ELKOMIKA: Jurnal Teknik Energi Elektrik, Teknik Telekomunikasi, & Teknik Elektronika

ISSN(p): 2338-8323 | ISSN(e): 2459-9638 | Vol. 13 | No. 1 | Halaman 57 - 69
DOI : http://dx.doi.org/10.26760/elkomika.v13i1.57 January 2025

Development of a Data Cleaning System for Consumer Master Data using Sorted Neighborhood and N-Gram Methods

Article History:
Received
2 January 2025
Revised
20 January 2025
Accepted
28 January 2025

YUSUF LESTANTO, RAHMA MUALIFA

Program Studi Informatika, Universitas Bakrie, Indonesia
Email: yusuf.lestanto@bakrie.ac.id

ABSTRAK

Penelitian ini mengembangkan sistem pembersihan data master menggunakan metode Sorted Neighborhood Method (SNM) dan N-gram untuk mendeteksi dan menghilangkan duplikasi serta menstandarkan format nama dan alamat. SNM menangani pra-pembersihan, menghapus karakter dan judul tertentu, dan membentuk token untuk perbandingan. N-gram menghitung kemiripan dengan nilai dan ambang batas yang ditentukan. Efektivitas metode dievaluasi menggunakan metrik recall, precision, dan F-measure pada dua set data: kecil dan besar. Ambang batas optimal, panjang token, dan nilai N-gram masing-masing adalah 0.7, 5, dan 2, menghasilkan nilai F-measure tertinggi. Hasilnya mengonfirmasi keberhasilan implementasi dan meningkatkan kualitas data. Identifikasi parameter optimal memberikan tolok ukur untuk upaya pembersihan data, berpotensi menyederhanakan proses dan mengurangi sumber daya pemeliharaan data.

***Kata kunci**: Pembersihan data, Deteksi duplikasi, Sorted Neighborhood, N-Gram, Kualitas data.*

ABSTRACT

This study developed a data cleaning system for master data using the Sorted Neighborhood Method (SNM) and N-gram methods to detect and eliminate duplicates and standardize name and address formats. The proposed SNM algorithm handles precleaning tasks, removes specific characters and titles, and forms tokens for comparison. The N-gram algorithm calculates record similarity using user-defined N-gram values and thresholds. The effectiveness was evaluated using recall, precision, and F-measure metrics on small and large datasets. The optimal threshold, token length, and N-gram values were 0.7, 5, and 2, respectively, yielding the highest F-measure scores. The results confirm the successful implementation and improvement of data quality. Identifying optimal parameters provides a benchmark for future data-cleaning efforts, potentially streamlining processes and reducing resources.

***Keywords**: Data cleaning, duplicate Detection, Sorted Neighborhood, N-Gram, Data quality.*

1. INTRODUCTION

Data heterogeneity, volume, and complexity can all pose challenges in the data cleaning process **(Maharana, Mondal, & Nemade, 2022)**. Data can be sourced from numerous locations, each with its own distinct format and data entry protocols, which complicates the cleaning process **(Liu & Panagiotakos, 2022)**. Large datasets can be a significant challenge because they often contain numerous errors that are difficult to identify and resolve. The data can be complicated and feature complex hierarchies and interconnected relationships, which necessitate a recursive cleaning process.

The data cleaning process involves identifying erroneous, insufficient, or ambiguous data and correcting these issues to produce high-quality data. The process involves verifying the format and accuracy of the data and removing any duplicates or other discrepancies. The data cleaning process typically requires repeated cycles and phases because of the possibility of uncovering additional problems as the process progresses. To guarantee accuracy and consistency, both automated tools and manual checks must be employed **(Aldoseri, Al-Khalifa, & Hamouda, 2023).** Furthermore, having a solid understanding of the subject matter is essential for efficiently cleaning data because doing so enables the identification and resolution of anomalies within the correct context and according to specific data requirements.

Information production relies heavily on data. Numerous organizations and businesses with complex operational processes require substantial amounts of data. Decision-making processes use data derived from various business operations **(Awan et al., 2021).** The quality of the available data significantly influences the effectiveness of an organization's or company's decision making. However, data from external sources often contain errors. This typically results from poor data quality, manifested as duplicate entries, spelling errors, and incomplete information. Such imperfect data lead to inaccurate reports, thereby causing flawed decision-making within organizations and companies. A Company maintains a database containing tens of thousands of records **(Bousdekis, Lepenioti, Apostolou, & Mentzas, 2021).** This study examines low-quality consumer data due to duplicate entries and non-standardized formatting, particularly in the recording of customer names and addresses.

To identify data duplication, this study employed a Sorted Neighborhood Algorithm that establishes a specific token and then combines and removes two or more duplicate data entries. Another technique used to determine whether a record is a duplicate is the N-Gram Method, which assesses string similarity **(Yu, Li, Deng, & Feng, 2016) (Wei, Yu, & Lu, 2017)**. This study implements both methods to detect duplicates in the consumer master data.

The Sorted Neighborhood algorithm is used to identify duplicate data by creating a unique token and then combining and eliminating all identical data **(Li, Xie, & Ding, 2015) (Kejriwal & Miranker, 2015)**. The process begins by placing the information based on a distinct identifier. Then, a window of predetermined size is moved through the ordered list, comparing only the entries within its scope. The window dimensions can be adjusted to strike a balance between precision and computational speed, which makes this approach adaptable to various data deduplication requirements. The N-gram approach is used to calculate the similarity between strings, which is used to determine if the records are duplicates **(Foroozan, Murad, Sharef, & Latiff, 2015) (Singh & Chaudhari, 2016) (Mishra, Mehta, & Rasiwasia, 2021)**. This method involves breaking strings into smaller substrings of length N (called N-grams) and comparing the overlaps between these substrings. The similarity score is typically calculated as the ratio of the matching N-grams to the total number of N-grams in

both strings. N-gram similarity can be particularly effective for detecting approximate matches and handling minor text variations, such as typos or slight differences in formatting.

This study constructs a data cleaning system for consumer master data that applies the Sorted Neighborhood Method (SNM) and the N-gram technique. The effectiveness of the detection results in terms of data duplication was also measured. The goal of this study is to create a data-cleaning system for consumer master data using SNM and N-gram algorithms.

2. METHOD

Data cleaning is closely linked to the process of data acquisition and definition with the aim of enhancing the quality of existing data within a system. As one of the initial stages in data mining, data cleaning is referred to in various terms such as data scrubbing, data cleansing, error checking, error correction, and error detection.

A crucial task in data cleaning or scrubbing is detecting duplicate data **(Faiz, 2019) (Jiang, Huang, & Zhou, 2020).** On the database, usually normally will face problems of data such as: (1) errors or less complete the writing of the result of human error when the process of entering data, (2) value entered is inconsistent due to the difference in format when entering data, (3) is not full of information, (4) the client is moved from one place to the other without any notice, and (5) the client of an error when entering your name and address.

Ridzuan & Wan Zainon (2019) noted that data cleaning lacks a universally accepted definition or perspective. The process of data cleaning is implemented differently across various knowledge discoveries in databases (KDD) and system data-mining approaches, depending on the specific issues present in the datasets being analyzed.

2.1 Data Cleaning

In data analysis, the crucial process of data cleaning involves detecting and rectifying errors, inconsistencies, and inaccuracies within the datasets to enhance their quality. This procedure encompasses various tasks including eliminating duplicate entries, addressing missing data points, unifying formats, fixing typographical and spelling errors, and addressing discrepancies across multiple data sources. To effectively clean data, a combination of automated tools and human oversight is necessary, often requiring several rounds of review as new issues emerge. Domain knowledge is crucial for understanding the context and requirements of data **(Schuster, van Zelst, & van der Aalst, 2022).** While time-consuming, thorough data cleaning is essential to ensure the reliability and validity of subsequent analyses and insights derived from the data. Common techniques include data validation, transformation, outlier detection, and reconciliation across sources. The goal was to produce a clean, consistent, and accurate dataset that fits the intended analytical purpose.

Data cleaning is a multifaceted process that extends beyond simple error corrections. This involves a comprehensive approach to data quality improvement using various techniques and methodologies. A key component of data profiling is the examination of data characteristics. This process involves scrutinizing the dataset's organization, information, and internal connections to identify potential problems and irregularities. This step helps us understand the overall data landscape and guides subsequent cleaning efforts. In addition, data standardization plays a vital role in ensuring consistency across datasets, particularly when dealing with information from multiple sources or time periods. This may involve normalizing the units of measurement, date format, or categorical variables to facilitate accurate comparisons and analyses.

In addition, data cleaning often requires a deep understanding of the business context and domain-specific knowledge. This expertise is essential for making informed decisions on how to handle complex data issues, such as outliers or ambiguous values. For instance, in financial datasets, an unusually large transaction might be a genuine high-value sale or data entry error, and distinguishing between these scenarios requires industry insight. Moreover, data cleaning is not a one-time activity but an ongoing process that should be integrated into the data management lifecycle. As new data are collected or external factors change, previously cleaned datasets may require re-evaluation and further refinement to maintain their quality and relevance for analytical purposes.

2.1 Data Cleaning Method

In data analysis, the crucial step of data cleaning involves detecting and rectifying errors, discrepancies, and inaccuracies in the datasets to enhance their overall quality. This process encompasses various techniques, such as eliminating duplicate entries, addressing missing data through imputation or removal, ensuring consistency in formats and units across different variables, and reconciling disparities between multiple data sources. These practices are essential for improving the reliability and usability of datasets during analytical processes. Effective data cleaning often requires multiple iterations, a combination of automated tools, and manual reviews to produce a clean, consistent, and accurate dataset suitable for analysis.

2.1.1 Duplicate Data Detection Algorithm

The process of merging or deleting data, which combines the duplication of data from two or more duplicates **(Fan & Geerts, 2022)**, uses the following method to detect duplicate data:

1. Sorted Neighborhood Method (SNM)
   The proposed method addresses the issue of duplicate entries in consecutive positions. The data redundancy was then identified within a defined window size to limit comparisons between the data elements. The memory consumption of this technique is $\boldsymbol{O}(N \log N)$, where N denotes the total number of entries in the database.
2. SNM method with clustering technique.
   This approach is similar to the SNM technique. The key difference is the initial segmentation of data into several clusters or categories. This categorization can be performed separately depending on the attributes of the data that require cleaning. Following categorization, data replication was performed for each cluster. The memory required to process this method was O(N log N/C), where N is the total number of database entries, and C is the size of each cluster.

This analysis suggests that employing clustering techniques with SNM is more memory efficient than standard SNM methods, as evidenced by the relationship $\boldsymbol{O}(N \log N/C) < \boldsymbol{O}(N \log N)$. As a result, this study utilized clustering techniques in conjunction with SNM as the preferred approach for identifying duplicate data.

2.1.2 The Sorted Neighborhood Method for Detecting Duplicate Data

The Sorted Neighborhood method is used in this study, and it sorts by the clustering neighborhood consisting of the following steps:.

1. Cluster data. Constant partitioning was used to obtain cluster data by dividing the information into multiple groups based on attribute values. The data were first identified and then separated based on the characteristics of the study area. Comparisons and detection were performed exclusively within each individual group or cluster region.

2. Form a key or token: A pre-cleaning process was initiated before the forming stage. This process involves the removal of specific components, such as deletions, titles, punctuation, and characters. Observational data were first examined to identify the characters that should be eliminated. Then, each field generates a new token by combining one or more letters or numbers for every word. An illustration is given by the use of the first three characters of a word or string. This technique is known as process-record tokenization.
3. Sort data: Strings or words in the field are sorted based on the key outlined in Item 1.
4. Combining the data: A sliding window with a width w moves across each entry to limit the comparison of entries that may contain identical information. The *w* value represents the number of segments in each group. A new entry enters the window and is evaluated against the previous *w-1* entries to identify those entries with the corresponding information.

2.1.3 N-Gram Algorithm to Calculate Similarity Between Strings

The N-gram approach algorithm was used to compare two records by calculating the similarity between two strings or distinct words. The purpose of N-Grams is to use n consecutive letters for word representation. The values of n are 2, 3, and 4. When n = 2, it is referred to as a diagram or bigram. For n = 3, the trigram term is used **(Obiedat et al., 2022).** The following formula was used to compute the N-gram similarity between strings A and B:

$$sim_{AB} = \frac{\left(2\ (|ngram(A) \cap ngram(B)|)\right)}{ngram(A) + ngram(B)} \quad (1)$$

Assessment of similarity between two strings with values ranging from 0 to 1. The closer a string's similarity is to a particular standard, the closer its value is to one. In contrast, a nearby value is almost zero. This example demonstrates the application of an N-gram in measuring the similarity between strings, specifically, for n = 2.

String A = "APOTIK", has 2-gram: AP, PO, OT, TI, IK.
String B = "APOTEK", has 2-gram: AP, PO, OT, TE, EK.

The two examples of the string above have three pieces of the same bigram: AP, PO, and OT, which have similarity values:

$$Sim_{AB} = \frac{2x3}{5+5} = 0.6 \quad (2)$$

Determine whether two data strings qualify for provisions by setting a specific threshold value. Here, the predetermined threshold value is 0.6. Consequently, if the similarity value between two strings exceeds 0.6, the string is classified as having a similarity rating of 2. If the similarity value between the two strings is below 0.6, they are considered distinct.

2.2 Algorithm Flow Design

The algorithm flow design of data cleaning typically involves several key steps to systematically process and improve data quality. The process begins with data profiling, which analyzes the structure, content, and relationships of the dataset to identify potential issues. This was followed by data standardization, which ensured consistency across the dataset by normalizing the formats, units and categorical variables. The next step involves duplicate detection and removal, which often uses methods such as the Sorted Neighborhood Method (SNM) or N-gram algorithms to identify, merge, or delete duplicate records. Missing value handling is then addressed using techniques such as imputation and deletion. In the following, we describe the

error correction, focusing on fixing the typos, inconsistencies, and inaccuracies identified during the profiling stage. Data transformation can be applied to convert data into more suitable formats for analysis. Data validation checks were performed throughout the process to ensure that the cleaned data met predefined quality criteria. The flow often includes iterative steps because new issues can be identified during the cleaning process, which requires multiple passes through the data. The cleaned dataset underwent a final quality assurance check before being approved for use in subsequent analyses.

2.2.1 The Design Flow Of Data Duplication Detection Algorithm

Duplicate data detection is performed in three distinct stages: preprocessing, processing, and postprocessing stages. Data duplication was detected in the master consumer dataset:

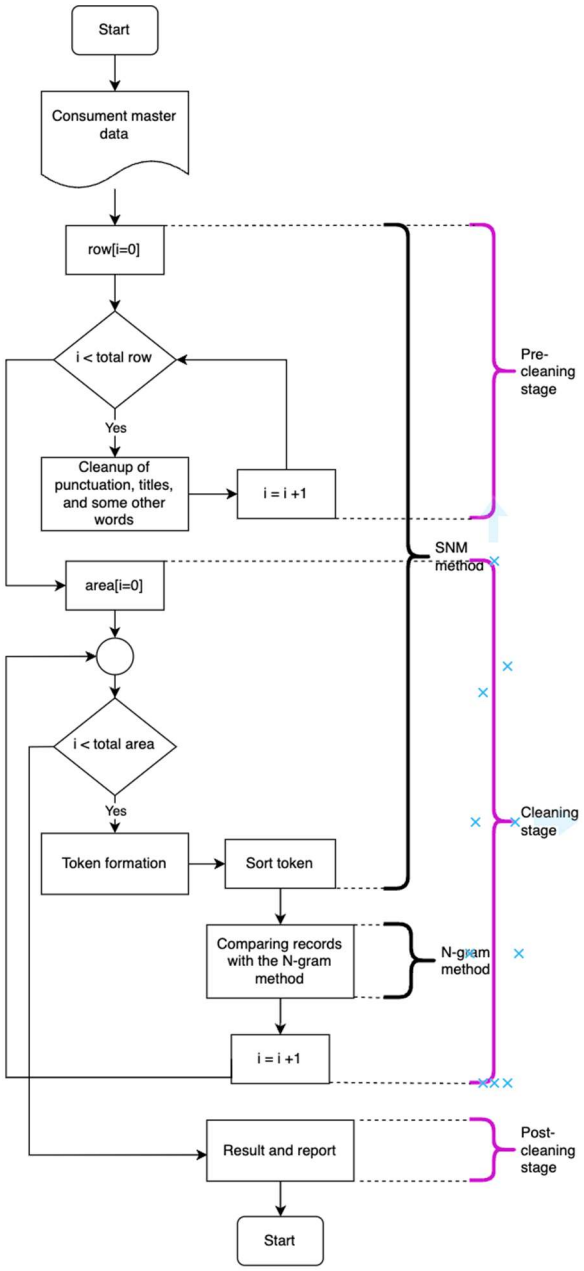

Figure 1. Flowchart duplicate data detection algorithm

2.2.2 The Design Flow Of Data Duplication Detection Algorithm

The initial stage of pre-cleaning involves stripping characters from the name and address fields before processing, which allows the system to detect duplicate data more efficiently.

Table 1. Before passing pre-cleaning step

ID	Name	Address
109456	GUARD MEDAN SUN PLAZA, SPM	K.H. ZAINUL ARIFIN No. 7, JL
142789	GUARDIAN SUN PLAZA, APT	K.H. ZAINUL ARIFIN No. 7, JL

Table 2. After passing pre-cleaning step

ID	Name	Address
109456	GUARD MEDAN SUN PLAZA	KHZAINUL ARIFIN 7
142789	GUARDIAN SUN PLAZA	KHZAINUL ARIFIN 7

An examination of Tables 1 and 2 reveals that certain elements have been removed from the data fields. The name category underwent a refining process, with the removal of commas and the terms "SPM" and "APT". The address column underwent a cleaning process to remove punctuation marks, commas, and the abbreviation "JL".

The data cleaning process resulted in the creation of streamlined datasets that were cleared of unwanted punctuation, and the elimination of these extraneous components ensured a uniform data format across all entries. Standardizing the data not only increased overall consistency and ensured easier analysis and streamlined information processing. In addition, eliminating unnecessary terms and punctuation improves data readability.

2.2.3 Data Clustering

Data clustering involves segmenting information based on data source locations to streamline the record comparison process. This approach divides records into clusters or groups based on location, allowing comparisons to be restricted to each area rather than across the entire database. For example, records from place one would only be compared to other place records, not to the full consumer database. The proposed method significantly reduced the computational complexity and processing time required for duplicate detection on large datasets. It can analyze data more efficiently by limiting comparisons to specific location clusters.

However, this method may miss potential duplicates across different regions, and the location-based clustering approach is a viable solution for addressing the challenge of duplicate detection in large-scale datasets. The proposed method balances acceptable accuracy with computational efficiency, which is crucial when dealing with large amounts of data. By leveraging geographical information to group potential duplicates, this approach significantly reduces the search space and computational requirements compared to exhaustive pairwise comparisons.

The proposed technique’s compromise is particularly valuable in scenarios where perfect accuracy is not essential, but the rapid processing of extensive datasets is paramount. This allows identification of a substantial proportion of duplicate entries while minimizing the time

and resources required for the task. This efficiency gain has become increasingly important as the volume of data grows exponentially across various domains.

In addition, the location-based clustering approach can be further refined and adapted to specific use cases, potentially incorporating additional metadata or employing more sophisticated clustering algorithms to enhance performance. Although it may not capture every single duplicate, especially in cases where location data are imprecise or missing, it provides a solid foundation for large-scale duplicate detection efforts that can be complemented by other techniques when necessary.

2.2.4 Formation of Token

The token identification process using n involves extracting the initial letter from each word in a given string. The value of n is specified by the user when a duplicate data-detection procedure is initiated. For illustrative purposes, table below demonstrates the methodology using n = 3.

A token identification process using n is employed in data analysis and text processing to create unique identifiers for words or phrases. The proposed method extracts the first n letters from each word in a given string, where n is a user-defined parameter. Thus, it generates a simplified representation of the original text, which can be useful for various purposes, such as duplicate detection, data clustering, and efficient indexing.

When applied to duplicate data detection, the proposed method allows for more flexible comparison between strings by focusing on their initial characteristics. The value of n can be adjusted based on specific task requirements. For example, setting n to 3 extracts the first three letters of each word, thus providing a balance between uniqueness and computational efficiency. This approach can be particularly effective in identifying similar entries that may have slight variations in spelling or formatting because it reduces the impact of minor differences while still capturing the essence of the original text.

Table 3. Sample after tokenization process

ID	Name	Address
109456	GUA MED SUN PLA	KHZ ARI 7
142789	GUA SUN PLA	KHZ ARI 7

2.2.5 Sorting the Tokens

In information retrieval tasks, token sorting is a critical process. After tokenization of the text into discrete words or subwords, token sorting facilitates efficient indexing, searching, and analysis. Prevalent methodologies include alphabetical, frequency-based, and semantic sorting based on word embedding. Alphabetical sorting enables expeditious lookup and binary-search algorithms. Frequency sorting arranges tokens in descending order of occurrence, which is advantageous for identifying key terms. The selection of the sorting method is contingent on specific applications and objectives. Appropriate token sorting facilitates downstream tasks such as duplicate detection, clustering, and information extraction, through the systematic organization of lexical units. Subsequently, the tokens within each field were sorted and consolidated, as illustrated in the following exemplary table.

Table 4. After the token is sorted and combined

ID	Name	Address
109456	GUAMEDPLASUN	7ARIKHZ
142789	GUAPLASUN	7ARIKHZ

2.2.6 Comparison of Value Similarity in Record using N-Gram Method

The records used in the N-gram method were analyzed to determine similarity values using Equation (1). The procedure for calculating the similarity between strings with n = 2 is as follows.

String 1 = GU UA AM ME ED DP PL LA AS SU UN = 11

String 2 = GU UA AP PL LA AS SU UN = 8

A total of the same gram = 7. The similarity value for *the field Name* is (2 × 7) / (11+8) = 0.7.

Another example:

String 1 = 7A AR RI IK KH HZ = 6

String 2 = 7A AR RI IK KH HZ = 6

A total of the same gram, = 6. The similarity value for *the field Address* was (2 × 6) / (6+6) = 1.0.

For the field names and addresses, a predetermined threshold value of 0.6 was established. In this example, the existing record is classified as a duplicate because the similarity scores for both the Name and Address fields exceed this established threshold.

3. RESULTS AND IMPLEMENTATION

The necessary environments to set up the data cleaning system comprised of a Microsoft SQL Server Database, an IIS version 8.0 web server, and either Internet Explorer or Mozilla Firefox as the web browser.

Database design is employed to illustrate the connections between data within a database based on fundamental data entities that exhibit interdependent relationships. The following diagram outlines the database schema of the consumer data master.

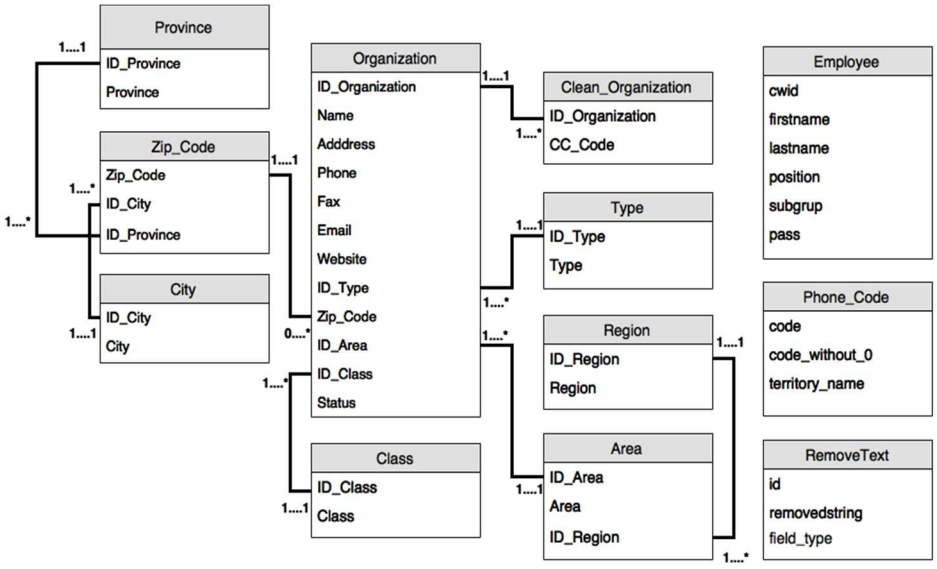

Figure 2. Design of a data cleaning system for a consumer master data database

This phase was conducted to evaluate the effectiveness of the methods used in the data-cleaning systems on two distinct dataset samples.

1. A small dataset of 2,500 samples was selected randomly from the entire dataset.
2. A large dataset comprising 25,000 samples was selected to yield more accurate results.

Before conducting the evaluation, it was imperative to manually enumerate known duplicate data points to assess the efficacy of the selected samples. The effectiveness value was determined by calculating recall, precision, and F-measure values.

$$precision = \frac{Total\ correct\ data\ duplication\ found}{Total\ data\ that\ has\ been\ found} \quad (3)$$

$$recall = \frac{Total\ correct\ data\ duplication\ found}{Total\ actual\ duplicate\ data} \quad (4)$$

$$f_{measure} = \frac{2\ x\ precision\ x\ recall}{precision + recall} \quad (5)$$

The initial study used a limited dataset (data 2.500). A 70-line repetition is identified within the small dataset. During the process of detecting data duplication, the authors must determine the threshold value, token cutting length, and n-gram data for small-scale experiments. Subsequently, they were evaluated using a combination of these three parameters. This approach facilitates identification of the optimal combination to achieve the most effective results or the highest F-measure (a composite of recall and precision values). The outcomes of the tests using this limited dataset are as follows.

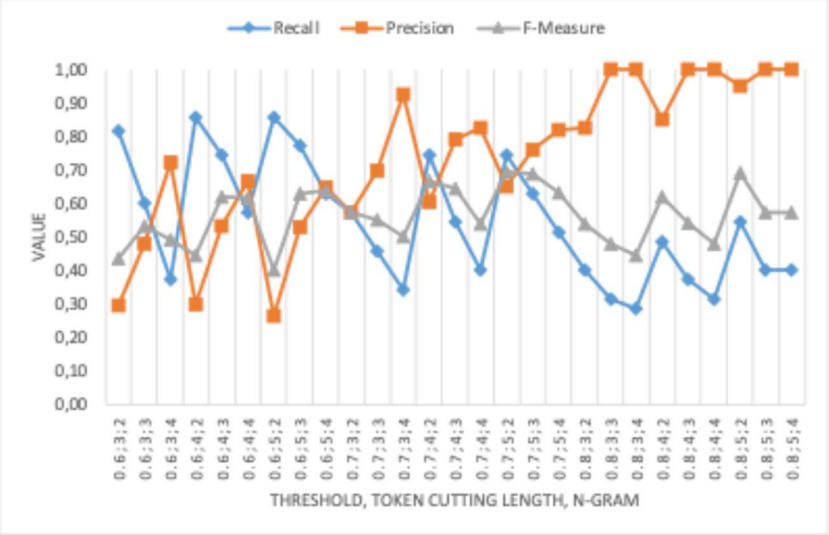

Figure 3. Graph test results small dataset = 2.500 data

The above graph illustrates that the optimal f-measure scores were attained with a threshold of 0.7, a token cutting length of 5, and N-gram values of 2, yielding F-measure scores of 0.69, 0.74, and 0.65, respectively.

The following examination employed an extensive dataset containing 25,000 entries. Within this substantial collection of 25,000 data points, 1,388 entries were duplicates. The results of the investigation conducted using this comprehensive dataset are summarized as follows.

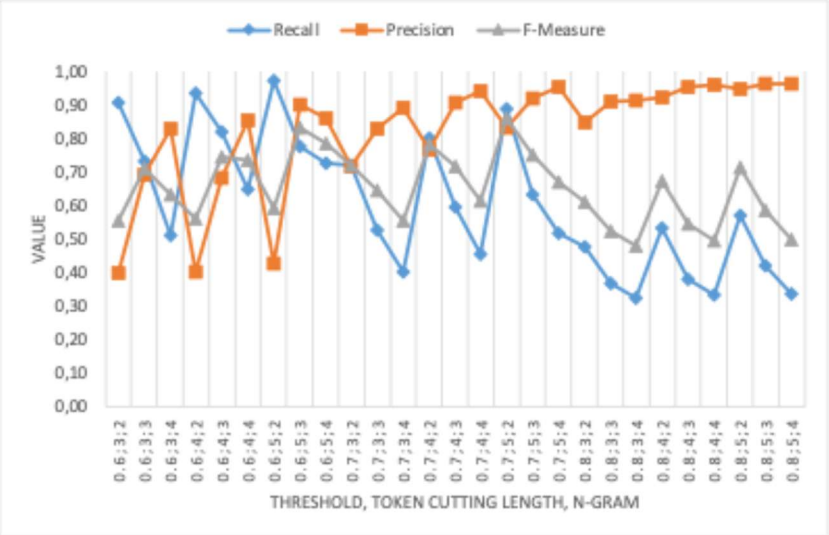

Figure 4. Graph test results large dataset = 25.000 data

The graph in Figure 4 indicates that the optimal f-measure values were achieved when the threshold was set to 0.7, the token cutting length was 5, and the N-gram value was 2. These parameter settings resulted in F-measures of 0.86, 0.89, and 0.83, respectively, representing the peak performance observed in the analysis.

F-measure measurements were obtained from experiments using both large and small datasets. The best f-measure was achieved by assessing two values for each of the following parameters: fusion threshold (0.7), token truncation length (5), and N-gram value (2).

4. CONCLUSION

This study proposed a data cleaning system for consumer master data by implementing the Sorted Neighborhood Method (SNM) and N-gram techniques to identify and remove duplicate data, as well as standardizing name and address formatting. The SNM algorithm performs preliminary tasks and generates tokens for data comparison, whereas the N-gram algorithm computes record similarity based on user-defined values and thresholds. The method's performance was assessed using recall, precision, and F-measure metrics on datasets comprising 2,500 and 25,000 records, respectively. The parameters that achieved the best F-measure scores were an optimal threshold of 0.7, token length of 5, and N-gram values of 2. The findings validated the effective deployment of the proposed method and enhanced the quality of the master data. The establishment of optimal values serves as a standard for future data cleaning operations, thereby simplifying the process and reducing the time and resources required for data upkeep. The scalability of the proposed method was demonstrated by testing it on datasets of varying sizes, encompassing both smaller and larger ones. The findings demonstrate that the proposed approach is adaptable to businesses of different scales and provides a flexible solution to enhance data accuracy.

REFERENCES

Aldoseri, A., Al-Khalifa, K. N., & Hamouda, A. M. (2023). Re-Thinking Data Strategy and Integration for Artificial Intelligence: Concepts, Opportunities, and Challenges. *Applied Sciences, 13*(12), 7082. Retrieved from https://www.mdpi.com/2076-3417/13/12/7082

Awan, U., Shamim, S., Khan, Z., Zia, N. U., Shariq, S. M., & Khan, M. N. (2021). Big data analytics capability and decision-making: The role of data-driven insight on circular economy performance. *Technological Forecasting and Social Change, 168*, 120766. doi:https://doi.org/10.1016/j.techfore.2021.120766

Bousdekis, A., Lepenioti, K., Apostolou, D., & Mentzas, G. (2021). A review of data-driven decision-making methods for industry 4.0 maintenance applications. *Electronics, 10*(7), 828.

Faiz, T. (2019). *Multi-approaches on scrubbing data for medium-sized enterprises.* Paper presented at the 2019 International Conference on Digitization (ICD).

Fan, W., & Geerts, F. (2022). *Foundations of data quality management*: Springer Nature.

Foroozan, S., Murad, M. A., Sharef, N., & Latiff, A. A. (2015). *Improving sentiment classification accuracy of financial news using n-gram approach and feature weighting methods.* Paper presented at the 2015 2nd International Conference on Information Science and Security (ICISS).

Jiang, T., Huang, P., & Zhou, K. (2020). Achieving high data reliability at low scrubbing cost via failure-aware scrubbing. *Journal of Parallel and Distributed Computing, 144*, 220-229.

Kejriwal, M., & Miranker, D. P. (2015). *Sorted neighborhood for schema-free RDF data.* Paper presented at the European Semantic Web Conference.

Li, M., Xie, Q., & Ding, Q. (2015). *An improved data cleaning algorithm based on SNM.* Paper presented at the Cloud Computing and Security: First International Conference, ICCCS 2015, Nanjing, China, August 13-15, 2015. Revised Selected Papers 1.

Liu, F., & Panagiotakos, D. (2022). Real-world data: a brief review of the methods, applications, challenges and opportunities. *BMC Medical Research Methodology, 22*(1), 287. doi:10.1186/s12874-022-01768-6

Maharana, K., Mondal, S., & Nemade, B. (2022). A review: Data pre-processing and data augmentation techniques. *Global Transitions Proceedings, 3*(1), 91-99.

Mishra, R. S., Mehta, K., & Rasiwasia, N. (2021). Scalable approach for normalizing e-commerce text attributes (SANTA). *arXiv preprint arXiv:2106.09493*.

Obiedat, R., Qaddoura, R., Al-Zoubi, A. M., Al-Qaisi, L., Harfoushi, O., Alrefai, M., & Faris, H. (2022). Sentiment Analysis of Customers' Reviews Using a Hybrid Evolutionary SVM-Based Approach in an Imbalanced Data Distribution. *IEEE Access, 10*, 22260-22273. doi:10.1109/ACCESS.2022.3149482

Ridzuan, F., & Wan Zainon, W. M. N. (2019). A Review on Data Cleansing Methods for Big Data. *Procedia Computer Science, 161*, 731-738. doi:https://doi.org/10.1016/j.procs.2019.11.177

Schuster, D., van Zelst, S. J., & van der Aalst, W. M. P. (2022). Utilizing domain knowledge in data-driven process discovery: A literature review. *Computers in Industry, 137*, 103612. doi:https://doi.org/10.1016/j.compind.2022.103612

Singh, N., & Chaudhari, N. S. (2016). *N-gram approach for a URL similarity measure.* Paper presented at the 2016 1st India International Conference on Information Processing (IICIP).

Wei, H., Yu, J. X., & Lu, C. (2017). String similarity search: A hash-based approach. *IEEE Transactions on Knowledge and Data Engineering, 30*(1), 170-184.

Yu, M., Li, G., Deng, D., & Feng, J. (2016). String similarity search and join: a survey. *Frontiers of Computer Science, 10*, 399-417.