Journal of Engineering and Technology for Industrial Applications

ITEGAM-JETIA

Manaus, v.12 n.58, p.1020-1029. March/April, 2026.
DOI: https://doi.org/10.5935/jetia.v12i58.3239

ISSN ONLINE: 2447-0228

RESEARCH ARTICLE OPEN ACCESS

A BOTTOM-UP K-ANONYMIZATION APPROACH FOR BIG DATA PUBLISHING

Abderrahmane Saidi[1], Salheddine Kabou*[2] , Imad eddine Kimi[3] and Laid Gasmi[4]

[1, 2, 3] Higher Normal School of Bechar, Algeria
[4,] Ahmed Draia University -Adrar, Algeria.

[1]https://orcid.org/0009-0007-0611-0445, [2] http://orcid.org/0000-0002-1423-7215
[3]http://orcid.org/0009-0005-6997-9194, [4]https://orcid.org/0000-0001-8925-0089

Email: *kabou.salheddine@ensbechar.dz

ARTICLE INFO

Article History
Received: January 5, 2026
Reviewed: February 5, 2026
Accepted: March 10, 2026
Published: April 30, 2026

Keywords:
Big Data,
K-Anonymity,
L-Diversity,
Bottom-Up,
Top-Down Specialization.

ABSTRACT

As governments and other organizations share larger datasets, keeping individual information private has become increasingly difficult to solve. When publishing the data, data anonymization models like k-anonymity and l-diversity are employed to ensure the trade-off between privacy and data utility. This paper presents a method called Bottom-Up k-anonymization (BU-K), implemented on Apache Spark. It improves efficiency by applying the Bottom-Up Generalization (BUG) approach. BU-KC performs better than Top-Down Specialization (TDS) in terms of scalability, and data privacy, while still keeping the data useful. Moreover, using Apache Spark's distributed computing architecture significantly improves processing time compared to traditional MapReduce approaches. This work fills a gap in distributed anonymization on Spark by offering a new, efficient, and scalable solution.

I. INTRODUCTION

In recent years, many organizations have required the web-based exchange of huge volumes of data with research institutes in order to undergo analysis and identify subtle patterns. A significant difficulty in data publishing is protecting people's privacy by anonymizing information that could expose their identities. The key lies in developing systems that effectively balance privacy protection and data usefulness [1]. The field of study that combines all of these methods is known as Privacy Preserving Data Publishing (PPDP) [4]-[2]. As data volumes have expanded and the big data paradigm, defined by the 42 Vs, has emerged, various privacy models have been extremely prominent in this area. Among these techniques, k-anonymity [3] and l-diversity [4] are widely employed. Generalization processes are commonly used in data anonymization, where original values are replaced with generalized ones using a taxonomy tree to maintain semantic compatibility between the old and new values [5].There are generally two main ways to navigate the taxonomy tree: Top-Down Specialization (TDS) starts from the root and moves down, while Bottom-Up Generalization (BUG) begins at the leaves and moves up.

Unlike the top-down approach, we use Bottom-Up Generalization (BUG), a more efficient strategy that preserves data quality [6]. As data volumes approach zettabytes, improving processing time becomes crucial for efficient data interchange and publication. Traditional centralized solutions fail to manage huge volumes of data, resulting in inefficiencies in time processing, scalability, and overall performance [7]. To address these difficulties, distributed computing frameworks such as MapReduce and Apache Spark have been developed. Among these, Apache Spark stands out as an open-source, scalable platform capable of effectively processing massive datasets. Its ability to analyze large amounts of data quickly and efficiently makes it a suitable tool for modern data processing tasks [8]. Programs running on the Spark platform can achieve significantly faster processing speeds up to 100 times faster in memory and 10 times faster on disk compared to MapReduce. Consequently, Spark has emerged as a highly efficient ecosystem for performing the anonymization process on large-scale data. By leveraging Spark's capabilities, the anonymization process can be executed more efficiently, enabling faster processing and improved scalability [9].

The Bottom-Up Generalization (BUG) approach, proposed in [10] as a centralized way to support k-anonymity model, was later improved by [11] to work with MapReduce Framework, making it faster to process. Since then, a lot of research has explored using the k-anonymity model with MapReduce. However, to the best of our knowledge, no one has tried to fully implement a Bottom-Up anonymization approach on Apache Spark. Our work aims to fill this gap by developing a distributed Bottom-Up k-anonymity (BU-K) framework tailored for Spark. This approach seeks to enhance both data utility and processing efficiency, leveraging Spark's parallel processing and scalability features to achieve significant improvements in execution speed and data quality preservation during anonymization. The remainder of this paper is organized as follows: Section 2 reviews the existing literature on related works. In Section 3, we delve into the core concepts and essential components of the proposed approach. Section 4 presents and discusses the results from the evaluation of our method. Finally, Section 5 offers conclusions and outlines potential directions for future research.

II. RELATED WORKS

Privacy-Preserving Big Data Publishing (PPBDP) is a critical research field focused on tackling the challenge of releasing large-scale datasets while ensuring an optimal balance between data privacy and utility [12]. In the context of big data, several privacy models, such as k anonymity, have been widely adopted. Section 2.1 explores various centralized algorithms, most of which focus on implementing the k-anonymity model. Section 2.2 presents a range of distributed big data algorithms that perform anonymization using either top-down or bottom-up generalization techniques.

II.1 CENTRALIZED APPROACHES

LeFevre et al [13] introduced the Mondrian algorithm, which employs a top-down generalization approach for data partitioning. Initially, all data instances are grouped into a single partition. The algorithm then recursively divides the data instances into sub-partitions using a top-down approach until the resulting partitions satisfy the k-anonymity requirements. At each step, the algorithm selects the widest dimension of the data as the cut dimension and uses the median of the data values in this dimension as the split point for partitioning.

Yaseen et al. [14] proposed three novel generalization methods to overcome the limitations of traditional techniques: conventional generalization hierarchies, divisors generalization hierarchies, and cardinality-based generalization hierarchies. Each method prioritizes data utility and partitions the attribute domain space in a distinct way. Their performance is evaluated based on the discernibility penalty, distortion ratio, and the number of nodes at each level.

In this paper [15], a new data anonymization algorithm that enhances users' community privacy is presented. The algorithm assesses the vulnerability of each attribute in a user's dataset to effectively protect community privacy. It conducts adaptive data generalization while simultaneously assessing attribute susceptibility and entropy.

Basapur et al [16] introduced a novel data generalization algorithm aimed at improving group privacy and data utility. This algorithm evaluates attribute susceptibility, the frequency of sensitive attributes, and their interrelationships to create equivalence classes that satisfy both k-anonymity and β-likeness. The approach incorporates these factors into the data anonymization process and is tested using Apache Spark

Recently, Torra and Navarro [17] proposed a new data publishing for minimizing the attribute disclosure risk in k-anonymity model. To avoid Internal and external attacks, the solution is to apply well-known codes to identify sensitive cells in tabular dataset by using p-sensitivity and p-diversity models.

In order to avoid the danger of skewness and similarity attacks, Su et al. [18] presented a k-anonymity mechanism based on clustering for multi-dimensional data, combined with t-closeness privacy model. Based on an enhanced African vultures optimization, the suggested approach adheres to extremely accurate clustering of the multi-dimensional dataset and can offer the ideal solution with multiples dimensions.

II.2 DECENTRALIZED APPROACHES

Sopaoglu and Abul [19] developed a Top-Down Specialization (TDS) anonymization solution designed specifically for the Apache Spark platform. This method assesses both numerical and category attributes by iteratively generalizing the original data, beginning with the most general node and using the k-anonymity model.

Zakerzadeh et al. [20] enhanced the centralized Top-Down Mondrian technique for massive data applications by using the MapReduce framework. Reducing the algorithm's processing time across the Mapper, Combiner, and Reducer computational nodes is the aim of this work.

In a related study, Ashkouti et al. [21] enhanced earlier work on the Apache Spark Framework by applying a multidimensional Top-Down Mondrian approach. The primary difference between the two approaches resides in selecting the axis dimension creterion to which the partitioning is carried out. While [20] utilized the domain width as the primary criterion for axis dimension selection, the improved study diverges by prioritizing the dispersion of attribute values over the domain. This research aims to enhance time efficiency through the implementation of RDD programming, while simultaneously improving data privacy by adopting the l-diversity model in lieu of k-anonymity. Jain et al [22] developed a model that introduces an additional layer between the Hadoop Distributed File System (HDFS) and MapReduce to enhance data sharing for mining operations. This scalable SMR model performs as part of the Map and Reduce phases and aims to balance privacy with utility through lightweight encryption, randomization, and perturbation techniques. In the presence of big data, the centralized traditional BUG techniques are not suitable for adhering privacy needs. This motivated researchers to propose parallel bottom-up approaches to address limitations brought about by centralized BUG.

In [23] [11], a scalable advanced Bottom up generalization method is proposed to meet the k-anonymity model using MapReduce platform on Cloud. The process involves two main steps: first, the original dataset is divided into multiple tables, and each partition is anonymized in parallel to generate intermediate results. In the second step, these intermediate results are grouped, and an additional anonymization operation is performed to ensure compliance with the privacy model. In terms of scalability, the authors demonstrate that the parallel BUG approach is more efficient than the top down specialization method. The existing literature has predominantly focused on implementing top-down k-anonymization approaches, either within the MapReduce environment or on Apache Spark.

In contrast, bottom-up k anonymization methods have been primarily applied within MapReduce frameworks, with no prior implementations in the Apache Spark ecosystem. Our work addresses this gap by pioneering the implementation of a bottom-up k-anonymization approach specifically on Apache Spark. This advancement is significant because, as established by various studies [6], the bottom-up approach is generally more effective than the top-down approach in terms of efficiency and privacy preservation. By leveraging Apache Spark's advanced processing capabilities, our implementation aims to deliver superior performance, enhancing both data utility and processing speed compared to the existing top-down methodologies. This contribution represents a notable advancement in the field of privacy-preserving data publishing, setting a new benchmark for efficiency and scalability in large scale data anonymization.

III. PROPOSED METHOD

III.1 PRELIMINAIRES

In the context of privacy-preserving big data publishing, data is typically structured in a table with different types of attributes, classified into three main categories:(i) Explicit Identifiers (EI): These are attributes that immediately reveal an individual's identify, such as a name. (ii) Quasi-Identifiers (QI): While these attributes are not individually revealing, when paired with other data, they can be utilized to identify individuals, such as zip code. (iii) Sensitive Identifiers (SI): These attributes include confidential information, such as medical conditions [22]. It’s important to note that EI and QI traits might be numerical or categorical, whereas SI attributes are always categorical [5]. To achieve data anonymization, it is crucial to convert the table into generalized groups known as Equivalence Classes (EC) or QI-groups. The k anonymity model requires that each EC contains at least k records, ensuring that each individual cannot be uniquely identified within the group. he l-diversity model requires that each EC contains at least l distinct sensitive values, thereby enhancing individual privacy while preserving data utility.

III.2 THE PHASES OF THE PROPOSED METHOD

Figure 1 provides a visual representation of the sequential phases involved in the implementation of our distributed k concealment model. The process is executed using a series of RDDs, which are manipulated through transformations and actions in the Apache Spark library.

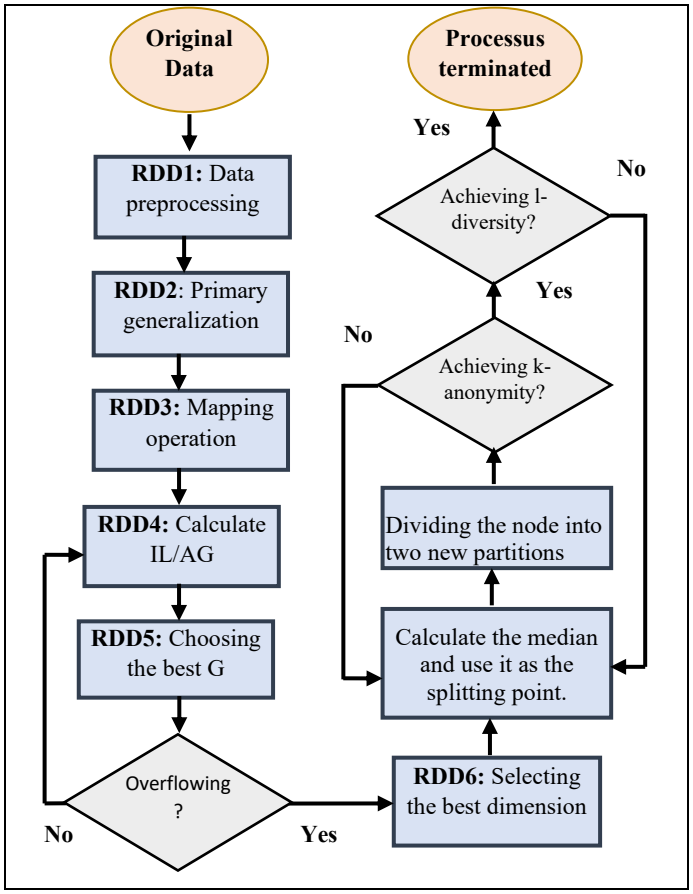

Figure 1: Summary of the proposed approach.
Source: Authors, (2026).

III.2.1 Data preprocessing

The dataset obtained from the data holder contains a significant number of missing and duplicate entries. The primary objective of this phase is to preprocess the data to ensure it is suitable for the subsequent anonymization process. This entails identifying and removing erroneous records, followed by distributing the cleaned data across multiple worker nodes for further processing.

III.2.2 Primary generalization

In this phase, the worker nodes concurrently anonymize the data in parallel by applying the primary generalization operation. The goal is to achieve compliance with the k-anonymity model, resulting in an intermediate anonymized dataset.

III.2.3 Mapping operation

In this phase, the mapping operation is performed to associate the <key, value> pairs for each data record, as described in Algorithm 1. In the preceding phase, a collection of equivalence classes was generated based on the k-anonymity model. Here, the "key" represents the set of distinct sensitive values within the equivalence class to which the record belongs (labels), while the "value" corresponds to the attribute values of the individual data record

Algorithm 1: Mapping operation.

Input : RDD2, ECs
Output : (key, value) pairs as RDD3
    Key = [ ] / list of SI of each EC
    Values = [ ]
    for each EC in data records **do**
        Value = Value.append (QI)
        Key = All_SI (value)
end for

Source: Authors, (2026).

III.2.4 Generalization process

III.2.4.1 Calculate Score (G)

Our Bottom-Up Generalization approach traverses the generalization hierarchies from the leaves to the root. It iteratively performs generalizations deemed optimal in the sense that they best preserve the quality of the information while achieving the desired anonymization model. Each optimal generalization is selected from a set of candidate generalizations. A generalization is considered optimal if it yields the highest score, determined by applying the IL/AG (Information Loss/Anonymity Gain) trade-off metric, which is used to assess the loss of information concerning classification and the security gain associated with anonymization. The formula (01) used to calculate the score of a generlization G, denoted as Score(G), is as follows:

$$\text{Score(G)} = 1 + \frac{IL\ (G)}{AG\ (G)}, \qquad \text{AG} \neq 0 \qquad (01)$$
$$IL\ (G)$$

InformationLoss IL(G) corresponds to the loss of information following the generalization G (equation 2). It is defined as follows:

$$IL\ (G) = Entropy\ (Rg) - \sum_{di} \frac{|Rdi|}{|Rg|}\ Entropy\ (Rdi) \qquad (02)$$

Where Rg (respectively Rdi) represents the set of records containing the value g (respectively the value di). Entropy(Rx), where x ∈ {g,di}, corresponds to the entropy of the set Rx. It is calculated as follows:

$$Entropy\ (Rx) = -\sum_{cls} \frac{freq(Rx, cls)}{|Rx|} \times \log_2 \frac{freq(Rx, cls)}{|Rx|} \qquad (03)$$

Where freq (Rx,cls) represents the percentage of individuals belonging to the class labeled cls within Rx. AnonymityGain AG(G) corresponds to the anonymity gain that could result from the performing of generalization G. Intuitively, this measure is calculated by comparing the degree of anonymity of a table before and after the application of generalization G. Formally, it is equivalent to:

$$AG\ (G) = Anonymity\ (T, after\ G) - Anonymity\ (T, befor\ G) \qquad (04)$$

Anonymity (T, after G) corresponds to the size of the smallest equivalence class (the equivalence class containing the fewest individuals sharing the same QI) of T after the application of G, and Anonymity (T, before G) corresponds to the size of the smallest equivalence class of T before the application of G.

III.2.4.2 Choosing the optimal G

At each iteration, the generalization process algorithm selects, from a set of candidate generalizations, one that is considered optimal based on the score(G) metric (Algorithm 2). The selected generalization is then applied to the table undergoing anonymization

Algorithm 2: Generalization process.

Input : RDD3, ECs
Output : (key, value) pairs as RDD4
  while *Node* is not overlowed **do**
  for all generalization G **do**
    compute *ScoreG*
  end for
  determine the optimal generalization G_{opt}
  generalize ECs by G_{opt}
  end while

Source: Authors, (2026).

As a result, the outcome of this phase is recorded in RDD4 as <Key, G_{opt}>. Here, “Key” denotes the set of distinct Sensitive Identifiers within the equivalence class, while “G_{opt}” represents the optimal generalization determined by the highest score.

III.2.5 Split process

III.2.5.1 Selecting the best dimension

In this phase, the emphasis is on determining the most suitable axis dimension for splitting nodes. Given that data records are distributed across multiple worker nodes, this selection must occur in parallel, with each node independently determining its split axis. This approach leverages the criteria outlined in the Mondrian algorithm, aiming to maximize data partitioning while adhering to privacy constraints. The axis dimension selected is the one associated with the widest domain attribute, as a broader range improves the chances of identifying an optimal split point and enhances the potential for effective data partitioning. Consequently, the outcome of this phase is stored in RDD6 as <Key, Att>, where "Key" represents the set of distinct Sensitive Identifiers within the equivalence class, and "Att" indicates the attribute chosen for splitting.

III.2.5.2 Node splitting

In this phase, node splitting is performed on the overloaded node (Algorithm 3). The main goal is to find an appropriate partition for a node with an excess of data records DR while ensuring compliance with privacy constraints. To facilitate the node splitting process, the following steps are undertaken:

Step 1: The partitions determination: The algorithm focuses on splitting overloaded leaf nodes containing bounds of equivalence classes initially inserted for k-anonymity tables. The aim of the split is not only to divide the node but also to minimize overlap between resulting partitions, thereby enhancing data utility. To achieve this, the approach involves separating the two partitions around the median value of the split attribute. This step is illustrated in figure 2 as an example. In the chosen attribute "Age," the median value is identified as the partitioning point. This result in two distinct partitions: the first containing records with ages less than the median, and the second containing records with ages greater than the median.

Step 2: Checking the Privacy Requirement: In this step, the data is partitioned as extensively as possible while still meeting the l-diversity model, in order to minimize information loss. The key criterion here is to classify all data records such that each partition contains a minimum of k elements, and each equivalence class (EC) contains at least l distinct sensitive values. It's important to note that the process of partitions determination is repeated until both partitions satisfy the l-diversity model.

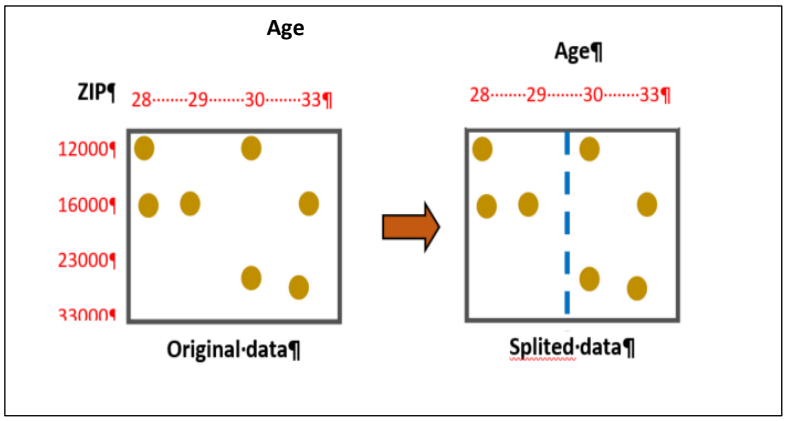

Figure 2: The partitions determination.
Source: Authors, (2026).

Algorithm 3: Node splitting.

Input : RDD5,
Output : Part 1 and Part 2
Step 1: The partitions determination
for each best attribute Att **do**
compute median ***m*** of Att
Selecting ***m*** as splitting point
Dividing the overlowed node from the splitting point into two new partitions
Part 1 containing DR smaller than **m** and part2 containing DR greater than **m**
end for
Step 2: Checking the privacy on static data
if (k<count_Partj) and (l<= labels) **then**
Generating final node splitting
else
➔Steps 2 is called

Source: Authors, (2026).

IV. EXPERIMENTAL EVALUATION

IV.1 EXPERIMENTAL CONFIGURATION AND DATASET

In our experiments, we employed the "Adults" dataset from the UCI Machine Learning Repository, which contains 48,842 records and 14 attributes. For quasi-identification purposes, we focused on the attributes of age, occupation, relationship, education, and workclass. The sensitive attribute in our analysis was disease. The taxonomy structures used for both quasi-identifiers and the sensitive attribute align with those outlined by [24]. To accommodate the large-scale data requirements of our implementation, we expanded the dataset by randomly augmenting it based on the selected quasi-identifiers, thereby generating additional records. A comprehensive overview of the resulting "Adults" dataset is presented in Table 1

Table 1: Adult producing table.

Dataset	Number of data records	Size
Adult 48k	48842	1.5 MB
Adult 500K	500 Thousand	15 MB
Adult 1M	1 Million	31 MB
Adult 10 M	10 Million	295 MB

Source: Authors, (2026).

IV.2 INFORMATION LOSS

The primary goal of a data publisher is not only to safeguard data privacy but also to preserve its utility. To quantify the extent of information loss caused by the generalization process, a parameter known as information loss is utilized. Several metrics can be employed to compute this parameter; however, in the context of big data, the Normalized Cardinality Penalty (NCP) metric is regarded as the most appropriate, surpassing other measures proposed by researchers such [25] and [26]. Equation (5) presents the formula for calculating the NCP. In the numerator, the difference between the upper and lower domains of attribute m in data record n is subtracted after the generalization operation. The denominator represents the difference between the maximum and minimum values of attribute m across the entire dataset. Here, x and y represent the number of data records and attributes, respectively.

$$NCP = \sum_{n=1}^{x} \sum_{m=1}^{y} \frac{|upper_{nm} - lower_{nm}|}{x.y\,|Max_m - Min_m|} \quad (05)$$

The synthetic Adult dataset was utilized to evaluate the performance of the Mondrian, top-down k-anonymization (TD-K), and bottom-up k-anonymization (BU-K) methods in terms of information loss. Figure 3 presents a visual depiction of the supervised evaluations performed on these methods. The assessments were conducted using a range of values for both k and l, with k varying from 30 to 160, and l ranging from 3 to 6. As depicted in figure 3, the quantity of NCP increases along with the variables k and l for the three algorithms. This increase is attributed to the growing number of data records in each equivalence class. It is also noteworthy that our BU-K proves to be more effective than the Top-down methods in several ways, particularly in minimizing information loss. It operates with finer granularity by starting with the most specific data and only generalizing when necessary, preserving as much detail as possible. Unlike the Top-down approaches, which applie broader generalizations from the outset, BU-K progressively optimizes the generalization process, minimizing unnecessary loss. Additionally, it better preserves data diversity by only anonymizing the data groups that require it, whereas Top-down techniques generalize larger portions of the dataset, reducing variability. Our method also adapts more effectively to heterogeneous datasets, maintaining the unique characteristics of subgroups, while other methods imposes uniform generalization. Finally, experimental results demonstrate that the Bottom-up approach achieves lower information loss, as evidenced by metrics like the Normalized Cardinality Penalty (NCP), which confirm its superiority in preserving data utility.

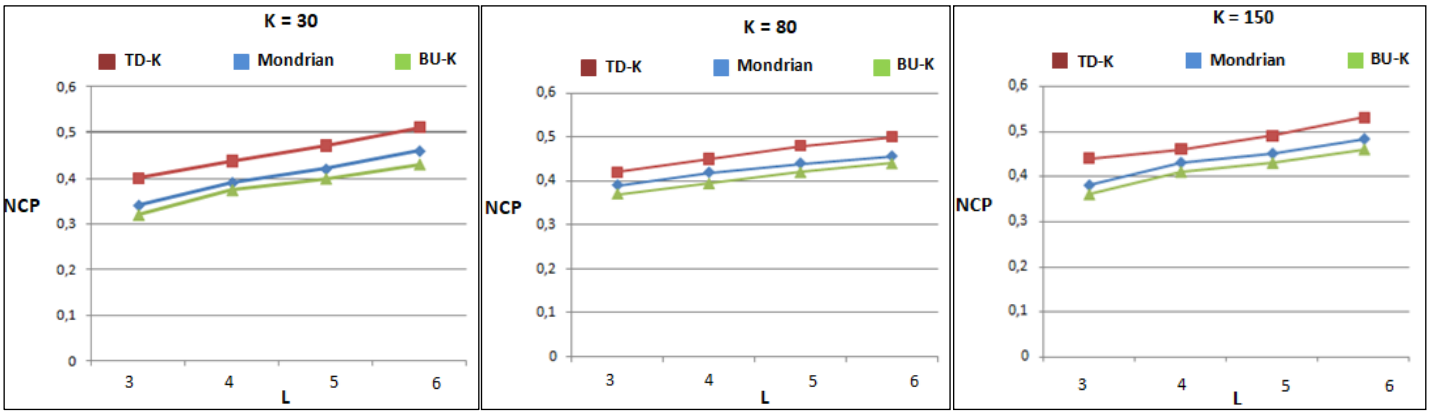

Figure 3: Information loss for different k.
Source: Authors, (2026).

IV.3 CLASSIFIER PERFORMANCE ASSESSMENT

With the growing demand for robust anonymization techniques, the scientific community faces increasing pressure to accurately evaluate the effectiveness of these methods. Traditional anonymization evaluation techniques often fall short, lacking the precision needed to capture subtle differences in the retention of essential data attributes. To address this gap, supervised learning models have emerged as powerful tools, capable of detecting complex patterns and relationships within datasets, thus offering a more comprehensive and detailed evaluation approach [27]. To validate the effectiveness of our proposed method, as shown in Figure 4, both the original and anonymized datasets were subjected to classification. The resulting outputs from the classifiers were then compared to determine the extent to which the anonymization process impacted the classifiers' evaluation metrics. Accuracy was used as the primary criterion for performance measurement and for comparing the results. The supervised learning models employed in this study include Naïve Bayes, decision tree, and random forest. Performance evaluation metrics for the synthetic Adult datasets were gathered, as depicted in Figure 5. Following this, anonymization was applied to these datasets using both the TD-K and BU-K models, with various k values ranging from 30 to 150, to evaluate the effectiveness of the anonymization techniques.

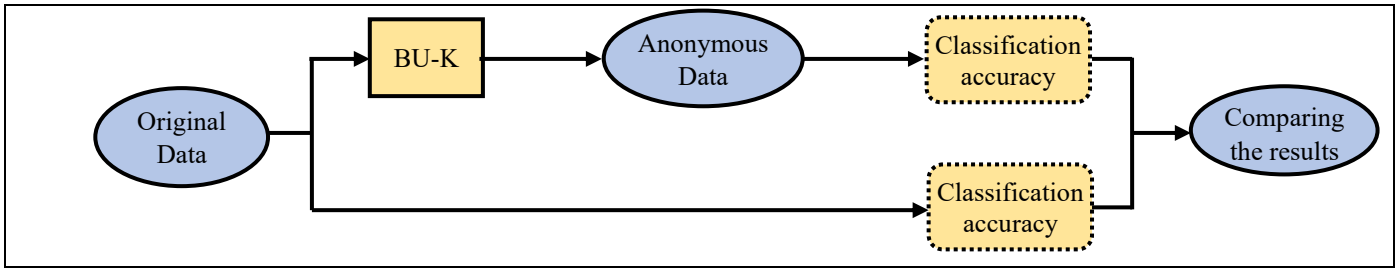

Figure 4: Evaluating classification model outcomes.
Source: Authors, (2026).

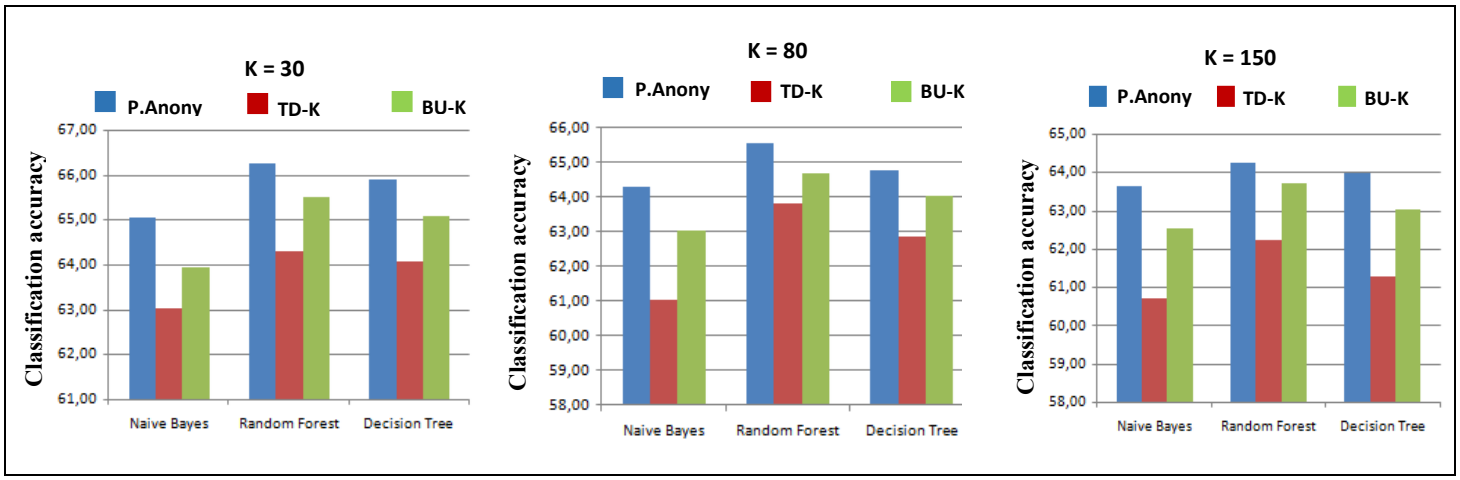

Figure 5: Classification accuracy pre- and post-anonymization for different k.
Source: Authors, (2026).

Given the intrinsic balance between privacy and data utility, the main objective of this study is to improve accuracy while preserving the required level of privacy. As demonstrated in Figure 5, the classifiers achieve their highest accuracy when the k value is set to 30. For the Naïve Bayes classifier, the accuracy rates are 63.03% for the TD-K method and 63.93% for the BU-K method. These

figures reflect decreases of 2.01% and 1.11%, respectively, when compared to the accuracy levels recorded prior to the anonymization process (P.Anony).

For the Random Forest classifier, the accuracy rates are 64.30% for the TD-K method and 65.51% for the BU-K method, representing decreases of 1.95% and 0.74%, respectively. Similarly, for the Decision Tree classifier, accuracy rates of 64.08% for TD-K and 65.07% for BU-K were recorded, with reductions of 1.80% and 0.81%, respectively, compared to pre-anonymization accuracy levels (P.Anony). These results clearly indicate that the accuracy reduction in our proposed approach (BU-K) is significantly smaller compared to the TD-K method. The Bottom-Up k-anonymization (BU-K) method outperforms the Top-Down k-anonymization (TD-K) approach in classifier performance due to its granular and targeted generalization strategy, which minimizes information loss and preserves critical data attributes. This leads to improved accuracy for classifiers trained on BU-K anonymized data. In contrast, TD-K's broader generalizations result in greater information loss and reduced classifier effectiveness. Overall, BU-K's adaptability to heterogeneous datasets and its ability to maintain data diversity contribute to its superiority in machine learning applications.

IV.4 RUNNING TIME

Apache Spark is a high-performance ecosystem specifically designed for processing large datasets, with one of its key advantages being its capability to store intermediate data in memory. This reduces the frequency of read-write operations and allows applications to execute up to 100 times faster than those using in-memory MapReduce and up to 10 times faster than those running on disk. However, it is crucial to recognize that comparing the execution time of MapReduce-Mondrian with our approach may not be methodologically appropriate, even if the tests are conducted under identical configurations. This is because the execution models and optimization techniques inherent to each system are fundamentally different, which can significantly affect performance outcomes [21]. Figure 6 presents a comparison of execution times between the proposed Bottom-up k-anonymization (BU-K) and Top-down k-anonymization (TD-K) methods, using a synthetic adult dataset. Running times, measured in seconds, are evaluated across different anonymization configurations with varying k values (30, 50, 80, and 100). To assess scalability, both methods are tested on datasets of varying sizes: Adult 500K, Adult 1M, and Adult 10M.

In the implementation phase of our study, we thoroughly evaluate the running time efficiency of the Bottom-up k-anonymization approach compared to the Top-down k-anonymization method, both within the Apache Spark environment. The findings reveal that the Bottom-up approach consistently exhibits superior performance in terms of execution time. This efficiency stems from its incremental and targeted generalization strategy, which processes data in a more refined manner by focusing solely on the specific areas requiring anonymization. Consequently, this reduces the computational overhead associated with unnecessary data operations. In contrast, the Top-down method begins with broad generalizations, which increases the volume of data that must be processed and thus leads to higher execution times. Our experiments, conducted across datasets of varying sizes demonstrate that the BU-K approach scales more effectively, showing reduced execution times even as dataset sizes grow. This enhanced performance underscores the Bottom-up method's advantage in efficiently managing large-scale data anonymization tasks, making it a more time-efficient alternative to the TD-K approach.

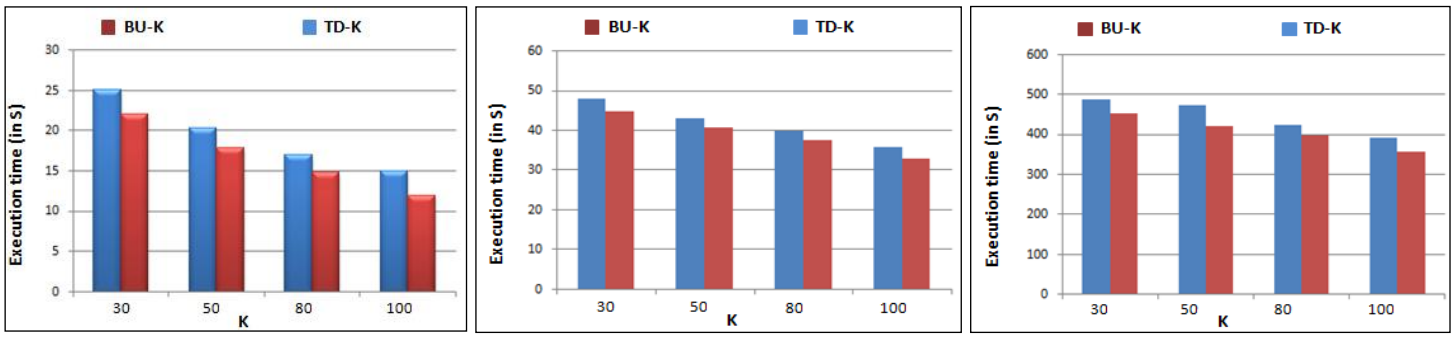

Figure 6: Running time for different Adult size.
Source: Authors, (2026).

V. CONCLUSIONS

This paper presented a Bottom-Up k-anonymization (BU-K) method implemented with Apache Spark to enhance the scalability and efficiency of data anonymization for large datasets. By leveraging Spark's distributed processing and using the Bottom-Up Generalization (BUG) approach, we achieved faster processing and better data utility preservation compared to traditional methods. Our results confirm the BU-K framework's effectiveness in balancing privacy and performance, making it ideal for large-scale data sharing. Future work could explore additional privacy models. Additionally, optimizing the framework for further reductions in computational overhead and expanding its application to other distributed platforms could pave the way for broader adoption in real-world data anonymization tasks.

VI. AUTHOR'S CONTRIBUTION

Conceptualization: Abderrahmane Saidi, Salheddine Kabou.
Methodology: Abderrahmane Saidi, Salheddine Kabou, Imad eddine Kimi.
Investigation: Salheddine Kabou, Abderrahmane Saidi.
Discussion of results: Salheddine Kabou, Imad eddine Kimi.
Writing – Original Draft: Salheddine Kabou, Abderrahmane Saidi.
Writing – Review and Editing: Salheddine Kabou, Imad eddine Kimi.
Supervision: Salheddine Kabou, Imad eddine Kimi, Laid Gasmi.
Approval of the final text: Salheddine Kabou, Abderrahmane Saidi.

VII. REFERENCES

[1] S. Kabou and S. M. Benslimane, "A new distributed anonymization protocol with minimal loss of information", Int. J. Organizational and Collective Intelligence, vol 7(1), pp.1–19, 2017.

[2] S. Kabou, S. M. Benslimane, and M. Mosteghanemi,"A survey on privacy preserving dynamic data publishing", In Research Anthology on Privatizing and Securing Data (pp. 1635-1657). IGI Global, 2021.

[3] L. Sweeney, "k-anonymity: a model for protecting privacy", Int. J. Uncertainty, Fuzziness Knowledge-Based Syst. Vol 10, no 05, pp.557–570, 2002.

[4] A. Machanavajjhala, D. Kifer, J. Gehrke, M. Venkitasubramaniam, "l-diversity: Privacy beyond k-anonymity", ACM Trans. Knowl. Discov. from Data, vol 1, no 01,2006

[5] S. Kabou and S. M. Benslimane, "Building virtual anonymized databases for the Cloud", International Conference on Cloud Technologies and Applications (CloudTech), Marrakech, Morocco, 2015.

[6] Fung, B. C., Wang, K., Chen, R., & Yu, P. S, "Privacy preserving data publishing: A survey of recent developments",ACM Computing Surveys (Csur), vol 42, no 04,pp. 1-53, 2010.

[7] Kabou, S., gasmi, L. & Kabou, A, " BuKc: A novel bottom-up approach for enhanced data anonymization in apache spark". Int. J. Inf. Secur. vol 25, no 22 (2026).

[8] Kabou, S., Gasmi, L., Kabou, A., & Benslimane, S. M, "ImDMI: Improved Distributed M-Invariance model to achieve privacy continuous big data publishing using Apache Spark", Big Data Research, vol 40, pp. 100519, 2025.

[9] Fernández-Gómez, A. M., Gutiérrez-Avilés, D., Troncoso, A., & Martínez-Álvarez, F., "A new Apache Spark-based framework for big data streaming forecasting in IoT networks". The Journal of Supercomputing, 1-23, 2023

[10] Wang, K., Yu, P. S., & Chakraborty, S, "Bottom-up generalization: A data mining solution to privacy protection". In Fourth IEEE International Conference on Data Mining (ICDM'04) (pp. 249-256), 2004.

[11] Irudayasamy, A., & Arockiam, L., "Parallel bottom-up generalization approach for data anonymization using mapreduce for security of data in public cloud". Indian journal of science and technology, vol 8, no 22, 2015.

[12] Kabou, S., Gasmi, L., Kabou, A., & Benslimane, S. M, "A new bottom-up l-diversity method for Apache spark", International Journal of Information and Computer Security, vol 26, no 03, pp.272-290, 2025.

[13] LeFevre, K., DeWitt, D. J., & Ramakrishnan, R, "Mondrian multidimensional k-anonymity. In 22nd International conference on data engineering (ICDE'06) (pp. 25-25). April, 2006.

[14] Yaseen, S, et al, "Improved generalization for secure data publishing". IEEE Access, 6, 27156-27165, 2018.

[15] Majeed, A., & Lee, S, "Attribute susceptibility and entropy based data anonymization to improve users community privacy and utility in publishing data". Applied Intelligence, vol 50, no 08, 2555-2574, 2020.

[16] Basapur, S. B., & Shylaja, B. S, "Attribute assailability and sensitive attribute frequency based data generalization algorithm for privacy preservation". In 2021 International Conference on Forensics, Analytics, Big Data, Security (FABS) (Vol. 1, pp. i-xiv), December, 2021.

[17] Torra, V., & Navarro-Arribas, G, "Attribute disclosure risk for k-anonymity: the case of numerical data". International Journal of Information Security, 1-10, 2023.

[18] Su, B., Huang, J., Miao, K., Wang, Z., Zhang, X., & Chen, Y, "K-Anonymity Privacy Protection Algorithm for Multi-Dimensional Data against Skewness and Similarity Attacks". Sensors, vol 23, no 03,pp. 1554, 2023.

[19] Sopaoglu, U., & Abul, O, "A top-down k-anonymization implementation for apache spark". In 2017 IEEE International conference on big data (Big Data) (pp. 4513-4521), December,2017.

[20] Zakerzadeh, H., Aggarwal, C. C., & Barker, K, "Privacy preserving big data publishing". In Proceedings of the 27th international conference on scientific and statistical database management (pp. 1-11), June, 2015

[21] F. Ashkouti, K. Khamforoosh and A. Sheikhahmadi, "DIMondrian-Distributed improved Mondrian for satisfaction of the L-diversity privacy model using Apach Spark",Information Sciences, vol. 546, pp. 1-24, 2021.

[22] Jain, P., Gyanchandani, M., & Khare, N, "Enhanced secured map reduce layer for big data privacy and security", Journal of Big Data, vol 6, no 01, 30, 2019.

[23] Pandilakshmi, K. R., & Banu, G. R, "An advanced bottom up generalization approach for big data on cloud". Int J Comput Algor, 3, 1054-9, 2014.

[24] Raju, N. L., Seetaramanath, M. N., & Rao, P. S, "An enhanced dynamic KC-slice model for privacy preserving data publishing with multiple sensitive attributes by inducing sensitivity". Journal of King Saud University-Computer and Information Sciences, vol 34, no 01, pp. 1394-1406, 2022.

[25] Jain, P., Gyanchandani, M., & Khare, N, "Improved k-anonymize and l-diverse approach for privacy preserving big data publishing using MPSEC dataset". Computing and Informatics, vol 39, no 03, pp. 537-567, 2020.

[26] Mehta, B. B., & Rao, U. P, "Improved l-diversity: scalable anonymization approach for privacy preserving big data publishing", Journal of King Saud University-Computer and Information Sciences, vol 34, no 4, pp 1423-1430, 2022

[27] S.kabou, Z. Rabhi, A.H.Seddik, and R.Masmoudi, " Data anonymization through supervised Machine Learning", Studies in Engineering and Exact Sciences, vol.5, no.3, pp. e12696-e12696, 2024.