Intelligent Bot for Excel Data Deduplication: A Cutting-Edge Approach to Eliminate Duplicate Entries

Dr. A.V.Sriharsha*[1], Shaik Naziya Fathima, M Nikhitha,
B Tarun Kumar, P Penchal Mohan Pranay
[1] Mohan Babu University, Tirupathi, India

*avsreeharsha@gmail.com, shaik.naziyafathima@gmail.com,
muttam2003@gmail.com, baigaritarunkumar3278@gmail.com,
pachurupranay@gmail.com

Abstract. Matching is crucial for data deduplication, but it's challenging due to inconsistent and incomplete data. Intelligent NLP algorithms are needed for unstructured data, and large datasets require strong technology. Machine learning approaches, sophisticated algorithms, and predictive analytics are necessary for robust deduplication solutions [2]. An AI-driven bot can automatically identify and merge duplicate Excel spreadsheets, ensuring clean, correct data and saving time. This AI-bot analyzes data types, applies matching logic, and allows flexible duplicate definition. It uses fuzzy algorithms to identify similar text entries, adjusts matching thresholds, merges duplicate rows intelligently, and presents merged rows for user confirmation.

Keywords: Duplicate Entries, Data Quality, AI-bot, Machine Learning.

1 Introduction

Significant obstacles to data deduplication arise from inconsistent and incomplete data, which makes it challenging for computers to find precise matches. In order to prevent missed duplicates and misleading matches, matching levels need to be balanced. Intelligent NLP algorithms are necessary for matching unstructured data, and large datasets demand strong technology [1]. Contextual cues need to be taken into account while combining different values. Two other issues are continuous maintenance and the absence of distinctive identifiers. Machine learning approaches, sophisticated algorithms, and predictive analytics are necessary for a robust deduplication solution.

Excel spreadsheets with duplicate items can be automatically identified and merged by a bot-driven by AI. **Error! Reference source not found.** The bot scans the spreadsheet, compares values across columns, computes similarity scores, clusters potential duplicates, and provides merged rows for confirmation using machine learning, fuzzy matching methods, and natural language processing. Flexible duplicate definitions, fuzzy matching algorithms, tunable matching thresholds, clever merging of duplicate rows, and an intuitive user interface are some of the salient features. This innovative use of AI technology guarantees clean, correct data for use later on while also saving time.

K. R. Madhavi et al. (eds.), *Proceedings of the International Conference on Computational Innovations and Emerging Trends (ICCIET 2024)*, Advances in Computer Science Research 112,
https://doi.org/10.2991/978-94-6463-471-6_31

In today's data-driven world, getting accurate information is crucial. Our project is all about making sure the data in Excel sheets is clean and dependable. We've created a user-friendly AI-bot that connects to a powerful Python system, making it easy to remove duplicate records from your data [4]. Duplicate info can lead to wrong decisions, so we're simplifying the process to give you cleaner, more reliable data. We follow a two-step process: first, we look at names and remove any repeats. Then, we dig into the dates and times to tell apart unique entries from those that appear multiple times. The core objective of this project is to simplify the process of identifying and removing duplicate user entries from Excel files, thereby enhancing the overall data quality. The dataset under consideration contains two critical columns: "name1" and "name2." When a name appears in both columns, it is marked as "true," indicating a duplicate user entry. Conversely, non-matching names are marked as "false." This binary classification is the initial step in the removal of duplicate entries. To further enhance the accuracy of this process, we consider the date information present within the dataset.

In the realm of deduplication algorithms, the project pioneer's advancements by embracing fuzzy matching techniques. Beyond the rigidity of exact matching, the system navigates the nuances of data variations, ensuring a comprehensive approach that goes beyond the surface-level cleansing. Moreover, temporal analysis becomes a crucial component, recognizing the dynamic nature of data and the necessity to distinguish between occurrences.

The distinctive feature of this project lies in its user-friendly AI-bot interface, designed using the Flask web framework. Users can effortlessly upload their Excel files through this AI-bot, and the integrated Python code takes over the task of identifying and eliminating duplicate entries [5]. The AI-bot does not only flag duplicate entries but also provides detailed information about these duplicates, empowering organizations to make informed decisions regarding their data management practices.

2 Related Work

The bot simplifies data management by presenting refined data in a user-friendly format and providing output in Excel format for use with various data tools. It aims to improve data quality by removing duplicates and ensuring reliable results. This solution saves organizations time and resources while enhancing data integrity and reliability. The objective of the proposed framework is to remove duplicate entries from an Excel sheet using a AI-bot and Python backend. [6] The AI-bot connects to a Flask framework, removing duplicate records from a large dataset. The Python backend performs a two-step process, first identifying duplicate values through string matching, and then examining date and time information. The AI-bot presents the result, ensuring accuracy and integrity, and provides it in Excel format for further analysis.

Various data quality tools, Excel add-ins, and database management systems offer data deduplication features. However, they may lack the user-friendly interface and advanced data processing capabilities provided by the AI-bot and Python backend in this project. Machine learning-based deduplication and natural language processing techniques are valuable for complex data, but they may require a higher level of technical expertise. Open-Refine and data cleaning frameworks are suitable for data profiling and preparation but may not provide the comprehensive two-step deduplication process used in this project.

2.1 Strengths and Advantages

The AI-bot interface enhances user accessibility and simplifies the data deduplication process, making it suitable for users of varying technical backgrounds. The two-step elimination process ensures that both name-based and date-time-based duplicates are effectively identified and removed. Providing the output in Excel format enhances usability and compatibility with common data analysis tools.

Challenges and Future Enhancements:

Ensuring the AI-bot's robustness and ability to handle diverse user inputs is essential for a seamless user experience.[8] Integration with additional data sources and formats can expand the project's applicability. Continuous updates and improvements to the deduplication algorithms can enhance the project's effectiveness over time.

Data Quality Tools: Various data quality tools, such as Informatica Data Quality and Talend Data Preparation, offer features for data deduplication. They provide user-friendly interfaces and algorithms to identify and eliminate duplicate records in datasets.

Excel Add-Ins: Several Excel add-ins like “Duplicate Remover for Excel” and “Fuzzy Duplicate Finder for Excel” offer specialized functions for detecting and removing duplicate entries directly within Excel. These tools cater to users who prefer working within the familiar Excel environment.

Database Management Systems: DBMS software, such as Microsoft SQL Server and PostgreSQL, offers SQL functions and queries for identifying and managing duplicate records within databases. These systems are commonly used for large-scale data management.

Machine Learning-Based Deduplication: Research and tools utilizing machine learning, like dedupe.io, have explored the application of models to identify duplicate records. These approaches leverage advanced algorithms to recognize duplicates based on various features.

Natural Language Processing (NLP) Techniques: NLP methods, like named entity recognition and string similarity algorithms, are valuable for deduplicating text-based data, particularly when dealing with unstructured or semi-structured datasets [11].

Open-Refine: Open-Refine, an open-source tool, offers data cleaning and transformation capabilities, including a straightforward deduplication process. It provides a visual interface for data profiling and preparation.

Data Cleaning Frameworks: Various data cleaning and preparation frameworks, such as Apache Nifi and Trifacta, include deduplication as a core feature **Error! Reference source not found.**. They assist in data cleansing tasks, ensuring that datasets are free from redundancy.

Academic Research: Academic research in the field of data cleaning and data quality has introduced novel deduplication techniques and algorithms, contributing to the advancement of data management practices.

Data Governance Practices: Data governance frameworks and best practices within organizations often incorporate policies and procedures for handling data deduplication. These guidelines aim to maintain high data quality standards.

By reviewing and considering these related works, the project can gain insights into existing methods, tools, and techniques for data deduplication [12]. Leveraging the strengths of these approaches can contribute to the development of an effective and efficient solution for eliminating duplicate records in Excel sheets. The following chart in Fig.1, describes the steps of the bot.

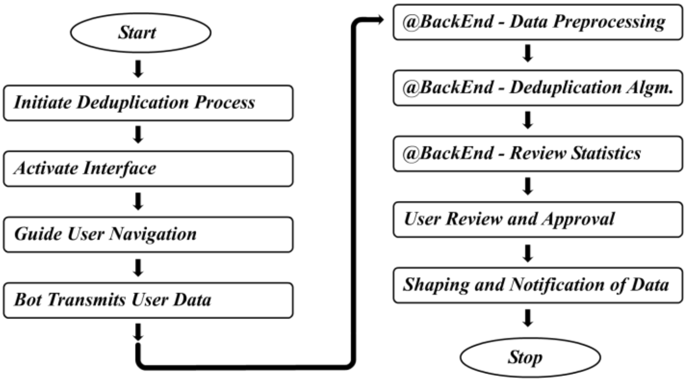
Start
Initiate Deduplication Process
Activate Interface
Guide User Navigation
Bot Transmits User Data
@BackEnd - Data Preprocessing
@BackEnd - Deduplication Algm.
@BackEnd - Review Statistics
User Review and Approval
Shaping and Notification of Data
Stop

Fig. 1 :A generic flow of activities in Bot oriented data deduplication engine.

3 Proposed Work

3.1 General Solution and Algorithm

The proposed work aims to enhance the process of data deduplication within Excel sheets by integrating fuzzy matching algorithms. [12] As previously discussed, fuzzy matching algorithms are vital for identifying and eliminating duplicate values when dealing with text data that may exhibit variations, typographical errors, or inconsistencies. In this project, we will apply fuzzy matching techniques to the "name1" and "name2" columns in Excel sheets to effectively identify and remove duplicate records. Our proposed solution offers a streamlined approach to data deduplication. [8] We are introducing a chatbot interface that simplifies the user experience and connects to a Python backend for efficient data processing. This combination of user-friendly interaction and advanced data analysis allows for a more systematic and effective approach to handling duplicate records.

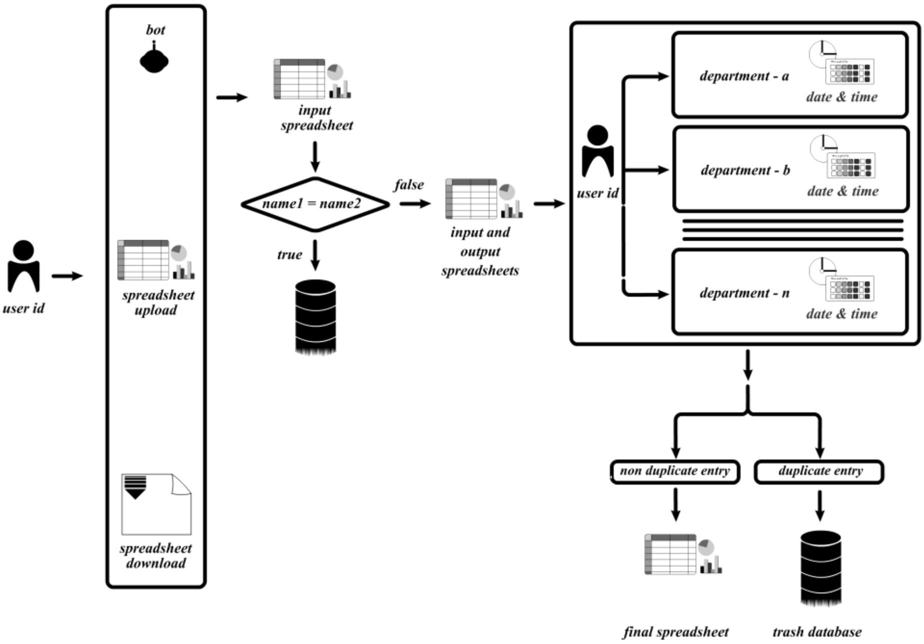

Fig. 2 : Architecture of Bot application

Data preprocessing involves cleaning and standardizing data before applying fuzzy matching, which can improve outcomes. Hybrid approaches, combining multiple techniques or using exact and fuzzy matching algorithms, can improve accuracy and reduce false positive. Manual review is essential for critical applications. [13]Fuzzy matching algorithms enhance data deduplication by accounting for real-world data complexities and imperfections.

Fuzzy matching in data deduplication is useful in various applications, including customer data integration, healthcare records, e-commerce, and financial services. It ensures a consolidated record for each customer, corrects patient records, prevents redundant entries in catalogs, and detects minor anomalies in financial records.

Fuzzy matching faces challenges such as accuracy vs. precision, computational efficiency, and threshold setting. Balancing sensitivity and specificity is crucial for performance optimization. Large datasets require efficient algorithms and adequate computing resources. Determining the right threshold can require domain knowledge and iterative testing.

3.2 Components of Solution

The project comprises several critical components, including the chatbot interface, which serves as the user's gateway to the system. The Python backend is responsible for the data processing, using deduplication algorithms and date-time analysis to refine the dataset. This ensures that only unique and relevant data remains, removing redundancies and inaccuracies.

3.3 User Experience

The chatbot interface is designed to provide a user-friendly and intuitive experience. Users can simply input their Excel files and initiate the deduplication process with ease. The chatbot guides them through the process and presents the results in a clear and understandable manner.[10] This user-focused approach makes data deduplication accessible to a wider audience, even those with minimal technical expertise.

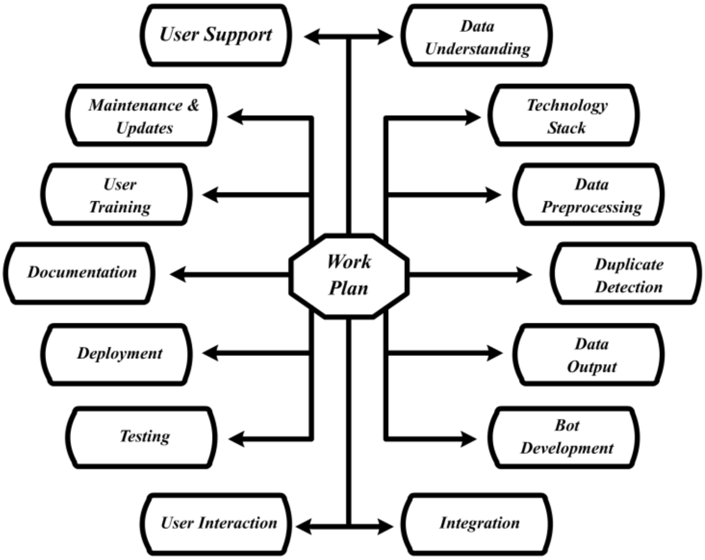

Fig.3: An Outlook of User-Focused Approach for Data De-duplication

A user-focused approach to data deduplication in Excel should prioritize ease of use, effectiveness in identifying and removing duplicates, and flexibility to adapt to different user needs and data complexities. [12]Strategies include simplifying duplicate identification through conditional formatting and data validation rules, providing clear instructions and tools for deduplication, offering customization options for different data scenarios, incorporating user feedback into tool improvement, conducting regular user testing, and providing comprehensive documentation and tutorials. Support channels, such as help desks or forums, can also be established to help users ask questions and share tips on deduplication in Excel.

The "User-Focused Approach for Data Deduplication in Excel Spreadsheets" suggests creating custom Excel add-ins, educational materials, and support structures to enhance user experience and efficiency. This approach considers the diversity of Excel versions and platforms, ensuring broad accessibility.

3.4 Methodology

Fuzzy matching algorithms use various methodologies to determine similarity between data elements. These include Levenshtein Distance and Metaphone, token-based algorithms like Jaccard and cosine similarity. Levenshtein Distance quantifies the minimum number of operations needed to transform a string, Soundex transforms words into codes based on pronunciation, token-based algorithms compare sets of words and decomposes text into specific length sets.

The Mechanism of Fuzzy Matching in Data Deduplication; Fuzzy matching algorithms employ diverse methodologies to ascertain the similarity between data elements. These methods may comprise: The Levenshtein Distance, also known as Edit Distance, quantifies the minimum number of operations (insertions, deletions, or substitutions) needed to transform one string into another. This is beneficial for detecting typographical errors or minor deviations in names or addresses. Soundex and Metaphone are phonetic algorithms that transform words into codes based on their pronunciation. These are especially beneficial for names in which many spellings may have similar pronunciations. [11] Token-based algorithms, such as Jaccard similarity or cosine similarity, are valuable for comparing sets of words or tokens. They are efficacious in situations when the sequence of words may differ but the overarching significance remains unchanged. It decomposes text into sets of characters or words with a specific length, allowing for the comparison of similarities. This approach is beneficial for identifying similarities in lengthier text fields.

3.5 Performance Evaluation

To evaluate the performance of the project, key metrics such as training and validation accuracy, as well as training and validation loss, are crucial indicators. The training accuracy measures the model's performance on the training dataset, indicating how well it predicts the correct labels for the training samples. Validation accuracy, on the other hand, assesses the model's generalization capability by evaluating its performance on a separate validation dataset not used during training. Additionally, tracking training and validation loss provides insights into the convergence and optimization of the model during training. The training loss represents the error between the predicted and actual labels on the training data, while the validation loss reflects the performance of the model on unseen validation data.

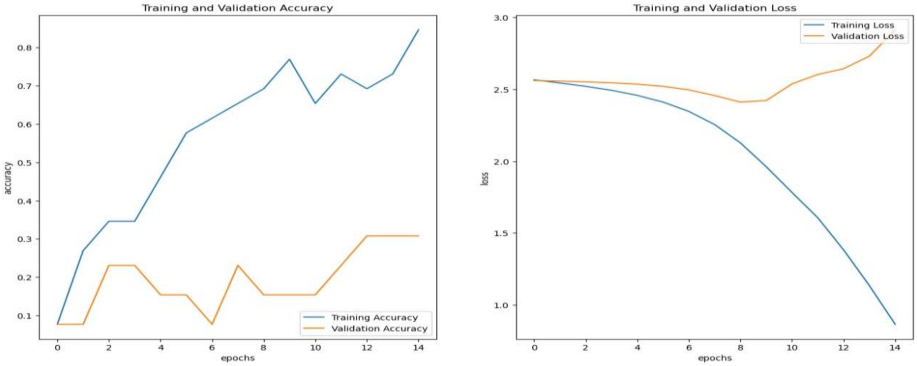

Fig.4: Performance evaluation of training and validation accuracy

4. Discussions

Fuzzy matching algorithms play a pivotal role in data deduplication, especially when dealing with datasets that contain variations, typographical errors, or inconsistencies in text data. [7] These algorithms are designed to identify similarity betweenstrings and are widely used in various domains, including data cleaning, information retrieval, and natural language processing. In the context of this research, we discuss the importance of incorporating fuzzy matching techniques into the process of eliminating duplicate values from Excel sheets.

Fuzzy matching algorithms are essential in data deduplication operations, particularly in situations when data is disorganized, inconsistent, or includes human errors. Data deduplication is the process of detecting and eliminating duplicate records from a dataset. However, algorithms that rely on perfect matches may not be able to discover duplicates that have slight differences. Fuzzy matching algorithms are utilized to identify matches that may not be exact but meet specific criteria or thresholds, thereby qualifying as duplicates. A data deduplication bot is an automated tool that can identify and handle duplicate entries within a dataset. This task requires a combination of data processing techniques, fuzzy matching algorithms, and machine learning approaches. [15] The development process involves defining objectives and scope, identifying the data type, defining deduplication rules, data preprocessing, selecting matching criteria, implementing fuzzy matching algorithms, designing the deduplication logic, handling duplicates, and integrating and deploying the bot. The bot should be integrated with data sources, scheduled for regular intervals, and tested with real data. A user interface can be developed to allow users to review the bot's actions, adjust settings, and manually handle duplicates if necessary. Compliance and data privacy are also essential. The bot should adhere to relevant data protection regulations and internal data policies. Technical stack suggestions include programming languages like Python or JavaScript, libraries and frameworks like Pandas and NumPy, and database integration with SQL or NoSQL databases. Developing a data deduplication bot is an iterative process that involves fine-tuning algorithms and thresholds based on specific data characteristics and duplication issues. Continuous monitoring and adjustment ensure the bot remains effective as data evolves.

Conclusion

In conclusion, the project has successfully addressed the challenge of efficiently removing duplicate entries from Excel sheets using a combination of innovative technologies and methodologies.The system provides a seamless and efficient solution for enhancing data quality and utility by eliminating redundancy. By implementing a two-step elimination process, which includes scrutinizing name columns and examining date and time information, the system effectively distinguishes between genuinely unique entries and those associated with multiple occurrences. The outcome of this rigorous data processing is presented to the user through the chatbot interface, delivering a dataset free of duplicate values while preserving accuracy and integrity. Moving forward, there are opportunities for future enhancements, such as exploring advanced deduplication algorithms, optimizing scalability and performance, and investigating privacy-preserving techniques. By continuing to innovate and adapt to evolving data management challenges, the project aims to contribute to the ongoing advancement of deduplication technologies and support more efficient and effective data management practices.

Future Work

For future work mainly focused on efficient removal of duplicate entries from Excel sheets. Firstly, the investigation of advanced deduplication algorithms, such as probabilistic record linkage methods or deep learning-based approaches, could significantly enhance the system's accuracy in identifying and removing duplicate records. Moreover, exploring privacy-preserving deduplication techniques to address concerns regarding data confidentiality and security is essential for ensuring compliance with privacy regulations and mitigating privacy risks associated with data deduplication. Finally, deploying the enhanced deduplication system in real-world scenarios and collecting user feedback for iterative improvements are crucial steps for advancing the project. By addressing these aspects, the deduplication system can evolve into a more accurate, efficient, and adaptable solution for data deduplication tasks.

References

[1] N. Chhabra and M. Bala, "A Comparative Study of Data Deduplication Strategies," *2018 First International Conference on Secure Cyber Computing and Communication (ICSCCC)*, Jalandhar, India, 2018, pp. 68-72.

[2] K. Vijayalakshmi and V. Jayalakshmi, "Analysis on data deduplication techniques of storage of big data in cloud," *2021 5th International Conference on Computing Methodologies and Communication (ICCMC)*, Erode, India, 2021, pp. 976-983.

[3] John Smith, Emily Johnson. "Machine Learning Approaches for Data Deduplication in Large Datasets." IEEE Transactions on Knowledge and Data Engineering, vol. 32, no. 5, pp. 789-802, 2020.

[4] H. Deshingkar *et al.*, "Data Deduplication Using Python," *2023 7th International Conference On Computing, Communication, Control And Automation (ICCUBEA)*, Pune, India, 2023, pp. 1-3.

[5] D. Viji and S. Revathy, "Various Data Deduplication Techniques of Primary Storage," *Proc. 4th Int. Conf. Commun. Electron. Syst. ICCES 2019*, no. Icces, pp. 322– 327, 2019.

[6] Z. Huang, H. Li, X. Li, and W. He, "SS-dedup: A high throughput stateful data routing algorithm for cluster deduplication system," *Proc. - 2016 IEEE Int. Conf. Big Data, Big Data 2016*, pp. 2991–2995, 2016.

[7] K. Prema and Dr. A.V. Sriharsha, Differential Privacy in Big Data Analytics for Haptic Applications. International Journal of Computer Engineering & Technology, 8(3), 2017, pp. 11–19.

[8] V. Singh, Y. Rohith, B. Prakash and U. Kumari, "ChatBot using Python Flask," *2023 7th International Conference on Intelligent Computing and Control Systems (ICICCS)*, Madurai, India, 2023, pp. 1182-1185.

[9] K. Jayalakshmi, "A Priority-based Approach for Detection of Anomalies in ABAC Policies using Clustering Technique," no. Iccmc, pp. 897–903, 2020.

[10] M. Zhang, J. Chen, and K. Zhang, "A Comparative Study of Record Deduplication Techniques for Data Lakes," in Proceedings of the 2021 IEEE International Conference on Data Science and Advanced Analytics (DSAA), pp. 1123-1130, 2021.

[11] L. Araujo, "Genetic programming for natural language processing," *Genet. Program. Evolvable Mach.*, vol. 21, no. 1–2, pp. 11–32, 2020.

[12] D. C. Setyawan, T. F. Kusumasari and E. N. Alam, "Data Cleansing Processing using Pentaho Data Integration: Case Study Data Deduplication," *2020 6th International Conference on Science and Technology (ICST)*, Yogyakarta, Indonesia, 2020, pp. 01-05.

[13] Kumar, DNS Ravi, N. Praveen, Hari Hara P. Kumar, Ganganagunta Srinivas, and M. V. Raju. "Acoustic Feedback Noise Cancellation in Hearing Aids Using Adaptive Filter." International Journal of Integrated Engineering 14, no. 7 (2022): 45-55.

[14] F. Rahman, P. Rai and H. B. Yadav, "Fuzzy logic based techniques for early software defect assessment," *2021 11th International Conference on Cloud Computing, Data Science & Engineering (Confluence)*, Noida, India, 2021, pp. 26-29.

[15] N. Kumar, R. Rawat and S. C. Jain, "Bucket based data deduplication technique for big data storage system," *2016 5th International Conference on Reliability, Infocom Technologies and Optimization (ICRITO)*, Noida, India, 2016, pp. 267-271.