Improve data backup strategies with machine learning predictive analytics

Andrii Harasivka [1,*,†], Anatolii Lupenko [1,†], Yuri Palaniza [1,†] and Mykhailo Fryz [1,†]

[1] *Ternopil Ivan Puluj National Technical University, 56, Ruska Street, Ternopil, 46001, Ukraine*

Abstract

This paper investigates the application of machine learning (ML) model to predict data backup needs and optimize backup solutions in any-scale IT environment. By leveraging ML-driven predictive analytics, users and companies can enhance the efficiency and reliability of their data backup processes, increase performance, reduce costs, and minimize data loss in case incident. The paper describes a solution utilizing the RandomForestRegressor model for learning on attributes of existing files in system to predict a priority of processing backup of files and avoid data redundancy via skipping extra files. It will allow to speed up the backup process and reduce the size of the backup. With enough training on metadata of files and filesystem behavior, the solution will help make backup software more resistant to errors, intelligent and dynamic.

Keywords

data backup, machine learning, software development, RandomForestRegressor

1. Introduction

Data is a critical asset for companies in the digital age, who use it as the basis of all operations, decision-making, and strategic planning [1]. Data loss due to hardware failures, cyber-attacks, or human errors can have catastrophic consequences, including financial losses, reputational damage, and operational disruptions. Traditional data backup strategies often rely on fixed schedules and heuristic rules, which can be inefficient and insufficiently responsive to the dynamic nature of modern IT environments.

The primary problem is the lack of adaptive, intelligent systems that can predict and optimize data backup scope needs in real-time. Fixed backup schedules often lead to excessive resource consumption by frequently backing up low-priority data and neglecting the needed backup of critical, high-priority data [2]. Additionally, these traditional methods do not take into account evolving data usage patterns, system loads, and emerging threats, resulting in insufficient backup performance and potential data loss [3].

Therefore, there is a need for a new approach that leverages new technics to perform predictive analytics for data backup strategies. For such needs perfectly fit one of machine

*ITTAP'2024: 4th International Workshop on Information Technologies: Theoretical and Applied Problems, November 20–22, 2024, Ternopil, Ukraine, Opole, Poland

[1] andriygarasivka@gmail.com (A. Harasivka), lupenkoan@gmail.com (A. Lupenko), palanizayb@tntu.edu.ua (Y. Palaniza), mykh.fryz@gmail.com (M. Fryz)

0009-0003-9271-1183 (A. Harasivka), 0000-0001-5127-4739 (A. Lupenko), 0000-0002-8710-953X (Y. Palaniza), 0000-0002-8720-6479 (M. Fryz)

learning models [4], for example, random forest regressor. By predicting possible scenarios of data modifications, and identifying critical entities of data, ML-driven solutions can improve the efficiency, reliability, and cost-effectiveness of data backup strategy: processes, environments, and schedule. This research aims to contribute to the field of developing data backup systems by providing a novel approach for analyzing data with the use of ML.

2. Literature review

Every company that processes any data should take responsibility not only for its secure and reliable storage but also for corresponding data backup solution. Currently, existing backup software solutions could perform backups of separate files, folders, or volumes of data via a straight and narrow approach by performing one or few backup schemes or its combinations (full, incremental, differential). Full data backup means the solution will make a full copy of source data, it offers the simplest recovery but consumes the most time and storage space. Incremental backup - only backs up data that has changed since the last backup, saving time and storage but requiring multiple backups for a full restore. Differential backup saves all changes made since the last full backup, balancing time and storage between full and incremental backups [5].

To avoid weak parts of existing backup solutions and make the system more dynamic, a new machine-learning-driven solution should help. ML algorithms can significantly improve data backup strategies through various approaches, optimizing both the efficiency and reliability of the backup strategy. This allows to create more intelligent schedule of backups, ensuring that critical data is backed up more frequently while less critical data is backed up less often. This reduces the amount of backups, storage usage and maintains efficient storage utilization.

Another area to optimize – make adaptive backup policies based on real-time monitoring of system resources. For instance, if the ML algorithm detects increased activity in certain datasets, it can trigger more frequent backups for those datasets temporarily. By monitoring the usage of machine resources and their trends (e.g., bandwidth, storage, memory, central processing unit), the ML algorithm can predict the best times for backups to ensure minimal disruption to regular operations and efficient use of resources.

Due to the increasing amount of data [6], every software solution should consider applying single or multiple options of data deduplication and compression. ML algorithm can analyze data type or attributes and classify data, to enhance deduplication and compression techniques, choose the best archiving mechanism, and reduce the amount of storage required. Critical data might receive more frequent and robust backup measures, while less important data is backed up less frequently or with fewer resources [7].

Additionally, ML algorithms can predict potential backup failures by analyzing system events, logs, and metrics. This allows the backup system to take proactive measures, such as rerunning a backup process or fixing underlying issues before a critical backup operation fails. If a system event happens during backup, the ML algorithm can initiate a spare process of verifying backups and checking their integrity. This ensures that backups are complete, uncorrupted, and can be restored successfully when needed.

Every company has its own policies for IT environment, including data storage, its backups and retention periods, security of systems; so the ML algorithm can ensure that data backup

strategies comply with organizational policies and regulatory requirements by continuously monitoring backup processes and systems to meet compliance standards.

To overcome the problems of existing backup software, a new machine-learning backup solution is proposed to solve fixed schedule and redundant data problems. For this purpose, RandomForestRegressor model was selected. This model has the resilience to noisy data, no need for extensive data preprocessing, ability to handle non-linear relationships which make it perfect for performing analytics among file system entries and user behavior.

3. Proposed solution

The proposed software solution should solve the problems mentioned above - avoid redundant files, and make the backup schedule by assigning priority for tasks. The main advantage of the proposed solution will be a feature of predicting priority to make backup of important files faster and speed up the backup process.

For that purpose will be enough of a simple console application. It should consist of a few modules: main – which is the entry point of a system; BackupScheduler – a service to fire backup activities (directly or via timer), BackupExecutor – a main component that performs backup activity, FileChangeStatisticsHandler – a module for tracking of file system changes. The system structure diagram is shown in figure 1.

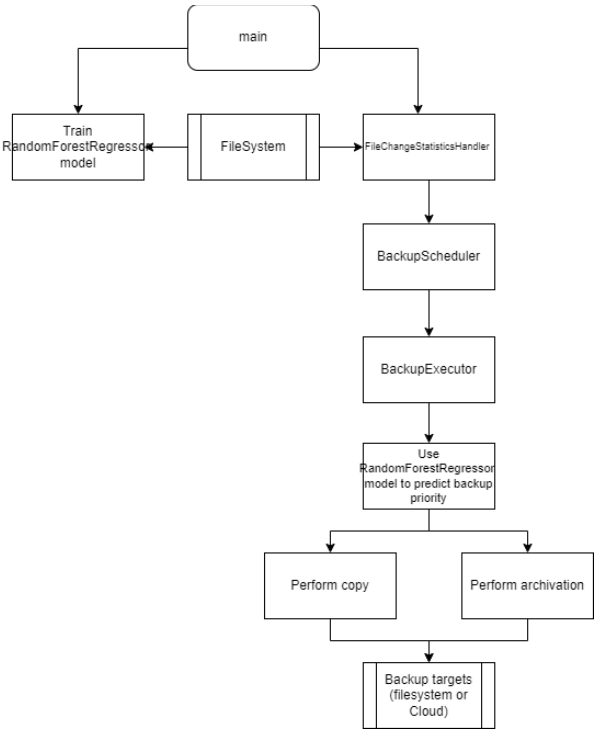

Figure 1: Structure diagram of a proposed solution

Development of such a solution will require a Python interpreter installed, Pip configured, Pandas and Scikit-learn, NumPy modules imported, text editor on Windows/Linux/MacOS machine. Additional features, like filesystem access were imported from os, shutil, and glob.

To be able to train a RandomForestRegressor model there will be a need for a large set of data, which could be any directory with different types of files, for the proposed solution it will be %userprofile%\Documents.

In the first step, need to define the criteria of files and how they should be preprocessed - the model should build relations to predict “backup_priority” and “backup_type” of file. Preprocessing allows data to make it suitable for training machine learning models. Model training will be performed by such attributes as: “size”, “last_modified”, “created“, “is_system”, “is_hidden”, “is_readonly”. Target attributes that should be predicted are: “backup_priority” and “backup_type”. Both groups of attributes should be defined, so a model could split data into logical relations between source and result and make sufficient decisions in the future.

The process of model training is iterative: which means the developer should pass incoming variables and review model prediction results in every iteration – if the prediction value does not fall into the expected range – adjust incoming values and repeat. All coefficients should be adjusted due to the exact filesystem and case. It could be implemented by simple conditions – for example, if the file was created less than a minute ago – increment priority by 10, or if file size is less than file system block size (for Microsoft Windows its 4kb) – then it could be more efficient to copy it rather then archive. The algorithm of estimating “backup_priority” is displayed in figure 2 (as there are many criteria to estimate – most of them hidden in function).

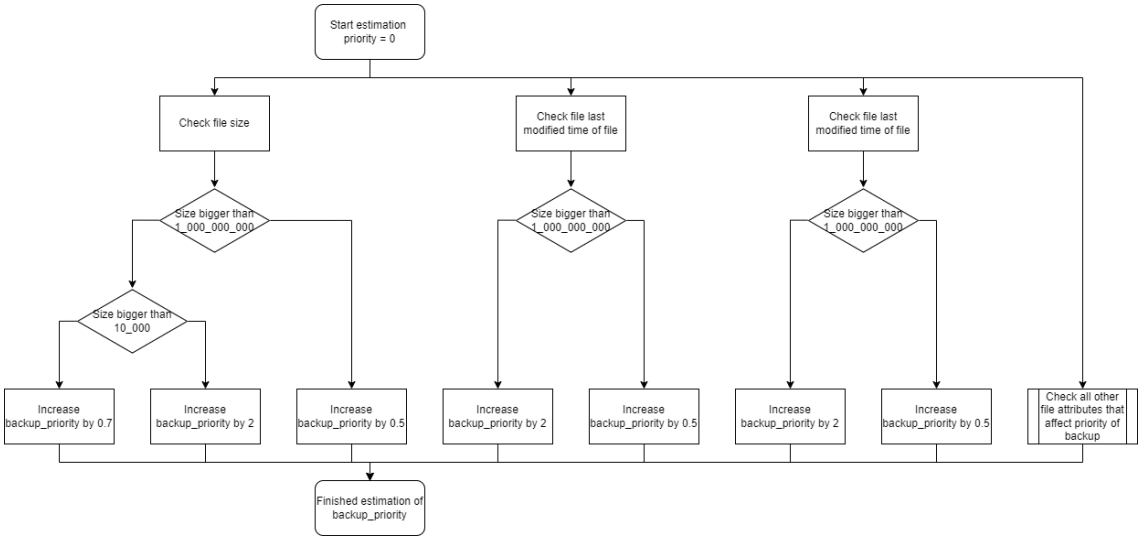

Figure 2: Algorithm of estimating backup priority of sample file

Start of the software will be from the Main module. It's an entry point of a console program, which takes arguments: training data path, backup source path, backup target path. Main initiate analyzing of scanning files in the training data path and start RandomForestRegressor model training. It includes the following steps:

1. splitting data into training and actual (made by function “train_test_split” from “sklearn.model_selection”)
2. creating multi-target regressor – to be able to predict multiple variables with the same model (class “MultiOutputRegressor”)

3. fit model to train data (function “fit”).

BackupScheduler has no logic except repeated timer and queue which allows to schedule, postpone, and send backup tasks to BackupExecutor. Additionally, it utilizes CPU and RAM metrics of OS to avoid making backup task when OS under load.

FileChangeStatisticsHandler is a handler derived from “FileSystemEventHandler” to be able to track file system modifications. It implements “on_modified” method which fired on any modification of a file. Unfortunately, it fired even if file metadata changed (in Microsoft Windows 10 environment). So, to avoid extra changes of “last_access” attribute of a file, handler should also calculate, save, and compare MD5 hash of file, it can be made by “md5” function of “hashlib”. Additionally, to have full history of file changes its save dictionary of dates by file into a temporary file.

BackupExecutor uses trained model to predict the backup priority of file(s) by its metadata. “Predict” function return array of results, it our case it is 2 target values due to 2 target attributes ”backup_priority” and “backup_type”. Based on the prediction - a corresponding backup action will be made: either ignore the file (e.g. ”backup_priority” have 0 importance for backup), copy the file (e.g. it’s a small file), or zip file (it's a large file).

4. Evaluation of results

Evaluation of data backup solutions involves comparing different factors to determine which option best meets specific needs. Data backup solutions should consider the type and volume of data, the speed and frequency of backups, redundancy, storage use and reliability. Need to have a set of metrics to evaluate and compare the performance and redundancy of existing and new backup solutions:

- Compressed size: size of compressed backup files [8], measured in bytes.
- Space saving: the reduction in the size of original data relative to the uncompressed data size, calculated by the formula (1).
- Time spent: The duration of the backup process, measured in s.
- Redundancy: The size of extra files that are included in the backup (temporary files made by other software, operating system files, non-accessible files) - redundant and irrelevant files, calculated by formula (2).

$$Space\ saving, \% = 1 - Compressed ¿¿ Uncompressed ¿¿ 100, ¿¿ \quad (1)$$

$$Redundancy, \% = 1 - Redundant\ files ¿¿ Compressed ¿¿ 100, ¿¿ \quad (2)$$

Attributes "Space saving" and "Redundancy" are the key indicators of the quality of a backup solution. With ideal software, the "Space saving" should be positive, and as large as possible, and the "redundancy value" should lead to 0.

Software solutions perform backup to the same disk as source data to keep hardware input/output delay as small as possible. The technical specification of the test machine is shown in table 1.

Table 1
Test machine configuration.

Category	Name	Specifications
Hardware	Main board	MSI X470 Gaming Pro
	CPU	AMD Ryzen 7 2700X 3.7 GHz
	Hard disk	Samsung SSD 970 EVO 1TB
	Network card	Intel (R) I211 Gigabit Network
	Memory	32 GB DDR4 3200 MHz
	Graphics	NVIDIA GeForce GTX 1060 6GB
Software	Operating system	Windows 10 Pro 22H2

Sample data backup software will include Acronis True Image 29, Paragon Backup & Recovery 17 CE, Duplicati 2.0.8.1. Also to check the performance of compression efficiency additional tests with archivers were performed: WinZip, PeaZip.

Every software provides timestamps or log entries that show the elapsed time of the backup process, so it will be used in “Time spent”. “Uncompressed size”, “Compressed size” values taken from “Properties” window of “Windows Explorer”, its needed to calculate “Space saving” column [9]. “Redundant files size” also taken from “Properties”, it’s a sum of redundant and irrelevant files for backup.

To get metrics of backup solutions need to perform tests - perform backup action of sample data in every software, test data contains 2 sets.

Data set 1 includes example “Documents” directory, enlisting 169 documents, 21 folders (10.8 MB of pictures, 71.3 MB of videos, 155 MB of audio, 491.1 KB of documents) – 250 774 175 bytes. Redundant files include hidden system file “desktop.ini“ and hidden folder “.tmp.driveupload” with read-only files (their sum is 11 702 bytes). Test results of backup test set 1 are provided in table 1.

Data set 2 include example “Downloads” directory, enlisting 460 documents, 21 folders (70,3 MB of pictures, 907.8 MB of videos, 17.5 KB of documents, 3.7 of binary files), total - 4 995 098 485 bytes. Redundant files include file hidden system files “desktop.ini“ and hidden folder “.tmp.driveupload” with read-only files (9 285 404 bytes). Test results of backup test set 1 are provided in table 2.

Table 2
Metrics of performance of making a full backup via software for sample data set 1.

Software	Uncompressed size, bytes	Time spent, s	Compressed size, bytes	Space saving, %	Redundant files size, bytes	Redundancy, %
Acronis	250 774 175	8	245 683 200	2.03	11 702	0
Paragon	250 774 175	90	251 187 200	-0.16	11 702	0
Duplicati	250 774 175	8	179 387 221	28.47	11 702	0.01
PeaZip	250 774 175	37,55	240 815 723	3.97	11 702	0
WinZip	250 774 175	2,60	241 507 281	3.7	11 702	0
Proposed solution	250 774 175	5,33	241 486 408	3.7	0	0

Table 3
Metrics of performance of making a full backup via software for sample data set 2.

Software	Uncompressed size, bytes	Time spent, s	Compressed size, bytes	Space saving, %	Redundant files size, bytes	Redundancy, %
Acronis	4 995 098 485	19	4 903 413 248	1.84	9 285 404	0.19
Paragon	4 995 098 485	106	4 946 669 568	0.97	9 285 404	0.19
Duplicati	4 995 098 485	47	4 918 623 708	1.53	9 285 404	0.19
PeaZip	4 995 098 485	664,44	4 887 399 494	2.16	9 285 404	0.19
WinZip	4 995 098 485	24,12	4 890 370 500	2.1	9 285 404	0.19
Proposed solution	4 995 098 485	91,483	4 880 723 436	2.29	2 429 307	0.05

As we can see from the results in table 2 most software performs a backup of redundant files (column “Redundancy files size” - 11 702 bytes), while the proposed solution has 0% redundancy, which is the perfect case among backup software. Also, proposed solution has a high level of “space saving”: 3,7%, while the biggest level is achieved by Duplicati: 28,47%. Additionally, the test for the proposed solution took only 5,33 seconds which is second place after the best - 2,60 seconds by WinZip.

The results of the second test displayed in table 3, confirms that the proposed solution has the smallest value of “redundant files size”: 2 429 307bytes (0,04% redundancy), which is ~4 times less than competitors. The value of “space saving” of the proposed solution is the biggest among competitive software - 2,29%, but it took 91,483 seconds.

So, the evaluation of the result of testing the proposed solution shows good conclusions: the smallest size of redundant files, a high value of “Space saving” and a low amount of “Time spent” means that the RandomForestRegressor model successfully fit for our needs. Applying this machine learning model will significantly improve the performance of backup software – so end-customers will see that backup software will check their latest files first and reduce time spent on backup.

5. Conclusion and future work

The paper provides insights into the potential of ML-driven predictive analytics solutions to improve data backup strategies, ultimately contributing to more robust, efficient, and adaptive systems. By integrating these ML approaches, software companies can develop more sophisticated and responsive data backup solutions to improve existing IT environments.

One of the key areas for improvement could be the use of more advanced multi-target regression models. Specifically, we plan to employ a more complex multi-target XGBoost regressor (in One-Model-Per-Target or Vector Leaf mode) for predicting the target attributes "backup_priority" and "backup_type". One of its primary benefits is its efficiency and scalability, making it suitable for both small and large-scale datasets.

Future steps for the research could be: using data sets with a bigger amount of different files to teach and train a model, including immunosensors [10-13], cyber-physical [14-17] and cardio diagnostic [18-19] systems, fitting and adjusting the model due to a higher amount of attributes: filesystem permissions, the file owner and monitoring system activity during processing

backup; implementing a multithreading execution of backup; deploying the application as a separate application for target operating systems.

References

[1] Mohamed Djerdjouri, Data and Business Intelligence Systems for Competitive Advantage: prospects, challenges, and real-world applications, No. 41: Mercados y Negocios: enero-junio, (2020), 10.32870/myn.v0i41.7537.

[2] Junhong Duan, Bo Yang, Distributed Method for the Backup of Massive Unstructured Data, Journal of Physics Conference Series (2021), 10.1088/1742-6596/1802/3/032106.

[3] Barry Elad, 50+ Backup Statistics 2022- Backup vs. Recovery, Disaster Recovery and Trends, 2022. URL: https://www.enterpriseappstoday.com/stats/backup-statistics.html .

[4] Nikhil Ghadge, MACHINE LEARNING: ENHANCING INTELLIGENT SEARCH AND INFORMATION DISCOVERY, Computer Science & Information Technology (CS & IT) (2024), 10.5121/csit.2024.141021.

[5] Mehul Pawar, Anuja S. Phapale, Enhancing Data Backup and Recovery in Cloud Computing with Secure Database Monitoring, International Journal of Applied and Advanced Multidisciplinary Research (IJAAMR)Vol. 1, No. 4, (2023), 10.59890/ijaamr.v1i4.594.

[6] Nadeem U. Shahid, Nasir J. Sheikh, Impact of Big Data on Innovation, Competitive Advantage, Productivity, and Decision Making: Literature Review, Open Journal of Business and Management (2021), 10.4236/ojbm.2021.92032.

[7] Dr. Ashwini Brahme, Satish Batrel, A Study Of Traditional And Recent Data Backup Techniques And Security Risks, International Journal in Multidisciplinary and AcademicResearch (SSIJMAR)Vol. 11, No. 3, June (2023), SSN 2278 – 5973.

[8] Neha Sharma, Usha Batra, EVALUATION OF LOSSLESS ALGORITHMS FOR DATA COMPRESSION, International Conference on Contemporary Issues in Computing (ICCIC-2020) – Virtual (2020), 10.26480/etit.02.2020.40.44.

[9] S.R. Kodituwakku, U. S.Amarasinghe, COMPARISON OF LOSSLESS DATA COMPRESSION ALGORITHMS FOR TEXT DATA, Indian Journal of Computer Science and Engineering, 1(4) (2010) 416-425.

[10] Martsenyuk V.P., Sverstiuk A.S., Andrushchak I.Ye. Approach to the study of global asymptotic stability of lattice differential equations with delay for modeling of immunosensors (2019) Journal of Automation and Information Sciences, 51 (2), pp. 58 – 71. DOI: 10.1615/jautomatinfscien.v51.i2.70.

[11] Sverstiuk A.S. Research of global attractability of solutions and stability of the immunosensor model using difference equations on the hexagonal lattice (2019) Innovative Biosystems and Bioengineering, 3 (1), pp. 17 – 26. DOI: 10.20535/ibb.2019.3.1.157644.

[12] Martsenyuk, V. P., Andrushchak, I. Ye., Zinko, P. N., & Sverstiuk, A. S. (2018). On Application of Latticed Differential Equations with a Delay for Immunosensor Modeling. In Journal of Automation and Information Sciences (Vol. 50, Issue 6, pp. 55–65). Begell House. https://doi.org/10.1615/jautomatinfscien.v50.i6.50.

[13] Martsenyuk, V., Sverstiuk, A., & Gvozdetska, I. S. (2019). Using Differential Equations with Time Delay on a Hexagonal Lattice for Modeling Immunosensors. In Cybernetics and

Systems Analysis (Vol. 55, Issue 4, pp. 625–637). Springer Science and Business Media LLC. https://doi.org/10.1007/s10559-019-00171-2

[14] Martsenyuk, V., Klos-Witkowska, A., Sverstiuk, A., Bahrii-Zaiats O., Bernas, M., Witos, K. Intelligent big data system based on scientific machine learning of cyber-physical systems of medical and biological processes. CEUR Workshop Proceedings, 2021, 2864, pp. 34–48.

[15] Martsenyuk, V., Sverstiuk, A., Bahrii-Zaiats, O., Kłos-Witkowska, A. Qualitative and Quantitative Comparative Analysis of Results of Numerical Simulation of Cyber-Physical Biosensor Systems. CEUR Workshop Proceedings, 2022, 3309, pp. 134–149.

[16] Martsenyuk V., Sverstiuk A., Klos-Witkowska L., Nataliia K., Bagriy-Zayats O., Zubenko I. Numerical analysis of results simulation of cyber-physical biosensor systems (2019) CEUR Workshop Proceedings, 2516, pp. 149 – 164.

[17] Martsenyuk V., Sverstiuk A., Bahrii-Zaiats O., Kłos-Witkowska A. Qualitative and Quantitative Comparative Analysis of Results of Numerical Simulation of Cyber-Physical Biosensor Systems (2022) CEUR Workshop Proceedings, 3309, pp. 134 – 149.

[18] Trysnyuk V., Zozulia A., Lupenko S., Lytvynenko I., Sverstiuk A. Methods of rhythm-cardio signals processing based on a mathematical model in the form of a vector of stationary and stationary connected random sequences (2021) CEUR Workshop Proceedings, 3021, pp. 197 – 205.

[19] V. MARTSENYUK; I. ANDRUSHCHAK; N. KOZODII; Y. KRAVCHYK; A. SVERSTIUK; Y. PALANIZA. COMPARISON OF RESULTS OF NUMERICAL ANALYSIS OF SIMULATION OF CYBERPHYSICAL BIOSENSOR SYSTEMS (2023) Herald of Khmelnytskyi National University. Technical sciences, pp. 202 – 212. https://doi.org/10.31891/2307-5732-2023-319-1-202-212