Entity Resolution. ACM Computing Surveys. 2020; 53(2): 1-42. doi: 10.1145/3377455
14. Wang T, Kou Y, Shen D, et al. SIER: An Efficient Entity Resolution Mechanism Combining SNM and Iteration. In: Proceedings of the 2014 11th Web Information System and Application Conference; 2014.
15. Binette O, Steorts RC. (Almost) All of Entity Resolution. arXiv; 2020.
16. Al Sarkhi A, Talburt J. A scalable, hybrid entity resolution process for unstandardized entity references. Journal of Computing Sciences in Colleges. 2020; 35(9): 19-29.
17. Al Sarkhi A, Talburt JR. Estimating the parameters for linking unstandardized references with the matrix comparator. Journal of Global Information Technology Management. 2018; 10(4): 12-26.
18. Sajid B, Abu-Halimeh A, Jakoet N. Pre-trained models for linking process in data washing machine. Computing and Artificial Intelligence. Published online November 1, 2024: 1450. doi: 10.59400/cai.v3i1.1450
19. Sanh V, Debut L, Chaumond J, Wolf T. Distilbert, a Distilled Version of Bert: Smaller, Faster, Cheaper and Lighter. arXiv; 2020.
20. Wang W, Wei F, Dong L, et al. MiniLM: Deep Self-Attention Distillation for Task-Agnostic Compression of Pre-Trained Transformers. In: Proceedings of the 34th Conference on Neural Information Processing Systems (NeurIPS 2020); 1 January 1970; Vancouver, Canada.
21. Talburt JR, Al Sarkhi AK, Pullen D, et al. An Iterative, Self-Assessing Entity Resolution System: First Steps toward a Data Washing Machine. International Journal of Advanced Computer Science and Applications. 2020; 11(12).
22. Zeakis A, Papadakis G, Skoutas D, et al. Pre-Trained Embeddings for Entity Resolution: An Experimental Analysis. Proceedings of the VLDB Endowment. 2023; 16(9): 2225-2238. doi: 10.14778/3598581.3598594