COMPUTER SCIENCE AND INTERDISCIPLINARY RESEARCH JOURNAL

Volume 2, Issue 1, 2025
ISSN: 3033-1218

https://csirjournal.com

Computer Science and Interdisciplinary Research Journal [CSIR Journal]

ISSN: 3033-1218
Volume 2, Issue 1, 2025.

Published by Science, Education, Technologies Association
Plovdiv, Bulgaria

Licensed under CC BY-NC-ND 4.0.

The views expressed in the articles are those of the authors and do not necessarily reflect the views of the publisher or editorial team.

Editorial Team

Editors-in-Chief

Section COMPUTER SCIENCE

Prof. Asya Stoyanova-Doycheva, Ph.D.

University of Plovdiv Paisii Hilendarski, Plovdiv, Bulgaria

Section MATHEMATICS

Assoc. Prof. Mariyan Milev, Ph.D.

Sofia University St. Kliment Ohridski, Sofia, Bulgaria

Section EDUCATION

Prof. Todorka Glushkova, Ph.D.

University of Plovdiv Paisii Hilendarski, Plovdiv, Bulgaria

Section INTERDISCIPLINARY RESEARCH

Prof. Stanka Hadzhikoleva, Ph.D.

University of Plovdiv Paisii Hilendarski, Plovdiv, Bulgaria

Editorial Board

Prof. Aldo Tagliani, Ph.D.

Università di Trento, Trento, Italy

Prof. Igor Britchenko, Ph.D.

University of the National Education Commission, Krakow, Poland

Prof. Emil Hadzhikolev, Ph.D.

University of Plovdiv Paisii Hilendarski, Plovdiv, Bulgaria

Prof. Desislava Paneva-Marinova, Ph.D.

Bulgarian Academy of Sciences, Sofia, Bulgaria

Prof. Miglena Koleva, Ph.D.

University of Rousse, Ruse, Bulgaria

Assoc. Prof. Senthil Kumar, Ph.D.

Mepco Schlenk Engineering College, Sivakasi, India

Assoc. Prof. Margarita Terziyska, Ph.D.

University of Food Technology, Plovdiv, Bulgaria

Assoc. Prof. Stoyan Cheresharov, Ph.D.
University of Plovdiv Paisii Hilendarski, Plovdiv, Bulgaria

Assoc. Prof. Nikola Shakev, Ph.D.
Technical University of Sofia, Plovdiv Branch, Plovdiv, Bulgaria

Assoc. Prof. Velika Kuneva, Ph.D.
Agricultural University, Plovdiv, Bulgaria

Assoc. Prof. Silvia Gaftandzhieva, Ph.D.
University of Plovdiv Paisii Hilendarski, Plovdiv, Bulgaria

Assoc. Prof. Hristina Kulina, Ph.D.
University of Plovdiv Paisii Hilendarski, Plovdiv, Bulgaria

Assoc. Prof. Mikhail Kolev, Ph.D.
University of Architecture, Civil Engineering and Geodesy, Sofia, Bulgaria

Assoc. Prof. Hristo Hristov, Ph.D.
University of Plovdiv Paisii Hilendarski, Plovdiv, Bulgaria

Assist. Prof. Shih-Hao Chang, Ph.D.
National Taipei University of Technology, Taiwan

Assist. Prof. Gordana Rakić, Ph.D.
University of Novi Sad, Serbia

Assist. Prof. Kostadin Yotov, Ph.D.
University of Plovdiv Paisii Hilendarski, Plovdiv, Bulgaria

English Proofreader

Assoc. Prof. Vanya Ivanova, Ph.D.
University of Plovdiv Paisii Hilendarski, Plovdiv, Bulgaria

Design Assistance

Assist. Prof. Maya Stoeva, Ph.D.
University of Plovdiv Paisii Hilendarski, Plovdiv, Bulgaria

Technical Assistance

Ph.D. student Maria Gorgorova
University of Plovdiv Paisii Hilendarski, Plovdiv, Bulgaria

Preface

Dear Readers,

We are delighted to present the second volume of the *Computer Science and Interdisciplinary Research Journal* (CSIR Journal). Following the warm reception of our inaugural issue, we are encouraged by the growing interest and engagement from researchers, practitioners, and readers who share our passion for the advancement of science and technology in an interdisciplinary context.

The second issue continues our mission to showcase high-quality scientific research and to promote dialogue between theory and practice in the field of computer science and its applications across various domains. The articles included in this issue reflect the diversity and relevance of current scientific exploration - from machine learning and data security to innovations in online services and financial cycle forecasting.

This issue features four articles that present innovative solutions and modern approaches in various areas of computer science. The first article introduces *flexmysqlbackup* – a flexible and efficient solution for creating backups of MySQL databases that effectively addresses real-world challenges in data management and recovery. The second contribution presents a hybrid model combining LSTM neural networks and Discrete Fourier Transform, aimed at improving the accuracy of forecasting cyclical movements in financial markets and supporting investment decisions. The third article offers a comparative analysis of key machine learning algorithms applied to classification tasks, evaluating their strengths and limitations across different data types. The final publication presents a classification model for innovations in online sports betting – a systematic approach to a rapidly growing and technology-driven industry.

We extend our sincere thanks to all the authors for their valuable contributions, as well as to the reviewers and the editorial team for their efforts in maintaining the high quality and scientific integrity of the publications. Your collaboration remains a key factor in the journal's success.

We remain committed to providing a platform for meaningful scientific exchange and an interdisciplinary perspective. We hope that this issue of *CSIR Journal* will serve as a source of knowledge, inspiration, and support in your academic and professional endeavors.

Enjoy reading!

EDITORS-IN-CHIEF:

Prof. Asya Stoyanova-Doycheva, Ph.D.
Section COMPUTER SCIENCE

Assoc. Prof. Mariyan Milev, Ph.D.
Section MATHEMATICS

Prof. Todorka Glushkova, Ph.D.
Section EDUCATION

Prof. Stanka Hadzhikoleva, Ph.D.
Section INTERDISCIPLINARY RESEARCH

Table of contents

Preface
Asya Stoyanova-Doycheva, Mariyan Milev, Todorka Glushkova, Stanka Hadzhikoleva 1

A Flexible Approach to MySQL Backup Creation – flexmysqlbackup
Dimitar Todorov 3

Cycle Forecasting in Financial Markets with LSTM and Discrete Fourier Transform
Kostadin Yotov 15

Analyzing the Efficacy of Base Algorithms: Application of Machine Learning Models in Classification Tasks
Yass Khudheir Salal 28

Innovations Classification Model in Online Sportsbook Betting
Petko Iliev 34

Computer Science and Interdisciplinary Research Journal
Volume 2, Issue 1, ISSN: 3033-1218, DOI: 10.70862/CSIR.2025.0201-10, 2025

A Flexible Approach to MySQL Backup Creation – flexmysqlbackup

Dimitar Todorov

University of Plovdiv Paisii Hilendarski, 236, Bulgaria Blvd, 4027 Plovdiv, Bulgaria

dimitar.todorov@uni-plovdiv.bg

Abstract – Creating backups in MySQL is an integral part of database management, particularly crucial in environments with large data volumes and critical information. Data recovery in the event of a system crash, data loss, or migration requirements is made possible by timely and accurate backups. MySQL is one of the most popular databases, utilized by numerous companies and projects. Mysqldump is a standard tool for backup and recovery. While effective for standard backups, mysqldump encounters several challenges in more complex scenarios, particularly when compatibility across different MySQL versions is required. This paper proposes a new approach, flexmysqlbackup, which provides more flexible backup options and rapid recovery without the limitations imposed by MySQL versions. Implementation scripts are presented, along with experiments demonstrating and proving its effectiveness.

Keywords – database backup, database recovery, storage system, MySQL, flexmysqlbackup

1. Introduction

Database backup and restore are essential aspects of database management. It is crucial for database professionals to possess the necessary knowledge and practical experience in data backup and recovery procedures.

DOI: https://doi.org/10.70862/CSIR.2025.0201-10

Corresponding author: Dimitar Todorov,
University of Plovdiv Paisii Hilendarski
Email: dimitar.todorov@uni-plovdiv.bg

Received: 26 April 2025.
Revised: 29 May 2025.
Accepted: 29 May 2025.
Published: 2 June 2025.

© 2024 Dimitar Todorov. This work is licensed under the Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License.

The article is published with Open Access at http://csirjournal.com/

System failures can result from various circumstances, including hardware malfunctions, malware, data corruption, or physical server inaccessibility due to extreme events such as fires or natural disasters. Such situations can range from accidental errors, like connecting to the wrong server or executing a command such as ALTER TABLE [1], to more severe incidents, including malicious attacks or software bugs [2]. Although the probability of any single disaster occurring is relatively low, the accumulation of various risks significantly increases the possibility of data loss or service interruption. It is also surprisingly common to encounter situations where users delete data and subsequently request its restoration.

In cases of application errors leading to incorrect data entries, or during legal proceedings, it is crucial to be able to determine the state of the data or schema at a specific point in the past. Many hosting service providers do not back up MySQL servers at all, while others simply copy the files while the server is running. This approach can potentially create a corrupted backup, rendering it useless [3]. The mysqldump tool provides capabilities for backing up MySQL databases. Despite its popularity and usefulness, mysqldump has limitations that hinder its use in more complex backup and migration scenarios [4,5].

This paper proposes an alternative method, flexmysqlbackup, which offers more flexible and compatible solutions for database backup and recovery across different MySQL versions. Implementation scripts for flexmysqlbackup have been developed and are available at https://github.com/tuthon/flexmysqlbackup.

2. Limitations of Mysqldump

Key limitations of mysqldump include:

- Lack of selectivity in backing up objects;
- Limited compatibility between different MySQL versions;

- Difficulty in migrating specific objects;
- Client/server communication settings issues.

One of the most significant limitations of mysqldump is its lack of options for selectively backing up specific objects such as triggers, events, procedures, or views. For example, if it is necessary to back up only a specific trigger [6] or view [7] from a particular schema, mysqldump does not offer this capability. The available options force users to back up either all objects of a certain type (like all triggers associated with the dumped tables) or the entire database structure, which generates unnecessarily large backups for larger databases and increasing recovery time.

When creating a backup, mysqldump includes system parameters specific to the current MySQL version in the generated script. This frequently leads to errors, difficulties [8], and the need for manual modifications [9] when restoring the data to a different MySQL version. For example, MySQL 8.0 uses a new default collation, utf8mb4_0900_ai_ci, which does not exist in older versions like MySQL 5.7. Attempting to restore a backup created with mysqldump from version 8.0 onto a server running MySQL 5.7 results in an error that halts the restoration process [10].

```
mysql> CREATE TABLE test (id int NOT NULL)
ENGINE=InnoDB DEFAULT CHARSET=utf8mb4
COLLATE=utf8mb4_0900_ai_ci;
- Unknown collation: 'utf8mb4_0900_ai_ci'
```

When the server and the client exchange data, they might use different character sets [11]. The server performs the necessary conversions (Figure 1).

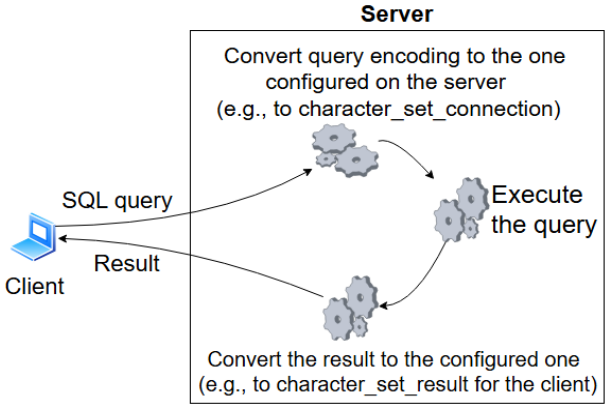


Figure 1. Client and server character sets.

- The server assumes that the client sends queries (or statements) in the character set specified by character_set_client.
- After the server receives a query from the client, it converts it to the character set specified by character_set_connection, which also determines how numbers are converted to strings.
- When the server returns results or error messages to the client, they are converted to the character set specified by character_set_result.

The solution involves using the server commands SET NAMES and SET CHARACTER SET. Client applications and APIs must also be configured with the correct character set, and mysql_real_escape_string() should be used to escape values [12,13].

During database migrations, it is often necessary to back up only specific objects or schemas to minimize errors and optimize the process. However, mysqldump does not offer sufficiently flexible options to perform this selectively. As a result, the database is frequently transferred with a larger volume of data than required, complicating both the restoration process and change management during migration.

A key disadvantage of mysqldump is the automatic inclusion of the mysql system schema when using options such as --all-databases. The mysql schema stores information regarding users, privileges, and other server-specific settings [14,15]. This can cause serious problems when restoring the backup onto a different server, such as:

- Overwriting existing users and privileges, which can lead to loss of access or unauthorized access.
- Incompatibilities if the servers use different MySQL versions or have specific configurations in the mysql schema.

These risks require careful handling and often necessitate manual editing of the backup file to exclude system schemas during migration.

To minimize these risks, it is recommended to:

- Exclude the mysql system schema by using more specific mysqldump options (instead of options like --all-databases).

```
mysqldump -u root -p --databases db1 db2 > backup.sql
```

- Manually edit the backup file to remove unwanted statements related to the mysql schema.
- Create separate backups for system data (like user grants and roles) if they are needed for restoring privileges or roles.

Due to these and other limitations, mysqldump is not always sufficiently flexible or reliable for modern application requirements.

A comparison between the main features of mysqldump and flexmysqlbackup is presented in Table 1.

Table 1. Comparison of main features of mysqldump and flexmysqlbackup.

Function	mysqldump	flexmysqlbackup
Version support	Good for the current version; some incompatibility between versions, manual intervention often required.	Better compatibility with versions 5.6 to 8.0
Archiving by schema	Partial (inconvenient)	Yes, through filters by schema and objects
Archiving of procedures, functions, triggers	Limited	Yes
Selective restore	Difficult or unsupported	Supported
Complexity of use	Low	Moderate
Performance	Average	Depends on configuration and version. In most cases, faster.

3. The Flexmysqlbackup Method for Flexible Backup

The proposed flexmysqlbackup method offers a flexible and effective solution for creating MySQL backups, overcoming the limitations of mysqldump. Its core concept relies on customized SQL scripts that extract definitions from the information schema database, the Stream Editor (sed) for text processing, and automation via Bash scripting. This approach provides flexibility, efficiency, and compatibility in backup management.

3.1. Core Elements of Flexmysqlbackup

INFORMATION_SCHEMA is a standard schema, supported by MySQL and other relational database management systems (RDBMS), which provides metadata about database objects such as tables, columns, views, triggers, and procedures. It serves as a read-only "data dictionary" or "catalog" providing access to information about the structure of the databases. MySQL introduced this schema in version 5.0, adhering to the SQL:2003 standard, but also including some MySQL-specific views. INFORMATION_SCHEMA is a key tool for administrators and developers as it allows standardized access to information about database structure. This facilitates database management and query optimization [16]. The most important tables within INFORMATION_SCHEMA are presented in Table 2.

A significant part of the flexmysqlbackup logic is implemented using stored procedures.

Stored procedures allow developers to group SQL statements that perform specific operations, such as transaction processing or data management [17]. They offer advantages like enhanced code reliability, increased security by limiting direct database access, and easier migration of business logic from applications to the database.

Table 2. Key tables in INFORMATION_SCHEMA.

Table	Description
TABLES	Contains information about all tables, such as name, type (BASE TABLE or VIEW), storage engine used (ENGINE), and other metadata
COLUMNS	Provides details about columns in tables, such as column name, data type (DATA_TYPE), character maximum length, and whether NULL values are allowed.
SCHEMATA	Lists available schemas, showing the default character set and collation.
VIEWS	Contains information about views, including their SQL definition and check option (update constraints).
TRIGGERS	Contains data about triggers, such as trigger name, event (INSERT, UPDATE, DELETE), and the table it is associated with.
EVENTS	Maintains information about scheduled events, such as start and end times, and the event type.

Bash is a compatible successor to the Bourne shell, incorporating key features from the C shell (csh) and the Korn shell (ksh), such as directory manipulation, job control, and brace expansion. One of the main reasons for Bash's popularity among database administrators is its compatibility with existing Unix shell environments and its extensive feature set that facilitates automation [18].

Bash scripts automate and coordinate the entire process by:

- Managing the execution of SQL commands.
- Generating backups for specific databases, tables, or procedures.
- Automating the processing of large data volumes by splitting backups into separate files for easier management.

Sed (stream editor) is a UNIX utility for filtering and transforming text. It operates directly on text streams or files, making it highly efficient for processing large files without requiring manual opening and editing [19, 20, 21]. Sed can be used for tasks such as:

- Removing DEFINER clauses to avoid permission issues during restoration.
- Adding supplementary SQL commands (e.g., USE database).

This ensures script compatibility across different environments.

3.2. Backup Process with Flexmysqlbackup

The main steps involved in the backup process using the flexmysqlbackup method are:

- Selecting objects for backup (e.g., schemas, tables, data, procedures, functions, triggers, events, views).

- Choose the scope for the selected object type (e.g., a single specific object, all objects of that type, or a defined group of objects).
- Generating SQL scripts.
- Use INFORMATION_SCHEMA to query metadata and structure definitions.
- Process the generated scripts using sed.
- Execute the scripts and store the results.
- Save the generated SQL scripts into appropriately named files.
- Arrange the backup files within a specific directory structure for easy access and management.

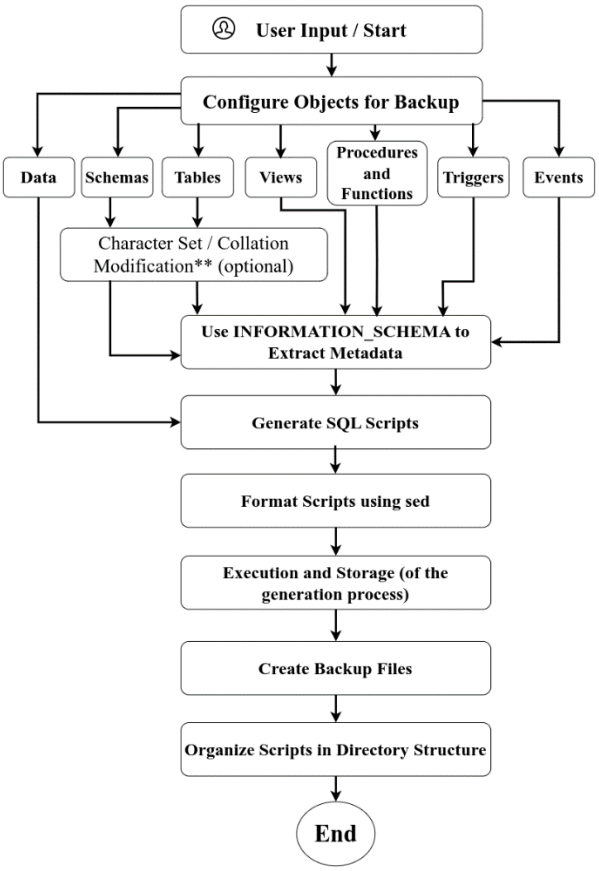


Figure 2. Backup process with flexmysqlbackup.

The primary advantages of flexmysqlbackup include:

- Selective Backup;
- Cross-Version Compatibility;
- Automation;
- Flexibility.

Selective Backup gives users the ability to choose precisely which objects to back up, such as:

- Specific tables from a given schema;
- Views and their dependencies;
- Stored procedures and functions.

This minimizes backup volume and restoration time.

Cross-Version Compatibility: Backups are automatically adapted for different MySQL versions, ensuring compatibility by:

- Removing unsupported or version-specific attributes.
- Ensuring consistency of character sets and collations during the restoration process.

Automation includes creating integrated backup processes using Bash:

- Simultaneous processing of multiple tasks, which reduces backup time.
- Easy integration into existing database management systems.

Flexibility offers the possibility to extend and customize the method according to the specific needs of the user:

- Archiving to specified engines, such as InnoDB or MyISAM.
- Changing character sets and collations during the process.

4. Implementation of Flexmysqlbackup

To implement the core object backup functionalities, stored procedures have been created for the following tasks:

1. Backing up schemas (database definitions);
2. Backing up table structures (CREATE TABLE statements);
3. Backing up table data (INSERT statements);
4. Backing up views;
5. Backing up procedures and functions;
6. Backing up triggers;
7. Backing up events.

For backing up procedures and functions, SQL queries implemented within a Bash script are used.

4.1. Backing up Schemas

The stored procedure gen_schemes_script is designed to generate an SQL script for creating databases (schemas) in MySQL, excluding the system databases (mysql, information_schema, performance_schema, sys). The procedure accepts an optional input parameter specifying the name of a single schema to process. If no schema name is provided, the procedure processes all non-system schemas.

The generated SQL code contains the necessary CREATE DATABASE statements.

An example Bash script command to generate a backup of all schemas and save it to a file named all_schemes.sql is:

```
$ mysql -hlocalhost -P3306 -udba -p --default-character-set=utf8mb4 --comments -e "CALL tools.gen_schemes_script(' ');" | sed '1d' > all_schemes.sql
Enter password: *******
```

The elements of the script are presented in table 3.

Table 3. Elements in the bash script for creating an archive of all schemas.

Element	Description
mysql	The MySQL client command used to interact with a MySQL server from the terminal.
-hlocalhost	The -h option specifies the host (server) the MySQL client will connect to; localhost indicates the server is running locally (on the same machine).
-P3306	The -P option specifies the port on which the MySQL server accepts connections. 3306 is the default port for MySQL.
-udba	The -u option specifies the username for the MySQL connection, which is dba in this example. *(Assuming 'dba' was a typo for 'dba')*
-p	The -p option indicates that a password is required. When the command is executed, it will prompt the user to enter the password for the dba user.
--default-character-set=utf8mb4	Specifies that the client should use utf8mb4 as the default character set. This is important to ensure correct encoding of text data when sending to and receiving from the MySQL server.
--comments	Instructs the MySQL client to preserve comments in SQL statements when they are sent to the server.
-e "CALL ..."	The -e option allows executing an SQL statement directly from the command line, in this case, "CALL tools.gen_schemes_script(' ');".
| sed '1d'	The symbol | (pipe) redirects the output from the mysql command to the sed command. In this case: '1d' instructs sed to delete the first line of the output, which usually contains the column headers returned by MySQL.
> all_dbs.sql	The > symbol redirects the result (after processing by sed) to a file named all_dbs.sql.

The generated SQL code for creating the databases is of the following form:

```
CREATE DATABASE IF NOT EXISTS `test_1`
CHARACTER SET utf8mb4 COLLATE utf8mb4_general_ci;
CREATE DATABASE IF NOT EXISTS `test_2`
CHARACTER SET utf8mb4 COLLATE utf8mb4_general_ci;
```

If an error occurs, the procedure returns a message containing an error code and a description.

4.2. Backing up Table Structures

The stored procedure gen_table_script generates SQL commands for creating tables, including column definitions, indexes, and keys. The following parameters can be provided to the gen_table_script procedure:

- database_name: The specific database to process, or all non-system databases if no name is specified.
- table_name: A specific table within the specified database. (If database_name is provided, this parameter allows targeting a single table within it)
- engine_name: If specified, only tables using this storage engine will be backed up.
- character set: If specified, the CREATE TABLE statement and string-based column definitions will be modified to use this character set.
- collation: If specified, the table default collation and column collations will be modified to use this collation.

By default, if the procedure is executed with empty parameters (e.g., CALL tools.gen_table_script(‘’, ‘’, ‘’, ‘’, ‘’);), it will back up the structure of all tables across all non-system databases.

The procedure can be executed with only the database_name parameter specified (e.g., CALL tools.gen_table_script('my_db', ‘’, ‘’, ‘’, ‘’);). In this case, it will generate a backup script for all tables within that specific database.

Specifying the table_name parameter requires that the database_name parameter also be provided (e.g., CALL tools.gen_table_script('my_db', 'my_table', ‘’, ‘’, ‘’);). It is not possible to specify only a table name without its database.

The engine_name parameter can be used independently. That is, if all other parameters are empty (‘’) and engine_name is set to, for example, INNODB or MYISAM (e.g., CALL tools.gen_table_script(‘’, ‘’, 'INNODB', ‘’, ‘’);), the procedure will generate a backup script for all tables (across non-system databases) that use the specified storage engine. This parameter can also be used in conjunction with database_name (e.g., CALL tools.gen_table_script('my_db', ‘’, 'INNODB', ‘’, ‘’);). In this scenario, it will create a backup script containing only those tables within the specified database that use the specified storage engine.

The character_set and collation parameters can be used in combination with the other parameters, or they can be left empty (‘’) if there is no need to modify the default character set and collation settings for the tables and columns during the backup generation.

An example command execution for the gen_table_script procedure to generate a backup of all table structures (across non-system databases) and save it to a file named all_tables.sql is:

```
mysql --default-character-set=utf8mb4 --comments -e "CALL tools.gen_table_script(' ', ' ', ' ', ' ', ' ');" | sed 's/\\n/\n/g' | sed '1d' > all_tables.sql
```

4.3. Backing up Table Data

The procedure gen_data_sql_script automates the creation of mysqldump commands specifically tailored for backing up data from a specific table, schema, or schemas. It provides high flexibility and is easily integrated into scripts. This procedure is executed on the source (primary) machine to generate a list of mysqldump commands required for creating the data backup. The parameters for the procedure are:

- ShemaName: The specific database (schema) to process, or all non-system databases if no name is specified.
- TableName: This parameter requires SchemaName to also be specified. It targets a specific table within the provided schema. If left empty (NULL) while SchemaName is specified, it defaults to including all tables within that schema.
- EngineName: This parameter functions identically to the corresponding parameter in the gen_table_script procedure, allowing filtering of tables based on their storage engine.

If an error occurs, the procedure returns a message containing an error code and a description.

The execution process can be understood in two main stages. First, a Bash script file (e.g., backup_data.sh) is created. Then, the gen_data_sql_script stored procedure is executed with the desired parameters. The output of this procedure, which consists of properly configured mysqldump commands for each targeted table or database, is redirected and saved into this .sh file. Finally, the generated Bash script (.sh file) is executed. Running this script executes the embedded mysqldump commands, which in turn produce the actual SQL file(s) containing the backed-up data (INSERT statements). An example execution sequence to create a data backup for all non-system databases is:

```
echo '#!/bin/bash' > all_data_script.sh
mysql  -h -P -u -p -e "call tools.gen_data_script(' ', ' ', ' ');" --silent
--raw --skip-column-names >> all_data_script.sh
```

The all_data_script.sh file, generated by executing the command that calls gen_data_sql_script, will contain content similar to this:

```
#!/bin/bash
mysqldump -P [port] -h [server_address] -u [username] -
p[password] --no-create-db --no-create-info --column-statistics=0
--skip-triggers --extended-insert [database_name_1]
[list_of_tables_1] > data_backup_db1.sql
mysqldump -P [port] -h [server_address] -u [username] -
p[password] --no-create-db --no-create-info --column-statistics=0
--skip-triggers --extended-insert [database_name_2]
[list_of_tables_2] > data_backup_db2.sql
# ... other mysqldump commands for different databases/tables as
generated by the procedure
```

Once the .sh file is created, it needs to be made executable. In Unix-based operating systems, this is done using the *chmod +x* command, as described in [22, 23].

If the system contains a large number of databases, and thus the generated .sh script produces many individual .sql backup files, it is recommended to distribute these generated SQL files across two or more directories. This organization allows the restoration process for these files to be performed in parallel (e.g., running multiple restore commands simultaneously, each targeting a different directory), which can significantly reduce the overall recovery time.

4.4. Backing up Views

The stored procedure gen_view_script.sql generates SQL commands (CREATE VIEW statements) for backing up views. The following parameters can be provided to the gen_view_script procedure:

- TableScheme: The specific database (schema) containing the views to back up. If left empty (‘’), it defaults to processing views across all non-system databases.
- ViewName: If this parameter is specified, the TableScheme parameter must also be provided. If both parameters are provided, only the specified view within the specified schema will be backed up. If ViewName is left empty (‘’), the procedure will back up all views either within the specified TableScheme (if provided) or across all non-system databases (if TableScheme is also empty).

An example command execution for the gen_view_script procedure to generate a backup of all existing views (across non-system databases) is:

```
$ mysql -hlocalhost -P3306 -udba -p --default-character-
set=utf8mb4 --comments -e "CALL tools.gen_view_script(' ', '
');" | sed 's/\\n/\n/g' | sed '1d' > sakila_views.sql
Enter password: *******
```

4.5. Backing up Procedures and Functions

The script file process_sql_opt.sh is used to generate SQL commands for backing up stored procedures and functions. It requires configuration settings to be filled in before it is executed.

Table 4. Parameters for the process_sql_opt.sh.

Parameter	Description
MYSQL_USER	The username for connecting to the database.
MYSQL_HOST	The address (hostname or IP) of the MySQL server.
MYSQL_PORT	The port number of the MySQL server.

Executing the process_sql_opt.sh script will display a menu on the screen. From this menu, the user must select the desired type of backup to create.

Table 5. Options for creating procedure and function backups.

Option	Description
1.All	Create a backup of all procedures and functions from all (non-system) databases.
2.Specific	Create a backup of only one specific procedure or function. Sub-options require specifying the Procedure/Function Name and the Database Name.
3.AllFromDb	Create a backup of all procedures and functions from a specific database. A sub-option requires specifying the Database Name.

An example diagram illustrating the execution process of the process_sql_opt.sh script for creating a backup of procedures and functions is provided in Figure 3.

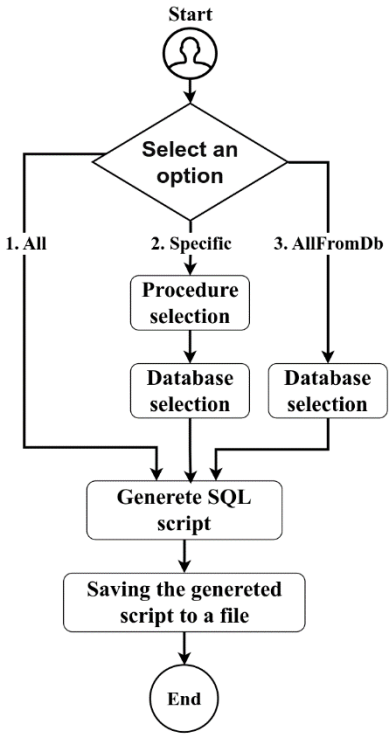


Figure 3. Procedure and Function Backup Process.

4.6. Backing up Triggers

The stored procedure gen_triggers_script generates SQL commands (CREATE TRIGGER statements) for backing up triggers. The following parameters can be provided to the procedure:

- ShemaName: The specific database (schema) containing the triggers to back up. If left empty (‘’), it defaults to processing triggers across all non-system databases.
- TriggerName: if this parameter is specified, the SchemaName parameter must also be provided. If both parameters are provided, only the specified trigger will be backed up. If TriggerName is left empty (‘’), the procedure will back up all triggers from the specified schema, or from all non-system schemas if none is provided.

An example command execution for the gen_triggers_script stored procedure to generate a backup of all triggers and save it to a file named all_triggers.sql is:

```
$ mysql -hlocalhost -P3306 -udba -p --comments -e "CALL
tools.gen_triggers_script(' ', ' ');" | sed 's/\\n/\n/g' | sed '1d' >
all_triggers.sql
Enter password: *******
```

4.7. Backing up Events

The stored procedure gen_event_script generates SQL commands (CREATE EVENT statements) for backing up scheduled events. The following parameters can be provided to the gen_event_script procedure:

- ShemaName: The specific database (schema) where the events are defined. If left empty (‘’), it defaults to processing events across all non-system databases.
- EventName: If this parameter is specified, the SchemaName parameter must also be provided. If both parameters are provided, only the specified event within the specified schema will be backed up. If EventName is left empty (‘’), the procedure will back up all events either within the specified SchemaName (if provided) or across all non-system databases (if SchemaName is also empty).

An example command execution for the gen_event_script stored procedure to generate a backup of all events (across non-system databases) and save it to a file named all_events.sql is:

```
$ mysql -hlocalhost -P3306 -udba -p --comments -e "CALL
tools.gen_event_script(' ', ' ');" | sed 's/\\n/\n/g' | sed '1d' >
all_events.sql
Enter password: *******
```

5. Experiments and Results

MySQL 5.1.6 and newer versions feature an event scheduler, which allows the execution of database operations activated at scheduled times. An event is essentially a stored program associated with a schedule. The schedule determines the time or time intervals during which the event executes, and optionally, when it expires (ceases to exist). Events are particularly useful for performing automated administrative tasks, such as periodic updates of summary reports, purging old data, or rotating log tables [12].

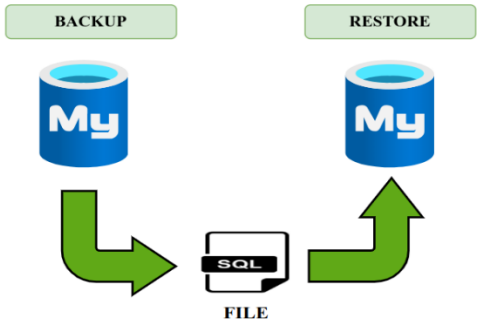


Figure 4. Simplified Backup and Restore/Deployment Model.

Figure 4 illustrates a simplified model for MySQL database backup/export and restore/import. The process begins with the backup (export) phase, where data and structure from the database are extracted and saved into an SQL file. This file contains a text-based representation of the database's structure and/or content. During the restore (import/deployment) phase, the SQL file is used to recreate the database in a new or existing MySQL instance.

This model ensures data portability and facilitates recovery in case of failures or migration to a new system. If the system utilizes automatic (scheduled) events that generate data, it is mandatory to execute the following command on both the source (backup) machine and the target (restore) ma-chine before creating the data backup and before restoring it:

```
SET GLOBAL event_schaduler = OFF
```

After the entire backup has been restored, execute the following command:

```
SET GLOBAL event_schaduler = ON
```

5.1. Example: Creating and Restoring a Full Backup

When creating a backup with flexmysqlbackup, the specific order in which the individual backup components (schemas, table structures, data, views, routines, etc.) are generated is generally not critical. However, to create a complete backup of the entire system, each relevant script or procedure call must be executed. The following sequence shows an example of how this can be done:

```
mysql -h -P -u -p -e "CALL tools.gen_schemes_script(' ');" | sed '1d' > all_schemes.sql
mysql -h -P -u -p -e "CALL tools.gen_table_script(1, ' ', ' ', ' ', ' ', ' ');" | sed 's/\\n/\n/g' | sed '1d' > all_tables.sql
mysql -h -P -u -p -e "CALL tools.gen_view_script(' ', ' ');" | sed 's/\\n/\n/g' | sed '1d' > all_views.sql
./process_sql_opt.sh
mysql -h -P -u -p -e "CALL tools.gen_triggers_script(' ', ' ');" | sed 's/\\n/\n/g' | sed 's/\\t//g' | sed '1d' > all_triggers.sql
mysql -h -P -u -p -e "CALL tools.gen_event_script(' ', ' ');" | sed 's/\\n/\n/g' | sed '1d' > all_events.sql
echo '#!/bin/bash' > all_data_script.sh
mysql  -h -P -u -p -e "call tools.gen_data_script(' ', ' ', ' ');" –silent --raw --skip-column-names >> all_data.sql
```

During the restoration (deployment) of the backup, the steps must follow a specific order because MySQL has dependencies between different database components. For example, a stored procedure cannot be created if its code references a table that does not yet exist. Similarly, a trigger cannot be created if the table it is attached to does not exist. The steps for restoring the backup, followed in the correct order, are:

```
mysql -h -P -u -p < all_schemes.sql
mysql -h -P -u -p < all_tables.sql
mysql -h -P -u -p < all_views.sql
mysql -h -P -u -p --comments < all_procedures.sql
mysql -h -P -u -p < all_data.sql
mysql -h -P -u -p < all_triggers.sql
mysql -h -P -u -p < all_events.sql
```

Figure 5 illustrates the process of database backup and restoration using SQL files. The diagram depicts the main stages of interaction between the user, the primary database server, the storage server, and the secondary (restore) server.

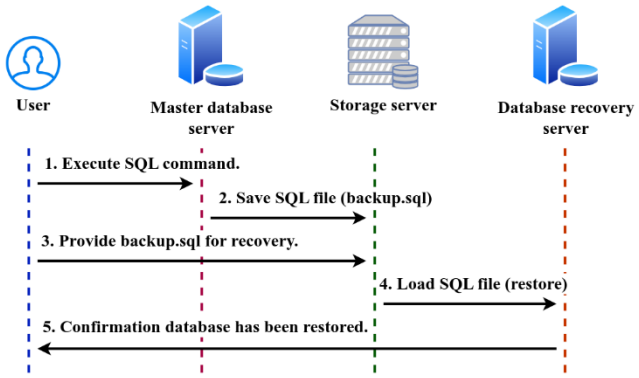


Figure 5. Unified Diagram of the Backup and Restoration Process.

The key steps in the backup and restoration process are:

- The user executes an SQL command to initiate the backup process.
- The primary database server generates and saves an SQL file – backup.sql – which contains the structure and data of the database.
- The backup file is transferred to and stored on a dedicated storage server.
- The backup.sql file is loaded onto a secondary database management server to perform the restoration.
- Finally, the successful restoration of the data is confirmed/verified.

This sequence provides clarity on the steps involved in backup management and recovery, ensuring data protection and minimizing potential downtime during database transfers between servers.

5.2. Experiments

The testing methodology consisted of repeatedly exporting and importing the Sakila database under controlled conditions. Each test measured the execution time and the success of full restoration under different MySQL versions and configurations. The tests were conducted in both clustered and standalone environments, as described in Section 5.2.1.

The sample Sakila database was developed by the MySQL team. It was created as a more complex alternative to the previous "world" sample database and serves for training purposes, as well as for testing SQL queries. The database can be freely downloaded and used from the official MySQL website [24].

Additionally, five events were added to the Sakila database. Two bash scripts were created for automated backup and restoration of the Sakila database. One script uses mysqldump, and the other uses flexmysqlbackup.

The scripts also incorporate *bc* (Basic Calculator), a powerful command-line utility for performing high-precision arithmetic calculations. It supports operations with integers and floating-point numbers, as well as complex mathematical functions, and can be used both interactively and within scripts [25]. In this context, *bc* is used solely for the purpose of the experiments: to measure the time taken for backup creation and restoration by each method, calculate the arithmetic mean of these times over multiple runs, and subsequently compare the performance values between flexmysqlbackup and mysqldump.

The testing process for each method (mysqldump and flexmysqlbackup) involved the following steps:

- Creating a backup of the sample sakila database.
- Restoring the database from the created backup.
- Comparing the results using the following metrics:
  - Size of the generated SQL backup file(s);
  - Time taken for the backup process;
  - Time taken for the restoration process.

It is important to note that the mysqldump utility varies between different MySQL versions and is optimized for the specific version it accompanies. Backups created with mysqldump from one version might not be restorable on another MySQL version without manual modifications to the script. In contrast, flexmysqlbackup is designed as a universal tool that can be used with various MySQL versions, and the backups created with it can be restored across different versions, potentially including future ones.

5.2.1. Test Environment and Hardware Configuration

Tests with MySQL 5.6 were conducted in a clustered environment with the NDB Storage Engine, based on MySQL 5.6.35 with NDB Cluster 7.4.14.

Cluster architecture:

- Data Node (ndbd): 2 nodes;
- Management Node (ndb_mgmd): 1 node;
- MySQL API Node (mysqld): 1 active node;
- Total: 2 data nodes, 1 MGM, 1 MySQL API node;
- The machines were virtualized using KVM/QEMU.

Memory usage in the NDB cluster (retrieved with ndb_mgm):

- Node 2: Data usage is 63%;
- Node 2: Index usage is 48%;
- Node 3: Data usage is 63%;
- Node 3: Index usage is 48%.

Table 6. Hardware configuration of MySQL nodes.

Compo-nent	CPU (model and cores)	RAM	Disk (model and size)	Disk usage
MySQL API Node	Common KVM processor, 2 cores	23 GiB	QEMU HARDDISK, 150 GB	~30 GB out of 142 GB (approx. 21%)
Manage-ment Node	Common KVM processor, 2 cores	2 GiB	QEMU HARDDISK, 30 GB	18 GB out of 27 GB (67%)
Data Nodes (1 and 2)	Common KVM processor, 4 cores	16 GiB	QEMU HARDDISK, 150 GB	106 GB out of 147 GB (72%)

Tests with MySQL 8.0.40 were conducted on a separate physical machine without NDB Cluster. Hardware configuration:

- System: Micro-Star International Prestige 15 A11SCX;
- RAM: 64 GiB (2x32GiB DDR4 3200 MHz);
- CPU: 11th Gen Intel(R) Core(TM) i7-1185G7 @ 3.00GHz;
- Cache: 192 KiB L1, 5 MiB L2, 12 MiB L3;
- Disk: 1 TB NVMe (Phison SM2801T24GKBB4S-E162);
- Kernel: Linux 6.11.0-26-generic.

Table 7. Operating Systems.

Cluster version	Component	Operating system
NDB 7.4.14	MySQL API Node	Debian GNU/Linux 8 (jessie)
	MGM Node	Debian GNU/Linux 9 (stretch)
	Data Node 1	Debian GNU/Linux 9 (stretch)
	Data Node 2	Debian GNU/Linux 9 (stretch)
MySQL 8.0.40	Standalone	Ubuntu 22.04.1 LTS

5.2.2. Test 1

Test 1 was conducted on MySQL version 8.0.40. The execution times for individual parts of the process and the sizes of the resulting backup scripts are presented in Tables 8 and 9, respectively.

Table 8. Test 1 Results – Time Metrics.

Metric	Mysqldump (sec)	Flexmysqlbackup (sec)	Difference (sec)	Percentage Difference
Average creation time	0.247	0.191	-0.056	-22.67% (faster)
Average restore time	1.578	1.584	+0.006	+0.38% (slower)
Average total time (create + restore)	1.825	1.776	-0.049	-2.68% (faster)

Flexmysqlbackup is significantly faster during the creation phase (≈22.7% faster). Restoration time is nearly identical for both methods, with flexmysqlbackup being marginally slower by only 0.38%. The overall time (creation + restore) is, on average, 2.68% better (faster) when using flexmysqlbackup.

The data files, which consist of INSERT statements, are identical in size for both methods. For the other database objects – procedures, triggers, events, table structures, and views – flexmysqlbackup demonstrates an advantage, consistently producing smaller backup files (table 9).

Table 9. Test 1 Results – Backup File Sizes.

Category	Mysqldump (KB)	Flexmysqlbackup (KB)	Reduction (%)
Data	3276.8	3276.8	0.00%
Stored Procedures & Functions	12	6.1	49.17%
Events	7.3	1.6	78.08%
Tables & Views (structure)	30	13	56.67%
Triggers	7.1	5.8	18.31%

5.2.3. Test 2

Test 2 was conducted on MySQL 5.6. In this test, flexmysqlbackup was significantly slower during the creation phase – more than 3 times slower. However, the restoration process was approximately 19% faster with flexmysqlbackup. The overall time (creation + restore) was more than twice as long when using flexmysqlbackup compared to mysqldump.

Further analysis revealed that the reason for the slowdown lies within the script responsible for generating table structures, specifically the part that constructs the foreign key constraints. When executing queries that retrieve foreign key information from INFORMATION_SCHEMA (using the REFERENTIAL_CONSTRAINTS and KEY_COLUMN_USAGE tables), significantly lower performance is observed in MySQL 5.6 compared to MySQL 8.0. The primary reason is that in MySQL 5.6, INFORMATION_SCHEMA is implemented using dynamically generated, non-indexed views. In contrast, MySQL 8.0 stores metadata in a centralized and indexed InnoDB-based structure known as the Data Dictionary, which considerably speeds up access [26].

Table 10. Test 2 Results – Time Metrics.

Metric	Mysqldump (sec)	Flexmysqlbackup (sec)	Difference (sec)	Percentage Difference
Average creation time	1.441	6.043	+4.602	+319.3% (slower)
Average restore time	1.431	1.160	-0.271	-18.94% (faster)
Average total time (create + restore)	2.584	7.204	+4.620	+178.8% (slower)

In a specific test case, a query retrieving foreign keys executed in approximately ~1 second on MySQL 5.6, compared to ~0.1 seconds on MySQL 8.0 - demonstrating roughly a 90% faster execution time on the newer version.

5.2.4. Test 3

In Test 3, a backup created on MySQL 8.0 was restored onto MySQL 5.6 (Table 11). It is important to note that for this test, the backup created using mysqldump required manual modifications. Specifically, the collations for tables, columns, procedures, triggers, and events had to be changed from utf8mb4_0900_ai_ci to utf8mb4_general_ci. This is typically done using a command like:

```
sed -i 's/utf8mb4_0900_ai_ci/utf8mb4_general_ci/g' *.sql
```

Furthermore, it's important to highlight that the address table contains a column named location defined with the geometry data type, and this column also has a SPATIAL index defined on it. Restoring this on MySQL 5.6 requires additional manual script transformations, including ensuring the table uses the MyISAM engine (as SPATIAL indexes were primarily supported on MyISAM in versions up to 5.6) and potentially adjusting the column and index definitions appropriately.

In contrast to mysqldump, flexmysqlbackup did not require these manual script transformations for the restoration. In the conducted experiment, disregarding the backward compatibility issues and the manual effort required for the mysqldump backup when migrating from version 8 to 5.6, the restoration time for both methods was found to be comparable (Table 11).

Table 11. Test 3 Results – Restore Time (MySQL 8.0 backup to MySQL 5.6).

Metric	Mysqldump (sec)	Flexmysqlbackup (sec)	Difference (sec)	Percentage Difference
Average restore time	1.111	1.150	+0.039	+3.51% (slower)

In Test 3, where a backup from MySQL 8.0 was restored onto MySQL 5.6, flexmysqlbackup was approximately 3.51% slower during the actual restoration process compared to mysqldump. However, an important clarification is that mysqldump required manual corrections (for collations, data types, and indexes), which makes it less convenient in this scenario, despite the slight time advantage.

5.2.5. Test 4

In test 4 backup created on MySQL 5.6 was restored onto MySQL 8.0 (Table 12).

Table 12. Test 4 results.

Metric	Mysqldump (sec)	Flexmysqlbackup (sec)	Difference (sec)	Percentage Difference
Average restore time	1.631	1.510	-0.121	-7.42% (faster)

In Test 4, flexmysqlbackup was 7.42% faster during the restoration process.

6. Future Work and Community Involvement

Further development of the flexmysqlbackup tool is planned, with a focus on higher automation, performance improvement, and fixing existing bugs. Expanding its functionality is planned through the creation of a user-friendly graphical user interface (GUI) to facilitate its use by a wider range of users, including those without advanced technical skills.

Tests have also been performed with MySQL version 8.4.4, where flexmysqlbackup demonstrated excellent stability and compatibility. Additional tests are planned with newer versions, including MySQL 9.3, aimed at ensuring maximum compatibility and reliability.

The goal is for the tool to become a stable, widely used solution that system administrators and developers can rely on in all kinds of environments – from small installations to corporate infrastructures. The project is already publicly available on GitHub, where community participation is encouraged. Suggestions, bug reports, and pull requests are welcomed from anyone wishing to contribute to the future development and improvement of the tool.

7. Conclusion

Flexmysqlbackup represents a significant advancement in database backup management, overcoming key limitations inherent in the standard mysqldump utility. It provides the necessary flexibility to adapt to the specific needs of various projects and organizations.

One of its primary advantages is the capability for selective backup. This significantly reduces restoration times and optimizes resource utilization. In complex systems where only critical information needs to be backed up, the ability to select specific objects – such as tables, triggers, or views – eliminates unnecessary processing and storage overhead.

Cross-version compatibility between different MySQL releases is another central aspect of flexmysqlbackup. By eliminating the need for manual intervention to handle differences in collations or version-specific parameters, administrators can perform migrations with minimal disruption and high confidence in successful recovery. This is particularly crucial for large companies that rely on complex and critical databases for their daily operations.

References

[1] Reese, G. (2007). *MySQL pocket reference: SQL functions and utilities* (pp. 37–41). O'Reilly Media, Inc.

[2] Norman, M. (2004). Changing data. In *Database design manual: Using MySQL for Windows* (pp. 87-101). Springer. https://doi.org/10.1007/1-85233-859-8_8

[3] Schwartz, B., Zaitsev, P., & Tkachenko, V. (2012). *High performance MySQL: Optimization, backups, and replication* (pp. 622–623). O'Reilly Media, Inc.

[4] Kofler, M. (2005). *mysql, mysqladmin, and mysqldump.* In *The Definitive Guide to MySQL 5* (pp. 61–70). https://doi.org/10.1007/978-1-4302-0071-0_4

[5] MySQL Documentation. (n.d.). *Backup and recovery*. MySQL 8.0 Reference Manual. Retrieved March 20, 2025, from https://dev.mysql.com/doc/refman/8.0/en/backup-and-recovery.html

[6] Reese, G. (2003). *MySQL pocket reference* (pp. 3, 61). O'Reilly Media, Inc.

[7] Kromann, F. M. (2018). MySQL views. In *Beginning PHP and MySQL* (pp. 733-750). Apress. https://doi.org/10.1007/978-1-4302-6044-8_31

[8] MySQL Documentation. (n.d.). *mysqldump — A database backup program*. MySQL 8.0 Reference Manual. Retrieved from https://dev.mysql.com/doc/refman/8.0/en/mysqldump.html

[9] MySQL. (n.d.). *Upgrading to MySQL 8.0? Here is what you need to know...* Retrieved from https://dev.mysql.com/blog-archive/upgrading-to-mysql-8-0-here-is-what-you-need-to-know/

[10] Christudas, B. (2019). *MySQL* (pp. 877–884). Apress. https://doi.org/10.1007/978-1-4842-4501-9_27

[11] Dyer, R. (2008). *MySQL in a nutshell* (pp. 68, 105–109, 235, 315–317, 323, 362–363). O'Reilly Media, Inc.

[12] DuBois, P. (2013). *MySQL* (pp. 212–213, 684–685, 1005, 1078, 1110, 1148). Addison-Wesley. ISBN 9780321833877.

[13] Schwartz, B., Zaitsev, P., & Tkachenko, V. (2012). *High performance MySQL: Optimization, backups, and replication*. O'Reilly Media, pp. 298–305.

[14] Bell, C., Kindahl, M., & Thalmann, L. (2014). *MySQL high availability: Tools for building robust data centers*. O'Reilly Media.

[15] Tahaghoghi, S. M. M., & Williams, H. E. (2006). *Learning MySQL*. O'Reilly Media, Inc. ISBN 9780596008642.

[16] Krogh, J. W. (2020). *MySQL 8 query performance tuning: A systematic method for improving execution speeds* (pp. 111–131). Apress. https://doi.org/10.1007/978-1-4842-5584-1

[17] Harrison, G., & Feuerstein, S. (2006). *MySQL Stored Procedure Programming: Building High-Performance Web Applications in MySQL*. O'Reilly Media.

[18] Newham, C. (2005). *Learning the bash shell: Unix shell programming* (p. 16). O'Reilly Media.

[19] Richardson, L. (2005). Text manipulation with sed. *Linux Journal*, Slashdot Media.

[20] Natarajan R. (2007). *Sed and awk 101 hacks*. Publ. The Geek Stuff.

[21] Dougherty, D., & Robbins, A. (1997). *sed & awk: UNIX power tools*. O'Reilly Media, Inc.

[22] Shotts, W. (2012). *The Linux command line: A complete introduction* (pp. 81–87). No Starch Press.

[23] Nemeth, E., Snyder, G., Hein, T. R., Whaley, B., & Mackin, D. (2018). *UNIX and Linux system administration handbook* (pp. 156–158). USENIX Open Access Policy, 59.

[24] MySQL Documentation. (n.d.). *Sakila sample database*. Retrieved April 1, 2025, from https://dev.mysql.com/doc/sakila/en/

[25] Blum, R. (2008). *Linux command line and shell scripting bible* (p. 228). Wiley Publishing, Inc. ISBN 978-0-470-25128-7.

[26] MySQL. (n.d.). *MySQL 8.0 Reference Manual: MySQL data dictionary*. Retrieved from https://dev.mysql.com/doc/refman/8.0/en/data-dictionary.html

Computer Science and Interdisciplinary Research Journal
Volume 2, Issue 1, 2025. ISSN: 3033-1218, DOI: 10.70862/CSIR.2025.0201-11

Cycle Forecasting in Financial Markets with LSTM and Discrete Fourier Transform

Kostadin Yotov

University of Plovdiv Paisii Hilendarski, 236, Bulgaria Blvd, 4027 Plovdiv, Bulgaria

kostadin_yotov@uni-plovdiv.bg

Abstract – The study of cyclicality in capital markets is important for forecasting the prices of traded financial instruments and improving investment strategies. Cycles can be influenced by various factors – economic conditions, business activity, political events, and even market participants' psychology – making their analysis challenging. This paper presents a study aimed at detecting cyclicality in time series using a hybrid model. Recurrent neural networks of the Long Short-Term Memory (LSTM) type are combined with Discrete Fourier Transform for frequency component analysis. The model is trained on simulated market data containing clearly defined sinusoidal cycles and stochastic noise to assess the LSTM network's ability to capture the period, frequency, and amplitude of market fluctuations. Experiments were conducted with varying numbers of hidden neurons in the LSTM layer, and the results show that increasing architectural complexity improves amplitude prediction, while frequency and period are captured with high accuracy across all configurations.

Keywords – financial markets, cyclicality, LSTM, spectral analysis, Fourier transforms.

1. Introduction

Forecasting the cyclicality of financial markets is of paramount importance in today's dynamic world, where an increasing number of individuals invest in stocks and other financial instruments. It enables stakeholders to better understand and anticipate market behavior. Recognizing market cycles helps investors manage their portfolios more effectively, increase returns, and minimize risk. Moreover, financial markets often reflect future economic changes in advance. Early detection of recession signals allows for loss minimization through hedging or portfolio rebalancing.

The cyclicality of capital markets depends on numerous factors. A wide range of circumstances and cycles – economic, political, monetary, behavioral, and others – simultaneously and variably influence the movement of financial instrument prices. Some cycles are short-term, lasting days or weeks, while others are long-term, spanning years or even decades. Additionally, financial data is characterized by a high level of noise generated by random, irrational, or insignificant events. The advancement of computational technologies and artificial intelligence in recent years has provided a foundation for conducting more complex and resource-intensive analyses, such as wavelet transformations, Monte Carlo simulations, artificial neural networks (ANNs), and others. New models inspired by physics, biology, and engineering are being developed. However, no universal tool, algorithm, or technology can capture all aspects of cyclicality. Some models perform better during periods of growth, others during crises; some are more effective at capturing short-term fluctuations, while others are better suited for long-term trends; some are sensitive to noise, while others are not, and so on. This drives the development and investigation of complex models that combine different methods and algorithms and demonstrate adaptability to varying conditions.

This paper discusses several fundamental methods for analyzing the cyclicality of financial markets and approaches for time series analysis. It also proposes and investigates a hybrid model based on a recurrent neural network with Long Short-Term Memory (LSTM) and spectral analysis using Fourier

DOI: https://doi.org/10.70862/CSIR.2025.0201-11

Corresponding author: Kostadin Yotov,
University of Plovdiv Paisii Hilendarski
Email: kostadin_yotov@uni-plovdiv.bg

Received: 22 May 2025.
Revised: 5 June 2025.
Accepted: 6 June 2025.
Published: 6 June 2025.

© 2025 Kostadin Yotov. This work is licensed under the Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License.

The article is published with Open Access at http://csirjournal.com/

transforms. The study addresses the following research questions:

- To what extent can LSTM networks trained on noisy, simulated cyclical data accurately capture the period and frequency of the underlying cycles?
- How does the number of hidden neurons in the LSTM architecture affect the prediction of the amplitude of cyclical patterns?
- Can the integration of DFT with LSTM improve the model's ability to detect frequency components that may remain hidden in the residual error of time series forecasts?

By addressing these questions, the study aims to evaluate the effectiveness and limitations of neural-based hybrid models in identifying cyclical structures within financial time series, contributing to the broader field of AI-assisted financial forecasting

2. Cyclicality in Financial Markets and Standard Tools for Its Analysis

The complex nonlinear, chaotic, and stochastic behavior of financial markets requires a rich set of tools for their analysis. A wide variety of instruments and technologies are used to study cyclicality, ranging from empirical methods to machine learning and ANNs.

2.1. Technical Indicators

Some of the most popular technical indicators used for analyzing and forecasting financial markets – and in particular their cyclicality – are moving averages. The Simple Moving Average (SMA) is calculated using the following formula:

$$SMA = \frac{1}{n}\sum_{i=1}^{n} P_i, \quad (1)$$

where n is the number of observations (periods), and P_i are the asset values for each of those periods. The SMA determines the average value of the asset over a specific time period and is commonly used to smooth out price fluctuations and detect cycles or simply identify trend changes [1–4].

On the other hand, the Exponential Moving Average (EMA) assigns more weight to recent values compared to older data. This makes it more sensitive to recent price changes than the SMA. For the current period t, the EMA is represented as:

$$EMA_t = \alpha P_t + (1-\alpha) EMA_{t-1}, \quad (2)$$

where EMA_t and EMA_{t-1} are the exponential moving averages for the current and previous periods, respectively. The smoothing coefficient, which generally takes the form:

$$\alpha = \frac{2}{n+1} \quad (3)$$

is designed to control the weighting of recent data (see Figure 1).

In their work, Karasu and Altan use exponential moving averages to generate features required for a crude oil price forecasting model that incorporates LSTM [5]. On the other hand, Lucas and Zhang utilize an extension of the Exponentially Weighted Moving Average (EWMA) and develop a methodology for modeling volatility using a recursive updating scheme [6].

The Relative Strength Index (RSI) is used to analyze the momentum of price movements. It helps identify whether an asset is overbought or oversold and can be used to detect the beginning or end of a cycle in the behavior of a financial or commodity asset. RSI is calculated based on the ratio between average gains (AG) and average losses (AL) over a given period:

$$RSI = 100 - \frac{100}{1+\frac{AG}{AL}}, \quad (4)$$

where:

$$AG = \frac{1}{n}\sum_{i=1}^{n} AG_i\,, AL = \frac{1}{n}\sum_{i=1}^{n} AL_i. \quad (5)$$

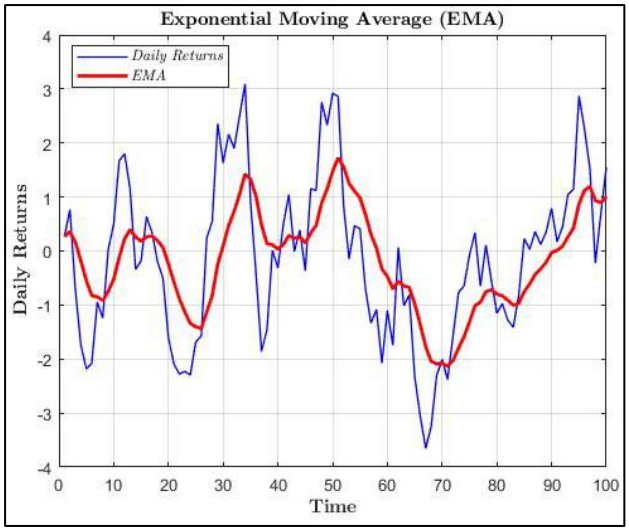


Figure 1. Example of an Exponential Moving Average for the daily return of a given asset (Matlab v.2018a).

RSI is one of the most commonly used indicators in technical analysis because it provides clear signals for overbought or oversold conditions, helping traders make informed decisions about market entry or exit. Examples of RSI applications can be found in [7–10].

Another popular technical tool that measures the momentum of price movements and is used to detect overbought or oversold market conditions is the Stochastic Oscillator [11–14]. It is based on the assumption that, in an uptrend, prices tend to close near the high of the price range for a given period, while in a downtrend, they tend to close near the low.

The Stochastic Oscillator consists of two lines:

- $\%K$ line (fast line): the main line that measures the current price relative to the price range.
- $\%D$ line (slow smoothed line): used as a signal line for $\%K$.

The mathematical model of the $\%K$ line expresses the position of the current price as a percentage of the price range over the past n periods:

$$\%K = 100 \times \frac{C - L_n}{H_n - L_n}, \quad (6)$$

where:

- C is the current closing price;
- L_n is the lowest price over the past n periods;
- H_n is the highest price over the past n periods;
- n is typically 14 but may vary depending on the analyst's preferences.

The formula for the $\%D$ line is based on the measured values of $\%K$ and is simply defined as a simple moving average of $\%K$ over the last m periods:

$$\%D = \frac{\sum \%K}{m}. \quad (7)$$

The following thresholds are commonly used when interpreting the Stochastic Oscillator to identify overbought or oversold conditions:

- Overbought conditions: When the Stochastic Oscillator (especially $\%K$) is above 80, the market is considered overbought, which may indicate a potential reversal or downward correction.
- Oversold conditions: When the Stochastic Oscillator value is below 20, the market is considered oversold, which may indicate a potential upward reversal.

Based on this, the following trading signals are formulated:

- Buy signal: When $\%K$ crosses above $\%D$ and both lines are in the oversold zone (below 20).
- Sell signal: When $\%K$ crosses below $\%D$ and both lines are in the overbought zone (above 80).

2.2. Time-Based Analysis of Market Cyclicality

Time-based analysis of market cyclicality is the process of examining time series data to identify and measure recurring patterns or cycles in market movements. The goal is to understand how and when certain market trends emerge that follow predictable patterns, including upward and downward movements, periods of volatility, and long-term trends. This analysis supports improved forecasting and decision-making in portfolio management, trading, and financial planning [15–18].

The main components of time-based cycle analysis include:

1. Identifying the cycle length: The analysis focuses on determining the periodicity of different cycles – whether they are short-term, medium-term, or long-term. Short-term cycles may span hours or days, medium-term cycles span months, and long-term cycles can last for years.
2. Estimating the cycle amplitude: Amplitude describes the price fluctuations between the peaks and troughs of a cycle and provides insight into the extent of variation, which is useful for risk assessment.
3. Analyzing the cycle phase: The cycle phase indicates whether the market is in an upward or downward trend. This is useful for predicting turning points where one trend ends and another begins, as well as for identifying support and resistance zones in market prices.
4. Identifying seasonal and temporal anomalies: Seasonal patterns – such as increased activity at the end of the year or summer downturns – are often revealed in time-based analysis and are significant for financial planning and forecasting.

A commonly used approach in cyclicality analysis is fractal time analysis of the market, where financial data is examined through the lens of fractal geometry [19–22]. This approach is based on the notion that market price movements are not purely linear and often exhibit recurring, self-similar structures across different time scales. Unlike traditional analysis, which assumes market movements are "smooth" and linear, fractal analysis is grounded in the understanding that they are often chaotic and irregular. Instead of following perfect cycles, markets move in complex patterns that nevertheless repeat with a certain degree of regularity.

Fractal analysis employs various mathematical methods and models to uncover the cyclicality of financial markets. One such model is the fractal dimension – a mathematical measure that describes how “complex” the structure of a market time series is [23,24]. Figure 2 presents a visualization demonstrating how specific segments of a time series can be analyzed through the prism of fractal geometry in order to identify self-similar structures in market movements.

One of the commonly used methods for measuring fractal dimension is the Correlation Dimension method.

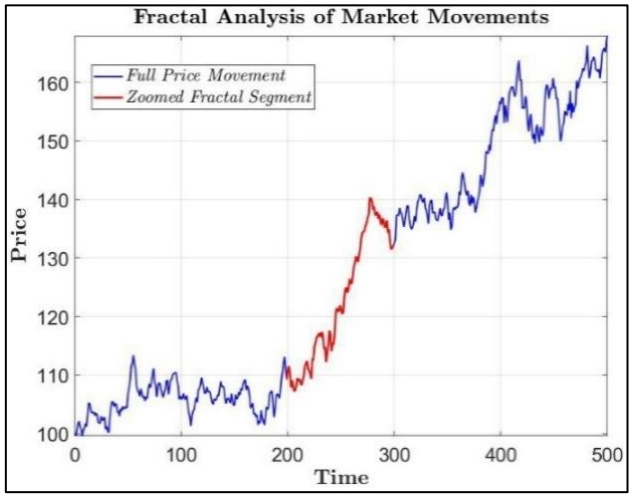


Figure 2. Example of fractality in market movement (Matlab v.2018a).

On the other hand, the Hurst exponent H is a fundamental tool for analyzing market cyclicality and predictability through fractals, as it measures the degree of long-term dependence in the data. It is calculated using the formula:

$$H = \lim_{n\to\infty} \frac{\log(R/S)}{\log(n)}, \quad (8)$$

where R is the range of deviations in values over a period n, and S is the standard deviation.

The following criteria are used when interpreting the Hurst exponent:

- If $H = 0.5$, the market behaves randomly and lacks predictability.
- If $H > 0.5$, the data exhibits a trend (positive autocorrelation), suggesting that previous movements are likely to continue in the same direction.
- If $H < 0.5$, the market shows mean-reverting behavior (negative autocorrelation), meaning frequent reversals in movement are more likely.

Examples of the application of the Hurst exponent H in the financial world can be found in studies such as [25–28].

In the 1930s, a concept in technical analysis became popular that describes recurring market patterns through a series of waves. This is the Elliott Wave Theory, which proposes that market movements reflect the collective psychology of investors and manifest in specific cycles and wave structures. According to Elliott, these patterns are fractal in nature and appear across multiple time scales.

The key principles of the Elliott Wave Theory include (Figure 3):

- The basic wave structure: According to the theory, market movements consist of impulse waves (moving in the direction of the main trend) and corrective waves (moving in the opposite direction). Impulse waves contain five sub-waves and follow the direction of the trend. Corrective waves consist of three sub-waves and correct the impulse movement, typically moving against the main trend.
- Impulse structure (5-3 wave pattern): A full cycle consists of 5 impulse waves (1, 2, 3, 4, and 5), followed by 3 corrective waves (A, B, and C). This means a complete market cycle comprises 8 waves – 5 upward (in the direction of the trend) and 3 corrective (in the opposite direction).

According to the theory, Elliott waves are fractal, meaning that each wave can be broken down into smaller waves that also follow the same 5-3 structure. This enables the theory to be applied across different time scales, from minute charts to yearly time frames. Elliott waves can be used to analyze cyclicality by identifying the sequence of waves that reflect recurring market behaviors. According to Elliott, when the market is trending, it passes through predictable phases of growth and correction, which are part of the natural market cycles.

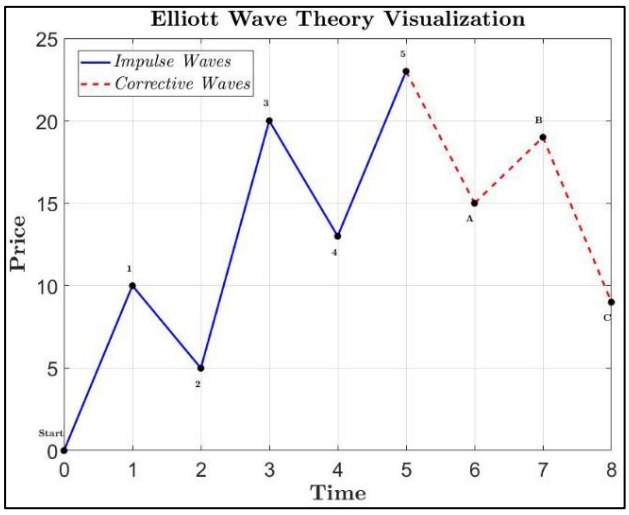


Figure 3. Fundamental principles of Elliott Wave Theory (Matlab v.2018a).

In fractal analysis, the Fractal Wave Theory is also used. It builds upon the concept of Elliott Wave Theory and is based on the idea that market movements consist of multiple waves of varying duration, appearing across different time scales and exhibiting similar structures. This theory is employed to identify patterns and cycles at different temporal levels [29–32].

On the other hand, Multifractal Time Analysis uses multifractals to examine the markets. It allows for the detection of different fractal dimensions across various time intervals. This method provides a more detailed understanding of market complexity and the

ability to study different cycles at different time scales [33–36].

A powerful technique within time-based analysis of market cyclicality is Fibonacci time extensions (Fibonacci extension levels). These utilize the Fibonacci sequence to forecast time points at which significant market changes – such as trend reversals or state transitions – are likely to occur [37–40]. Unlike traditional Fibonacci levels, which focus on price ranges, Fibonacci time extensions consider the temporal component of cyclicality and offer a method for analyzing the timing of trend reversals and major price movements. The idea is based on the Fibonacci sequence (1, 1, 2, 3, 5, 8, 13, 21, ...), where each number is the sum of the two preceding ones. This sequence produces ratios such as 0.618 (61.8%), 1.618 (161.8%), and 2.618 (261.8%), which are widely used in technical analysis. In time extensions, a key market cycle is typically identified (e.g., a move from point A to point B), and then Fibonacci time ratios are applied to determine when significant market shifts are likely to occur.

2.3. Statistical Models for Analyzing Market Cyclicality

Statistical models for analyzing the cyclicality of financial markets are methods that utilize historical data to forecast future movements. These models focus on identifying recurring patterns and cycles. Their goal is to uncover hidden cyclical structures in the data using various statistical and mathematical techniques.

One commonly used approach in this context is the $ARIMA$ model. This model combines three core components:

- AR (Autoregressive): The autoregressive component uses the relationship between the current value and previous values in the time series.
- I (Integrated): The integration component removes trends in the data through differencing.
- MA (Moving Average): The moving average component is based on the relationship between the current value and errors from previous forecasts.

The $ARIMA$ model is denoted as $ARIMA(p, d, q)$, where p is the order of the autoregression, d is the degree of differencing, and q is the order of the moving average. It is defined by the following equation:

$$Y_t = c + \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + \cdots + \phi_p Y_{t-p} + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \cdots \theta_q \varepsilon_{t-q}, \quad (9)$$

where:

- Y_t is the value of the time series at time t;
- c is a constant;
- ϕ_i are the autoregressive coefficients;
- ε_t is the random error (noise) at time t;
- θ_i are the moving average coefficients.

In study [41], the $ARIMA$ model approach is used to forecast the stock prices of pharmaceutical companies, while in [42] it is applied to predict financial anomalies and trends in corporate financial development, helping companies manage risk and make informed investment decisions.

A widely used extension of $ARIMA$ is the $SARIMA$ model, which incorporates seasonal components and is especially useful for analyzing cyclicality in market data with pronounced seasonality [43–46]. This model adds seasonal parameters to $ARIMA$ using the structure $SARIMA(p, d, q)\ (P, D, Q)_s$, where P, D, Q are the seasonal counterparts of the autoregressive, differencing, and moving average components, and s is the seasonal period (e.g., $s = 12$ for yearly seasonality in monthly data).

The $SARIMA$ formula incorporates both seasonal AR and MA components, as follows:

$$Y_t = c + \phi(B)Y_{t-1} + \Phi(B^s)Y_{t-s} + \varepsilon_t + \theta(B)\varepsilon_{t-1} + \Theta(B^s)\varepsilon_{t-s}, \quad (10)$$

where:

- B is the lag operator for the time series;
- $\phi(B)$ is the autoregressive operator;
- $\Phi(B^s)$ is the seasonal autoregressive operator;
- $\theta(B)$ is the moving average operator;
- $\Theta(B^s)$ is the seasonal moving average operator.

Examples of $SARIMA$ applications in financial event analysis can be found in the works of [42, 44–46].

Another approach within the framework of statistical models for analyzing market cyclicality is the Hodrick–Prescott Filter (HP Filter), which decomposes time series into trend and cyclical components (Figure 4). It is particularly popular in macroeconomic analysis and is useful for isolating long-term trends from short-term cycles in the data. The HP filter aims to minimize the following expression:

$$\min_{T_t} A = \min_{T_t} \sum_{t=1}^{T} (Y_t - T_t)^2 + \lambda \sum_{t=2}^{T-1} [(T_{t+1} - T_t) - (T_t - T_{t-1})]^2, \quad (11)$$

where:

- Y_t is the value of the time series at period t;
- T_t is the long-term trend component of the time series;

- λ is a parameter that controls the degree of smoothing of the trend (higher values of λ result in a smoother trend).

The first term, $(Y_t - T_t)^2$, minimizes the deviation between the original data and the trend component to keep the trend close to the actual data. The second term controls the smoothness of the trend, ensuring it is stable and gradual.

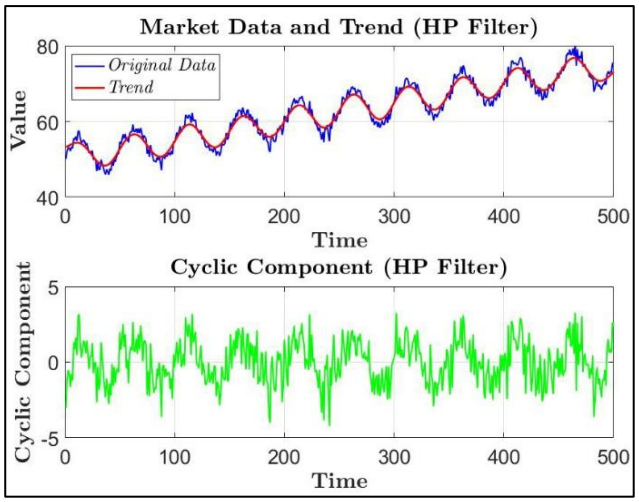


Figure 4. Hodrick–Prescott filter for decomposing a time series into trend and cyclical components (Matlab v.2018a).

In their work, Ilyas et al. use a modified Hodrick–Prescott filter to smooth historical stock price data by removing the cyclical component from the time series [47]. Quande et al. demonstrate how novel filter-based modeling with Hodrick–Prescott (HP) filter, that can identify repeated up and down structural features around a certain carbon price, negotiates the obstacle of the parallel–series hybridization concerning the linear and the nonlinear model [48]. Other notable applications of the HP Filter are presented in [49–52].

2.4. Cyclical Indices and Indicators

Several economic and financial indices can also be used to identify market cyclicality. For example, business activity indices, the Purchasing Managers' Index (PMI), employment data, and inflation figures are key economic indicators that provide insight into the state of the economy and are used to assess economic cycles. These indicators help to determine the current phase of the economy (expansion, slowdown, or recession) and to forecast future trends [53–56].

Business activity indices reflect the overall level of activity in various sectors of the economy. They are based on surveys of company managers who assess conditions in their sector, such as order volume, production, new orders, employment, etc. When business activity indices show high values, it indicates that the economy is in a period of expansion; conversely, a decline in these indices may signal an upcoming downturn or recession.

On the other hand, the PMI is a key indicator of economic activity in the manufacturing sector, based on surveys of purchasing managers. The index includes components such as new orders, production level, supplier deliveries, inventories, and employment [57–60].

A PMI value above 50 indicates growth in the manufacturing sector and signals economic development, while values below 50 suggest contraction, potentially foreshadowing a recession. PMI is often considered a leading indicator, as it tends to react faster to changes in the economy than some other measures.

Employment indicators are also widely used, including unemployment rates, changes in employment levels, and average wages. These metrics are critical for assessing economic health. Higher employment levels reflect a more active economy, whereas rising unemployment may indicate an economic slowdown.

Another major indicator used to assess the market is the inflation rate, which measures the pace of price increases for goods and services in the economy. Indicators such as the Consumer Price Index (CPI) and the Producer Price Index (PPI) are used to track inflationary pressures. Moderate inflation often accompanies economic growth, while excessive inflation may lead to an overheated economy and necessitate cooling measures (e.g., interest rate hikes). Conversely, deflation or very low inflation can signal economic weakness and low consumer demand.

When combined, these indicators offer a comprehensive picture of the current economic condition and enhance our understanding of its cyclicality:

- Expansion phase: High PMI values, rising employment, and moderate inflation are typical of periods of economic growth.
- Slowdown phase: If business activity indices and PMI begin to decline, and unemployment slightly rises, it may indicate an upcoming slowdown.
- Recession: A sharp drop in PMI and business activity indices, high unemployment, and low inflation (or deflation) often accompany a recession.
- Recovery: When PMI and business activity indices start to rise again and unemployment begins to decline, it signals the beginning of economic recovery.

2.5. Seasonality and Calendar Effects

Seasonal and calendar effects are specific time-based patterns in financial markets that exhibit cyclically recurring behavior during certain times of the year, month, or week. These effects are based on historical observations and can be used to forecast likely market movements.

Seasonal analysis focuses on trends during particular periods of the year when markets demonstrate predictable behavior [61–64]. One example is the so-called December effect (or Santa Claus rally), during which stock markets tend to rise in the final days of December and the first days of January. Another example is the summer trading slowdown ("Sell in May and go away"), where trading activity often declines during the summer months.

Seasonal effects can be analyzed using seasonal $ARIMA$ ($SARIMA$) models or seasonal decomposition, which views a time series as a combination of three components: trend (T_t), seasonality (S_t), and random noise (R_t):

$$Y_t = T_t + S_t + R_t, \quad (12)$$

where Y_t is the observed value of the time series at time t, and the seasonal component S_t is calculated based on recurring patterns in the data. It reveals how seasonal effects influence market cyclicality.

On the other hand, calendar anomalies refer to specific cyclical behaviors associated with certain days of the week or month [65–68]. An example is the Monday effect, where markets often underperform, or the Friday effect, which may be accompanied by increased volatility (Figure 6). Such effects are often attributed to specific trading practices or psychological factors.

To analyze whether certain days of the week influence returns, one can construct a regression model with dummy variables. For example:

$$Y_t = \alpha + \beta_1 D_{Mon} + \beta_2 D_{Tue} + \beta_3 D_{Wed} + \beta_4 D_{Thu} + \beta_5 D_{Fri} + \varepsilon_t, \quad (13)$$

where:

- Y_t is the return of the asset on day t;
- $D_{Mon}, D_{Tue} \ldots, D_{Fri}$ are dummy variables equal to 1 if the day is the corresponding weekday, and 0 otherwise;
- $\beta_1, \beta_2, \ldots, \beta_5$ are coefficients indicating the effect of each weekday on returns;
- ε_t is the random error.

The constant coefficient α in equation (13) holds particular significance – it represents the average return (or value) of the time series during the observed period when all dummy variables are equal to zero. If dummy variables are used for only four weekdays (e.g., Tuesday through Friday, with Monday as the baseline), then α represents the average return on Monday – i.e., the return when no specific weekday effects are present.

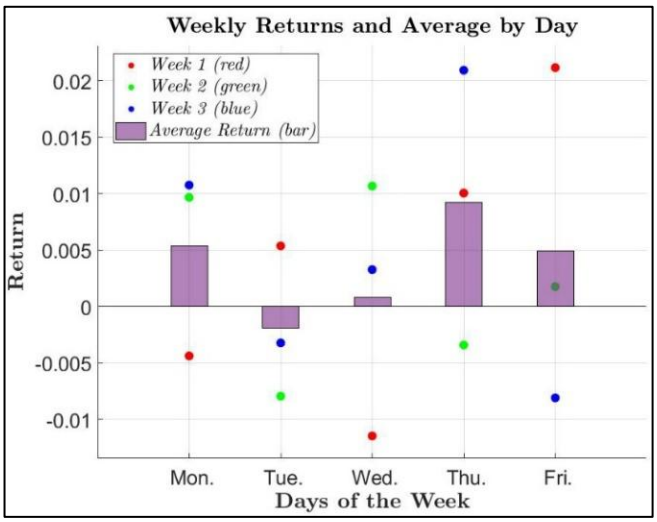


Figure 5. Example of asset returns on different weekdays (Matlab v.2018a).

3. Materials and Methods

While previous studies have explored the use of LSTM networks or spectral analysis individually for time series forecasting [69,70], the novelty of this study lies in the development and application of a hybrid model that integrates both techniques in a structured and complementary manner. The model is trained on synthetic time series with controlled cyclical components and stochastic noise, allowing for a clear evaluation of each model's capacity to detect cycle frequency, period, and amplitude. Additionally, a distinctive contribution of this approach is the use of the Discrete Fourier Transform on the residuals of the LSTM predictions, which serves to uncover latent frequency patterns that remain undetected by the neural network alone. This dual-layer approach – prediction plus residual spectral analysis – offers a new perspective on enhancing the interpretability and robustness of machine learning models in financial cycle detection.

Recurrent Neural Networks (RNNs) are a type of ANN used for tasks involving classification, clustering, and pattern recognition in data. They are particularly well-suited for identifying patterns and cyclical dependencies in complex, nonlinear data such as those found in financial markets. The main idea behind RNNs is that they have an internal state (memory), which allows information from previous steps to be carried forward to subsequent computations. Their architecture includes feedback loops to previous layers: the output from the current state is used as input for the next state (Figure 6).

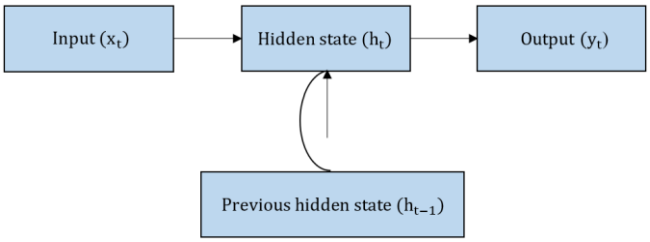


Figure 6. General architecture of a RNN.

In practice, the input x_t at each time step t is processed together with the previous internal state h_{t-1} to produce a new state h_t, and optionally an output y_t:

$$h_t = f(W_h h_{t-1} + W_x x_t + b),$$
$$y_t = g(W_y h_t + c), \quad (14)$$

where f and g are nonlinear functions (e.g., $tanh$, $ReLU$, or $sigmoid$), and W_h, W_x, W_y are weight matrices whose optimal values are learned during training.

With their structure and functionality, RNNs process data sequentially, using information from previous elements in the sequence. The RNN retains context from past time steps, making it well-suited for tasks where previous information is important.

4. Analyzing Cyclicality Using a Hybrid Model of RNNs Combined with Spectral Analysis via Fourier Transforms

To study market cyclicality, we developed a RNN model. Initially, we trained the RNN on a simulated cyclic market pattern to evaluate its ability to accurately determine the frequency and period of each cycle, as well as the amplitudes within individual cycles. The simulation of the cyclical market was defined using sine functions and random noise components:

$$Y = \sin\left(\frac{2\pi t}{20}\right) + \frac{1}{2}\varepsilon(n), \quad (15)$$

where t is the time step, and n is the number of observations The function $\varepsilon(n)$ adds noise to the data by generating a square matrix of dimension $n \times n$ filled with random values following a normal distribution $N(\mu = 0, \sigma = 1)$. This is implemented using the $randn$ function in Matlab v.2018a.

In the simulation defined above, the period is clearly $T = 20$, and the frequency is $\nu = 1/20 = 0.05$. The key question we pose is whether ANNs can capture these two parameters. The amplitude of the simulated market movement is adjusted through the stochastic nature of the term $1/2\,\varepsilon(n)$, which introduces noise into the data – and it was this component that we expected to challenge the models.

The time series data generated by equation (15) are formatted to be usable by the RNN for training and prediction – input-output sequences are created. The input sequence is a series of values fed into the RNN. It may consist of previous input values (e.g., values from $t = 1$ to $t = 10$). The output sequence is the target value Y that the RNN must approximate (predict). For instance, if the input sequence consists of values from $t = 1$ to $t = 10$, then the output sequence is the value of Y at $t = 11$.

The goal of creating input-output sequences is for the ANN to learn to recognize patterns in the input sequence in order to predict the next value in the output sequence. This process of forming input and output sequences is a fundamental step in time series modeling, allowing the ANN to learn how current values (input sequences) affect future values (output sequences).

In a separate supporting array, all values of the time series are stored. These are used as the target values for training. Since the LSTM networks in the software we use expect input data in cell array format, we convert the data from a numerical array into a cell array.

The RNN used in this study is of the LSTM type, specifically designed for working with sequential data such as the time series used in our case. We equipped the RNN with an internal structure that allows it to retain information over long periods of time, effectively avoiding the vanishing gradient problem commonly encountered in standard RNNs.

The LSTM network employs "input," "forget," and "output" gates to control which information should be retained or discarded. This makes the LSTM network highly effective for tasks such as time series forecasting. The LSTM network uses the input sequence to learn the dependencies between points in the time series. It then compares its forecast of the output sequence with the actual values in Y to calculate the error and adjust itself accordingly. After training, the ANN uses a new input sequence to predict the next value in the series, and so on.

In the hybrid model, we integrated the Discrete Fourier Transform to work in conjunction with the RNN:

$$X(f) = \sum_{t=1}^{N-1} x(t) e^{-\frac{i^2 2\pi f t}{N}}, \quad (16)$$

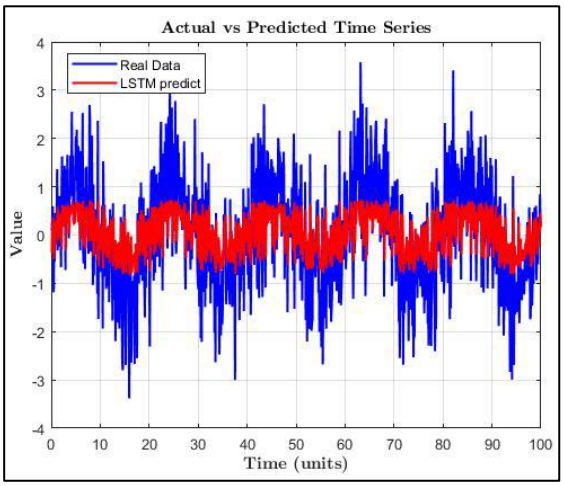
where:

- $X(f)$ is the amplitude of the frequency component f;
- $x(t)$ is the value of the time series at time t;
- N is the number of observations in the time series;
- i is the imaginary unit ($i^2 = -1$).

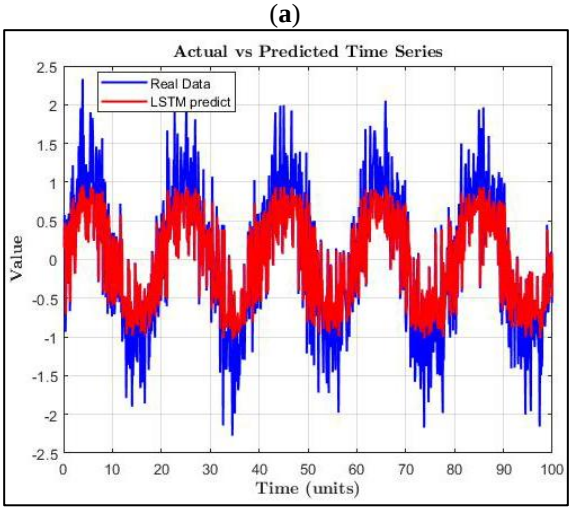
In practice, this mechanism is used to analyze the residuals (i.e., the differences between the actual

values and the RNN's predictions) in order to detect the presence of repeating cycles (frequency components).

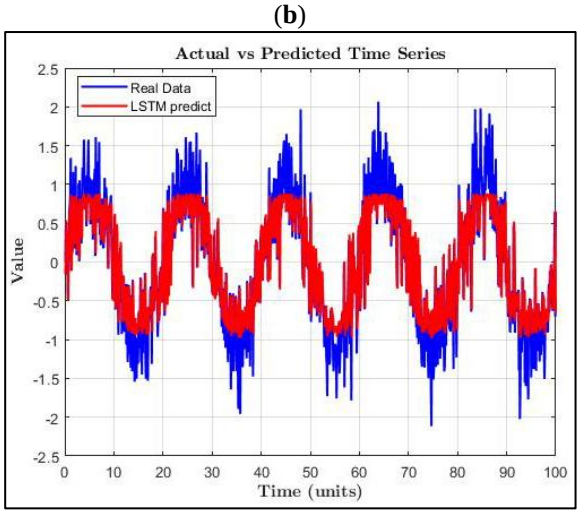
Figure 7 presents the results of the LSTM network with a single input layer (receiving the time series), one LSTM layer with hidden neurons, a connecting layer, and an output layer where the results are obtained. Increasing the number of neurons in the hidden layers improves the LSTM's forecasting capabilities.



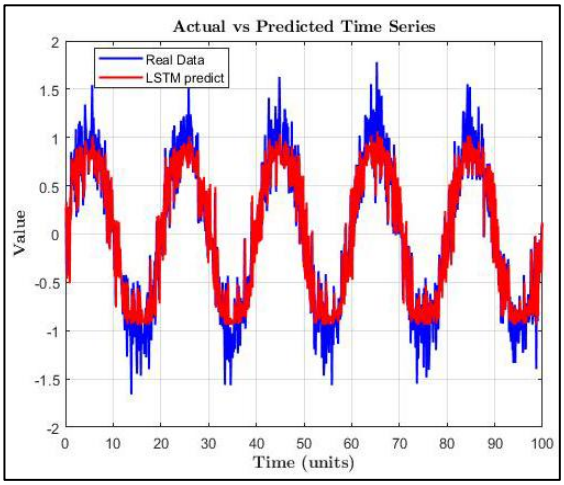
(**a**)



(**b**)



(**c**)



(**d**)

Figure 7. Simulated random time series and results of RNNs with different numbers of hidden neurons in the LSTM layer: (**a**) 25 hidden neurons; (**b**) 50 hidden neurons; (**c**) 120 hidden neurons; (**d**) 253 hidden neurons (Matlab v.2018a).

To enable a fair comparison of how the number of neurons affects approximation performance, all RNNs were trained using the Adam optimizer for 200 epochs. The transfer functions are built into the structure of the LSTM networks. For the input, output, and forget gates, the sigmoid function is used to constrain values between 0 and 1:

$$\sigma(x) = \frac{1}{1+e^{-x}}. \tag{17}$$

For generating candidate values for the cell state, the hyperbolic tangent function is used to constrain values between –1 and 1:

$$\tanh(x) = \frac{e^{x}-e^{-x}}{e^{x}+e^{-x}}. \tag{18}$$

The numbers of hidden neurons (25, 50, 120, and 253) were selected to represent a progressive increase in model complexity, enabling us to assess how the depth of the LSTM layer affects the network's ability to capture amplitude variation in cyclical patterns. The initial values (25 and 50) represent smaller architectures often used in earlier studies for time series forecasting tasks, while 120 and 253 were chosen based on preliminary tests showing that beyond 200 neurons, performance improves marginally but at the cost of computational time. This range also allows for clear visual comparison of model behavior under increasing representational power.

The choice of 200 training epochs was made empirically: during initial trials, convergence of the loss function was observed well before epoch 200 in all configurations, with no signs of overfitting. This value offers a stable benchmark to compare performance across network sizes and optimizers. As for optimization algorithms, Adam was selected as the

primary optimizer due to its proven performance in training recurrent models, while Stochastic Gradient Descent (SGD) was used for comparison, illustrating differences in convergence dynamics and loss values. These choices align with common practices in machine learning and provide a robust basis for evaluating model performance.

What was established at this preliminary stage is that increasing the number of hidden neurons improves the RNN's performance. Almost all ANNs performed very well in identifying the period (and therefore the frequency) of the signal, while capturing the amplitude of market movement posed a challenge for LSTM networks with fewer hidden neurons. Increasing the number of neurons enhances the ANN's ability to handle amplitude variations, resulting in reduced prediction error. Figure 8 presents the loss function of the RNN with 253 hidden neurons in the LSTM layer.

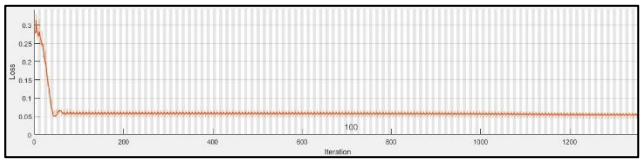


Figure 8. Behavior of the loss function during training (Matlab v.2018a).

The loss function used was the unweighted mean squared error (MSE):

$$Loss = MSE = \frac{1}{n}\sum_{i=1}^{n}(y_t - \hat{y}_t)^2. \quad (19)$$

where:

- n is the number of time periods,
- y_t is the actual market value,
- $\hat{y}_t$ is the value predicted by the RNN.

For the RNN with 253 hidden neurons in the LSTM layer, the final value of the loss was 0.058. Interestingly, the RNN precisely captured the period and frequency of market movements, reporting $T = 20, \nu = 0.05$. The error is primarily due to discrepancies between the actual and predicted amplitudes in the individual market cycles.

During experiments involving different training algorithms, it was found that the optimization algorithm either fails to converge for the problem under consideration, or – if it does converge – it has little effect on the final result, influencing primarily the convergence speed instead (Figure 9).

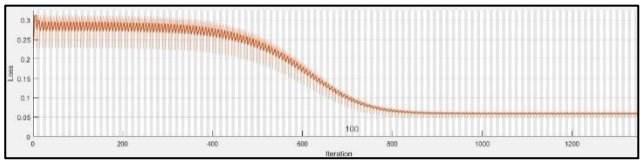


Figure 9. Loss function under stochastic gradient descent (Matlab v.2018a).

The final mean squared error when using the stochastic gradient descent algorithm was 0.061. Once again, the period and frequency were identified with high accuracy, with $T = 20, \nu = 0.05$. The error was primarily due to inaccuracies in predicting the amplitude of market movements within individual cycles.

The conclusions that can be drawn at this preliminary stage – based on simulated market cyclicality – are that RNNs are quite successful in capturing the cycle period, which also implies successful detection of its frequency. However, amplitude appears to be a more challenging aspect for the RNNs to predict accurately. Still, as the ANN architecture becomes more complex, its effectiveness improves, and this issue can potentially be addressed with satisfactory precision.

Additionally, it was found that the Adam optimizer is the most suitable training algorithm for the problem studied. Other converging algorithms mainly affect the speed of convergence to a local minimum of the error, rather than the final prediction result.

5. Conclusion

The study of cyclicality in capital markets is of great importance to financial analysts, investors, and economists. Understanding cyclical patterns enables more accurate forecasting of market movements, which is critical for making timely and informed investment decisions. On the other hand, recognizing cyclicality helps investors identify periods of high volatility or weakness, allowing them to adjust their market exposure accordingly. Upward and downward cycles present favorable opportunities for buying or selling assets, and familiarity with these patterns can even assist in synchronizing investment strategies.

In the conducted study, a LSTM was used in combination with spectral analysis based on the Discrete Fourier Transform. This hybrid architecture was chosen due to the ability of LSTM networks to capture temporal dependencies, and the capability of spectral analysis to identify frequency components in market data. LSTM networks have proven to be effective in processing time series, while spectral analysis enables the detection of cyclical components that might otherwise remain hidden. The experiments conducted demonstrated the applicability of this hybrid model for studying the cyclicality of financial markets.

Acknowledgements

This research is supported by the Bulgarian Ministry of Education and Science under the National Program "Young Scientists and Postdoctoral Students - 2".

References

[1] Muangprathub, J., Intarasit, A., Boongasame, L., et al. (2020). Portfolio risk and return with a new simple moving average of price change ratio. *Wireless Personal Communications, 115*, 3137–3153. https://doi.org/10.1007/s11277-020-07374-3

[2] Prami Swari, M. H., Susila Handika, I. P., & Susila Satwika, I. K. (2021). Comparison of simple moving average, single and modified single exponential smoothing. In *IEEE 7th Information Technology International Seminar (ITIS)* (pp. 1–5). https://doi.org/10.1109/ITIS53497.2021.9791516

[3] Yudono, M. A. S., et al. (2022). Bitcoin USD closing price (BTC-USD) comparison using simple moving average and radial basis function neural network methods. *Jurnal Teknik Elektro, 4*, 29–34. https://doi.org/10.52005/fidelity.v4i2.74

[4] Mazumder, S., et al. (2023). Simple moving average (SMA) crossover strategy with buy sell indicator. *Asia-Pacific Journal of Management and Technology, 3*(4). https://doi.org/10.46977/apjmt.2023.v03i04.004

[5] Karasu, S., & Altan, A. (2022). Crude oil time series prediction model based on LSTM network with chaotic Henry gas solubility optimization. *Energy, 242*, 122964. https://doi.org/10.1016/j.energy.2021.122964

[6] Lucas, A., & Zhang, X. (2016). Score-driven exponentially weighted moving averages and Value-at-Risk forecasting. *International Journal of Forecasting, 32*, 293–302. https://doi.org/10.1016/j.ijforecast.2015.09.003

[7] Saksangsopa, S. (2024, September 13). Comparison of returns using technical strategies relative strength index (RSI) in the Stock Exchange of Thailand. *30th National Graduate Conference, 2*(9). Procedia of Multidisciplinary Research.

[8] Dhokane, R. M., & Agarwal, S. (2024). A predictive model of the stock market using the LSTM algorithm with a combination of exponential moving average (EMA) and relative strength index (RSI) indicators. *Journal of The Institution of Engineers (India): Series B, 105*(B), 1145–1157. https://doi.org/10.1007/s40031-024-01037-8

[9] Ming, L., & Chen, G. (2024). Stock price prediction based on relative strength index, moving average convergence divergence and XGBoost model. In *IEEE 13th Data Driven Control and Learning Systems Conference (DDCLS)* (pp. 1988–1993). Kaifeng, China. https://doi.org/10.1109/DDCLS61622.2024.10606580

[10] Hasan, S., Nurhasanah, S., & Santoso, W. P. (2024). Analisis teknikal menggunakan moving average (MA), moving average convergence-divergence (MACD), dan relative strength index (RSI) untuk mengoptimalkan dalam pengambilan keputusan investasi pada saham sektor manufaktur index LQ45 BEI tahun 2022-2023. *El-Mal: Jurnal Kajian Ekonomi & Bisnis Islam, 5*(4), 3318–3334. https://doi.org/10.47467/elmal.v5i4.2029

[11] Ni, Y., Cheng, Y., Liao, Y., & Huang, P. (2020). Does board structure affect stock price overshooting informativeness measured by stochastic oscillator indicators? *International Journal of Finance & Economics, 27*(2), *2290-2302*. https://doi.org/10.1002/ijfe.2273

[12] Ni, Y., Huang, P., Ku, Y., Liao, Y., & Da, M. Y. (2020). Investing strategies as stochastic oscillator indicators staying in overreaction zones for consecutive days with big data concerns. *Journal of Computers, 31*(1), 1–17. https://doi.org/10.3966/199115992020023101001

[13] Bhatt, A., & Gor, R. (2022). A review paper on quantum harmonic oscillator and its applications in finance. *IOSR Journal of Economics and Finance (IOSRJEF), 13*(3), 61–65.

[14] Paik, C. K., Choi, J., & Vaquero, I. U. (2024). Identifying oversold levels and developing low-frequency trading algorithms for the S&P 500: An analysis using stochastic oscillator, Williams %R, and trading volume. *Preprints, 2024090383*. https://doi.org/10.20944/preprints202409.0383.v1

[15] Barra, S., Carta, S. M., Corriga, A., Podda, A. S., & Recupero, D. R. (2020). Deep learning and time series-to-image encoding for financial forecasting. *IEEE/CAA Journal of Automatica Sinica, 7*(3), 683–692. https://doi.org/10.1109/JAS.2020.1003132

[16] Livieris, I. E., Pintelas, E., & Pintelas, P. A. (2020). NN–LSTM model for gold price time-series forecasting. *Neural Computing and Applications, 32*, 17351–17360. https://doi.org/10.1007/s00521-020-04867-x

[17] Cheng, D., Yang, F., et al. (2022). Financial time series forecasting with multi-modality graph neural network. *Pattern Recognition, 121*, 108218. https://doi.org/10.1016/j.patcog.2021.108218

[18] Cao, L. (2022). AI in finance: Challenges, techniques, and opportunities. *ACM Computing Surveys, 5*(3), 1–38. https://doi.org/10.1145/3502289

[19] Karaca, Y., Zhang, Y.-D., & Muhammad, K. (2020). Characterizing complexity and self-similarity based on fractal and entropy analyses for stock market forecast modelling. *Expert Systems with Applications, 144*, 113098. https://doi.org/10.1016/j.eswa.2019.113098

[20] Frezza, M., Bianchi, S., & Pianese, A. (2021). Fractal analysis of market (in)efficiency during the COVID-19. *Finance Research Letters, 38*, 101851. https://doi.org/10.1016/j.frl.2020.101851

[21] Moradi, M., Jabbari Nooghabi, M., & Rounaghi, M. M. (2021). Investigation of fractal market hypothesis and forecasting time series stock returns for Tehran Stock Exchange and London Stock Exchange. *International Journal of Finance & Economics, 26*, 662–678. https://doi.org/10.1002/ijfe.1809

[22] Blackledge, J., & Lamphiere, M. (2022). A review of the fractal market hypothesis for trading and market price prediction. *Mathematics, 10*(1), 117. https://doi.org/10.3390/math10010117

[23] David, S. A., et al. (2020). Measuring the Brazilian ethanol and gasoline market efficiency using DFA-Hurst and fractal dimension. *Energy Economics, 85*, 104614. https://doi.org/10.1016/j.eneco.2019.104614

[24] Mnif, E., Jarboui, A., & Mouakhar, K. (2020, October). How the cryptocurrency market has performed during COVID-19? A multifractal analysis. *Finance Research Letters, 36*, 101647. https://doi.org/10.1016/j.frl.2020.101647

[25] Borowski, K., & Matusewicz, M. (2020). Calculating Hurst exponent with the use of the Siroky method in developed and emerging markets. *Finanse i Prawo Finansowe, 3*(27), 25–61. https://doi.org/10.18778/2391-6478.3.27.02

[26] Gómez-Águila, A., Trinidad-Segovia, J. E., & Sánchez-Granero, M. A. (2022). Improvement in Hurst exponent estimation and its application to financial markets. *Financial Innovation, 8*, 86. https://doi.org/10.1186/s40854-022-00394-x

[27] Li, Y., & Teng, Y. (2022). Estimation of the Hurst parameter in spot volatility. *Mathematics, 10*, 1619. https://doi.org/10.3390/math10101619

[28] Zournatzidou, G., & Floros, C. (2023). Hurst exponent analysis: Evidence from volatility indices and the volatility of volatility indices. *Journal of Risk and Financial Management, 16*(5), 272. https://doi.org/10.3390/jrfm16050272

[29] Dogra, P., Koay, E. J., Wang, Z., et al. (2021). Is the worst of the COVID-19 global pandemic yet to come? Application of financial mathematics as candidate predictive tools. *Translational Psychiatry, 11*, Article 299. https://doi.org/10.1038/s41398-021-01429-0

[30] Xu, C., Ke, J., Peng, Z., Fang, W., & Duan, Y. (2022). Asymmetric fractal characteristics and market efficiency analysis of style stock indices. *Entropy, 24*(7), 969. https://doi.org/10.3390/e24070969

[31] Fernandes, L. H. S., et al. (2022). Multifractal cross-correlations risk among WTI and financial assets. *Fractals, 30*(9). https://doi.org/10.1142/S0218348X22501912

[32] Wang, G., et al. (2023). Fractal wave velocity evolution model of ultrasonic propagation and attenuation in water-bearing coal. *International Journal of Rock Mechanics and Mining Sciences, 161*, 105283. https://doi.org/10.1016/j.ijrmms.2022.105283

[33] Jung, N., Le, Q. A., Mafwele, B. J., et al. (2020). Fractality and multifractality in a stock market's nonstationary financial time series. *Journal of the Korean Physical Society, 77*, 186–196. https://doi.org/10.3938/jkps.77.186

[34] Aslam, F., et al. (2021). Investigating efficiency of frontier stock markets using multifractal detrended fluctuation analysis. *International Journal of Emerging Markets, 18*(7), 1650–1676. https://doi.org/10.1108/IJOEM-11-2020-1348

[35] Zhuang, X., & Wei, D. (2022). Asymmetric multifractality, comparative efficiency analysis of green finance markets: A dynamic study by index-based model. *Physica A: Statistical Mechanics and Its Applications, 604*, 127949. https://doi.org/10.1016/j.physa.2022.127949

[36] Abril-Bermúdez, F. S. (2024). Multifractality approach of a generalized Shannon index in financial time series. *PLOS ONE, 9*(6), e0303252. https://doi.org/10.1371/journal.pone.0303252

[37] Lusindah, L., & Sumirat, E. (2021). Implementation of Fibonacci retracements and exponential moving average (EMA) trading strategy in Indonesia Stock Exchange. *European Journal of Business and Management Research, 6*(4), 402–408. https://doi.org/10.24018/ejbmr.2021.6.4.1033

[38] Kurujitkosol, T., Takhom, A., & Usanavasin, S. (2022). Forex price movement prediction using stacking machine learning models. In *Proceedings of the 7th International Joint Symposium on Artificial Intelligence and Natural Language Processing (iSAI-NLP)* (pp. 1–6). Chiang Mai, Thailand. https://doi.org/10.1109/iSAI-NLP56921.2022.9960245

[39] Oburu, J., Were, J., Oduor, B., & Nyakinda, J. (2023). On (P,Q)-binomial extension of Cox-Ross-Rubinstein model in Skorohod spaces. *International Journal of Open Problems in Computer Mathematics, 16*(4), 59–77. https://www.ijopcm.org/Vol/2023/2023.4.5.pdf

[40] Azzam, N. A., & Batulan, R. A. (2023). Fibonacci trading strategy. In *Research on Islamic Business Concepts* (pp. 347–356). Singapore: Springer. https://doi.org/10.1007/978-981-99-5118-5_21

[41] Meher, B. K., et al. (2021). Forecasting stock market prices using mixed ARIMA model: A case study of Indian pharmaceutical companies. *Investment Management and Financial Innovations, 18*(1), 42–54. https://doi.org/10.21511/imfi.18(1).2021.04

[42] Dong, X., Dang, B., Zang, H., Li, S., & Ma, D. (2024). The prediction trend of enterprise financial risk based on machine learning ARIMA model. *Journal of Theory and Practice of Engineering Science, 4*(1), 65–71. https://doi.org/10.53469/jtpes.2024.04(01).09

[43] Nokeri, T. C. (2021). Forecasting using ARIMA, SARIMA, and the additive model. In *Implementing machine learning for finance* (pp. 21–50). https://doi.org/10.1007/978-1-4842-7110-0_2

[44] Dubey, A. K., et al. (2021). Study and analysis of SARIMA and LSTM in forecasting time series data. *Sustainable Energy Technologies and Assessments, 47*, 101474. https://doi.org/10.1016/j.seta.2021.101474

[45] Sirisha, U. M., Belavagi, M. C., & Attigeri, G. (2022). Profit prediction using ARIMA, SARIMA and LSTM models in time series forecasting: A comparison. *IEEE Access, 10*, 124715–124727. https://doi.org/10.1109/ACCESS.2022.3224938

[46] Luo, J. (2023). A feature-engineered ARIMA-SARIMA hybrid model for stock price prediction. In *Proceedings of the 1st International Conference on Data Analysis and Machine Learning (DAML 2023)* (pp. 47–53). https://doi.org/10.5220/0012814600003885

[47] Ilyas, Q. M., Iqbal, K., Ijaz, S., Mehmood, A., & Bhatia, S. (2022). A hybrid model to predict stock closing price using novel features and a fully modified Hodrick–Prescott filter. *Electronics*, *11*(21), 3588. https://doi.org/10.3390/electronics11213588

[48] Qin, Q., et al. (2022). Hodrick–Prescott filter-based hybrid ARIMA–SLFNs model with residual decomposition scheme for carbon price forecasting. *Applied Soft Computing*, *119*, 108560. https://doi.org/10.1016/j.asoc.2022.108560

[49] Phillips, P. C. B., & Jin, S. (2020). Business cycles, trend elimination, and the HP filter. *International Economic Review*, *62*(2), 469–520. https://doi.org/10.1111/iere.12494

[50] Qian, W., & Wang, J. (2020). An improved seasonal GM(1,1) model based on the HP filter for forecasting wind power generation in China. *Energy*, *209*, 118499. https://doi.org/10.1016/j.energy.2020.118499

[51] Prabheesh, K. P., et al. (2021). The dynamics of global financial cycle and domestic economic cycles: Evidence from India and Indonesia. *Economic Modelling*, *94*, 831–842. https://doi.org/10.1016/j.econmod.2020.02.024

[52] Malik, S., Abbas, A., Shabbir, M. S., et al. (2024). Business cycle fluctuations, foreign direct investment, and real effective exchange rate nexus among Asian countries. *Journal of the Knowledge Economy*, *15*, 5466–5479. https://doi.org/10.1007/s13132-023-01316-7

[53] Farrell, G., & Kemp, E. (2020). Measuring the financial cycle in South Africa. *South African Journal of Economics*, *88*(2), 123–144. https://doi.org/10.1111/saje.12246

[54] Schüler, Y. S., et al. (2020). Financial cycles: Characterisation and real-time measurement. *Journal of International Money and Finance*, *100*, 102082. https://doi.org/10.1016/j.jimonfin.2019.102082

[55] Chen, S., & Svirydzenka, K. (2021). Financial cycles – Early warning indicators of banking crises? *IMF Working Papers*. International Monetary Fund. https://doi.org/10.5089/9781513582306.001

[56] Trusova, N., et al. (2022). Attracting foreign investment in cyclic imbalances of the economy. *Scientific Horizons*, *25*(5), 101–116. https://doi.org/10.48077/scihor.25(5).2022.101-116

[57] Sobko, R., & Klonowska-Matynia, M. (2021). The relationship between the Purchasing Managers' Index (PMI) and economic growth: The case for Poland. *European Research Studies Journal*, *24*(s1), 198–219. https://www.um.edu.mt/library/oar/handle/123456789/98643

[58] Shaikh, I. (2021). On the relation between Purchasing Manager's Index and trade policy uncertainty: Evidence from China, Japan and the USA. *Journal of Chinese Economic and Foreign Trade Studies*, *14*(2), 202–223. https://doi.org/10.1108/JCEFTS-10-2020-0068

[59] Cifuentes-Faura, J. (2022). The Purchasing Managers' Index as a predictor of the economic situation during COVID-19 and its relevance for decision making. *International Journal of Public Policy*, *16*(1), 1–12. https://doi.org/10.1504/IJPP.2021.121893

[60] Dayan, V., & Erdogan, S. (2023). The relationship between the Purchasing Managers' Index (PMI) with economic and financial indicators in Turkey: A VAR analysis. *E-Journal of the Faculty of Economics and Administrative Sciences*, *12*(2), 183–202. https://doi.org/10.47934/tife.12.02.03

[61] Miloș, L. R., Haţiegan, C., Miloș, M. C., Barna, F. M., & Boțoc, C. (2020). Multifractal detrended fluctuation analysis (MF-DFA) of stock market indexes: Empirical evidence from seven Central and Eastern European markets. *Sustainability*, *12*(2), 535. https://doi.org/10.3390/su12020535

[62] Chang, C.-L., McAleer, M., & Wong, W.-K. (2020). Risk and financial management of COVID-19 in business, economics and finance. *Journal of Risk and Financial Management*, *13*(5), 102. https://doi.org/10.3390/jrfm13050102

[63] Ghaderpour, E., Pagiatakis, S. D., & Hassan, Q. K. (2021). A survey on change detection and time series analysis with applications. *Applied Sciences*, *11*(13), 6141. https://doi.org/10.3390/app11136141

[64] Rane, N. L., Choudhary, S. P., & Rane, J. (2024). Artificial intelligence-driven corporate finance: Enhancing efficiency and decision-making through machine learning, natural language processing, and robotic process automation in corporate governance and sustainability. *Economics and Business Relations*, *5*(2), 1–22. https://doi.org/10.48185/sebr.v5i2.1050

[65] Gayaker, S., et al. (2020). The day of the week effect and interest rates. *Borsa Istanbul Review*, *20*(1), 55–63. https://doi.org/10.1016/j.bir.2019.07.010

[66] Chiah, M., & Zhong, A. (2021). Tuesday blues and the day-of-the-week effect in stock returns. *Journal of Banking & Finance*, *113*, 106243. https://doi.org/10.1016/j.jbankfin.2021.106243

[67] Kinateder, H., & Papavassiliou, V. G. (2021). Calendar effects in Bitcoin returns and volatility. *Finance Research Letters*, *38*, 101420. https://doi.org/10.1016/j.frl.2019.101420

[68] Qadan, M., et al. (2022). Seasonal and calendar effects and the price efficiency of cryptocurrencies. *Finance Research Letters*, *46*(A), 102354. https://doi.org/10.1016/j.frl.2021.102354

[69] Mroua, M., & Lamine, A. (2023). Financial time series prediction under COVID-19 pandemic crisis with Long Short-Term Memory (LSTM) network. *Humanities and Social Sciences Communications*, *10*, Article 530. https://doi.org/10.1057/s41599-023-02042-w

[70] Škare, M., & Porada-Rochoń, M. (2020). Forecasting financial cycles: Can big data help? *Technological and Economic Development of Economy*, *26*(5), 974–988. https://doi.org/10.3846/tede.2020.12702

Computer Science and Interdisciplinary Research Journal
Volume 2, Issue 1, 2025. ISSN: 3033-1218, DOI: 10.70862/CSIR.2025.0201-12

Analyzing the Efficacy of Base Algorithms: Application of Machine Learning Models in Classification Tasks

Yass Khudheir Salal

General Directorate for Education in Al-Qadisiyah, Ministry of Education, Iraq

yasskhudheirsalal@gmail.com

Abstract – The article presents a comprehensive examination and comparison of several machine learning algorithms, such as Artificial Neural Network, K-Nearest Neighbor, Support Vector Machines, Decision Tree, and Random Forests, with a specific focus on the significance of neural networks in classification tasks. This study employs three distinct datasets: (1) the "Titanic" dataset is about the sinking of the British transatlantic liner, which resulted in the deaths of 1502 of the 2224 people who were on board. On April 15, 1912, the event occurred in the North Atlantic Ocean; (2) the "MNIST" dataset, which consists of 70,000 grayscale images (28×28 pixels) of handwritten digits; (3) and the "Fashion-MNIST" dataset, which showcases images of 10 fashion item classes such as T-shirt/Top, Trouser, Pullover, Dress, Coat, Sandals, Shirt, Sneaker, Bag, Ankle boots which have been classified. The analysis evaluates the strengths and weaknesses of each algorithm based on the research outcomes. The results suggest that for certain classification tasks, the use of machine learning algorithms may be more appropriate, as the accuracy levels may remain relatively unchanged, and vary depending on the size of the dataset used with the features it includes in the classification tasks.

Keywords **–** machine learning, deep learning, artificial neural networks, classification, Titanic.

DOI: https://doi.org/10.70862/CSIR.2025.0201-12

Corresponding author: Yass Salal,
General Directorate for Education in Al-Qadisiyah, Ministry of Education, Iraq
Email: yasskhudheirsalal@gmail.com

Received: 14 May 2025.
Revised: 3 June 2025.
Accepted: 11 June 2025.
Published: 22 June 2025.

© 2025 Yass Salal. This work is licensed under the Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License.

The article is published with Open Access at http://csirjournal.com/

1. Introduction

1.1. Objective of the Study

Machine learning and deep learning represent fields within artificial intelligence that apply distinct methods and algorithms to resolve problems associated with classification, regression, image recognition, and the automation of various processes [1-3].

The problem of choosing the best algorithm surfaced during the examination of the Titanic dataset [4], which entails estimating a person's chances of surviving based on particular traits. The objective is to evaluate which algorithms are most appropriate for handling categorization tasks. The objectives of this study are to train neural networks and machine learning algorithms on a variety of classification problems, compare and analyze performance metrics such as Accuracy (ACC), training time, prediction delay, and model memory, and assess whether applying methods to particular issues is feasible.

In this study, the selected algorithms and techniques include the Support Vector Machine (SVM) [5], a machine learning strategy that establishes a separating hyperplane in the parameter space. The K-nearest neighbor (KNN) [6] method classifies an object by examining the classes of its K-nearest neighbor. A Decision Tree (DT) algorithm is utilized to create decision paths based on the evaluation of various features [7]. Random Forest (RF) [8] is another machine learning approach that merges the results from multiple decision trees. Furthermore, the backpropagation method is applied to train the standard neural network model referred to as an artificial neural network (ANN).

In the present investigation, the primary emphasis is placed on the importance of neural networks in

classification endeavors in relation to alternative algorithms, utilizing a range of datasets. This implies that for particular classification tasks, using accuracy as a metric for each algorithm is used to identify the most effective classification approach, as shown below.

1.2. Literature Review

In recent decades, a variety of classification methods have been introduced [9]. Among the most frequently used techniques are KNN [10], ANNs [11-13], SVM [14,15], and ensembles of classification trees, such as Random Forest [16,17]. Algorithms that utilize Decision Trees are relatively easy to implement, as they require the estimation of fewer parameters; thus, they demonstrate a significant level of automation. However, this practical benefit of DT over ANN can be diminished by a tendency to overfit the data [18]. For these reasons, DT models and, to a lesser extent, ANN have, in some applications, been supplemented or replaced by more advanced and easier-to-train machine learning algorithms (MLAs). Over the last decade, kernel methods like SVM [19,20] and tree ensembles such as RF [21] have emerged as highly effective methodologies for classification tasks. Numerous studies suggest that MLAs often outperform statistical techniques like discriminant analysis or logistic regression, especially when the feature space is complex or when the input datasets are expected to display varying statistical distributions [22]. As computational power has improved, MLAs have garnered more attention, and the performance of pattern recognition systems has correspondingly enhanced [23]. In many studies, RF, SVM, and gradient boosting methods are considered among the top-performing classifiers, while KNN remains a simple and interpretable baseline [24]. The essential steps in deciding which algorithm to use will depend on several factors, including the number of examples in the training set, the dimensions of the feature space, the existence of correlated features, and the risk of overfitting [25]. Once these considerations have been addressed, the suitable algorithm can then be chosen. By applying methods from statistical physics, the generalization performance of SVM, which has historically been considered a robust alternative to ANN for certain classification tasks, was investigated [26].

1.3. Data Preprocessing

Data preprocessing occurs before the training of models. For each classification task, both a training set and a test set are created. The dataset containing various attributes of a selected group of Titanic passengers is referred to as the "Titanic" dataset. The dataset consisting of a collection of handwritten digit samples is known as the "MNIST" dataset [27]. Additionally, the dataset comprising images of clothing items is called the "Fashion-MNIST" dataset [28]. A fundamental principle underpins the processing of all algorithms: to achieve optimal results, hyperparameters must be determined through trial and error following the training of each machine learning algorithm and neural network on the training set. The subsequent step involves evaluating the models' accuracy using the test set after the training phase.

1.4. Evaluating Algorithm Performance

For the algorithms being studied, the prediction accuracy was selected as a crucial criterion. The Kaggle platform was used to gauge the algorithms' accuracy on the Titanic dataset classification task. After submission, the predicted outcomes were evaluated against the ground-truth labels stored on the platform. Test accuracy was also used to evaluate performance on the MNIST and Fashion-MNIST datasets. The accuracy metric is calculated using the following formula:

$$\text{Accuracy} = \frac{1}{n_{samples}} \sum_{i=0}^{n_{samples}-1} \mathbf{1}(\hat{y}_i = y_i). \qquad (1)$$

In multi-label classification, this function returns the subset accuracy [29]. If the entire set of predicted labels for a sample exactly matches the true set of labels, then the subset accuracy is 1.0; otherwise, the value is less than one. If the predicted value is the $\hat{y}_i$ of the i sample and y_i is the corresponding true value, then the proportion of correct predictions out of $n_{samples}$ is determined by the above formula.

Accuracy measures the proportion of correctly predicted instances (both positive and negative in binary classification) over the total number of instances.

In addition to “accuracy”, the following parameters are used for comparison: the training time, the prediction time of the result, and the storage space occupied by the model on the storage device. The tasks for each dataset and a comparison of the effectiveness of the algorithms will be described below.

2. Data Exploration

The study examines three specific tasks: "Titanic", "MNIST", and "FashionMNIST", which will be explained as follows:

2.1. Titanic Dataset

For the Titanic task, the objective is to predict the survival outcome of each passenger based on selected features. In processing the data, four representative features selected for the model are listed in Table 1.

Table 1. Key features of the Titanic dataset.

Variables	Definition	Key
Passenger Class	Ticket class	1 = 1st, 2 = 2nd, 3 = 3rd
Gender	Female, Male	
Sibsp	Siblings / spouses aboard the Titanic	Sibling = brother, sister, stepbrother, stepsister Spouse = husband, wife (excluding non-marital partners and unofficial relationships)
Parch	The number of parents and children (including foster children) aboard.	Parent = mother, father Child = daughter, son, stepdaughter, stepson Some children travelled only with a nanny, therefore parch=0 for them.

The target variable for this analysis is survival. The training dataset comprises 891 instances, while the testing dataset includes 418 instances. In the neural network model, the Adam optimizer is employed to enhance the learning process [30], alongside the "binary_crossentropy" loss function and the accuracy metric for performance evaluation. The architecture consists of four neurons in the input layer, 18 neurons in the hidden layer utilizing the Relu activation function, and a single neuron in the output layer with a Sigmoid activation function. The model is trained over 40 epochs.

2.2. The MNIST Dataset

The MNIST dataset contains labeled images of handwritten digits in the training data and unlabeled images of handwritten digits in the testing data. The testing data is only used when submitting to the competition and these images are unlabeled because if the testing data labels were provided, then it would be possible to cheat and to get a perfect score in the competition.

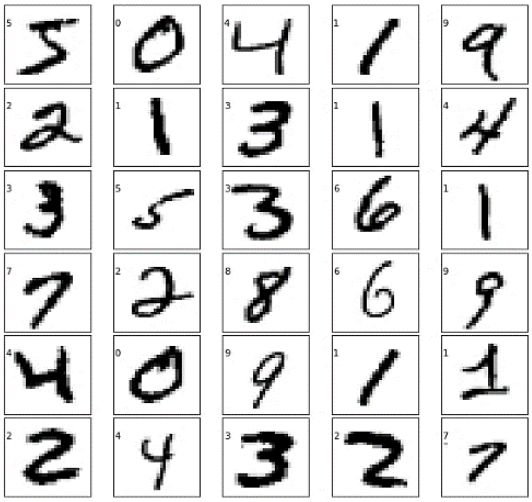


Figure 1. MNIST dataset examples.

In order to evaluate classifiers, scikit-learn's cross_val_score or train_test_split functions are used to set a portion of labeled training data for model evaluation.

The MNIST dataset comprises a total of 70,000 images, each measuring 28×28 pixels, representing handwritten digits from 0 to 9. This dataset is divided into a training set containing 60,000 images and a test set with 10,000 images. The primary features for analysis are the individual pixels of these images as shown in Figure 1.

2.3. Fashion-MNIST Dataset

The objective is to identify the type of garment depicted in a 28×28-pixel image. I will be using the Fashion-MNIST dataset, which is a collection of 70,000 images (28×28 pixels each). The collection contains ten types of clothing to be classified. The training sample contains 60,000 images, and the test sample contains 10,000 images. The input features are the pixel values of the images (see Figure 2).



Figure 2. Fashion-MNIST dataset examples.

3. Result and Discussion

3.1. Titanic Classification

The default hyperparameters were utilized for the Decision Tree model. The tasks analyzed in this study include "Titanic", "MNIST", and "FashionMNIST".

In the Random Forest model, the number of trees utilized (n_estimators) is set to 50, the k-neighbors parameter is configured to k=4. In the SVM model, all hyperparameters were applied with their default settings. The predicted outcomes were submitted to the Kaggle platform for the Titanic machine learning competition to assess prediction accuracy on the test dataset. The results were documented by evaluating the developed model on the test dataset, employing the "accuracy" metric as shown in Table 2.

Table 2. Results of classification accuracy on the Titanic dataset.

Metric / Classifier	ACC	Training time/sec.	Time to predict test sample /sec.	Storage space/KB
ANN	0.772	15.8800	0.2640	100.00
KNN	0.777	0.0029	0.0400	72.80
SVM	0.772	0.0030	0.0100	22.30
DT	0.764	0.0020	0.0003	9.55
RF	0.774	0.1920	0.0270	367.00

3.2. MNIST Classification

The following configurations were used to adapt the neural network: the "Adam" optimizer was utilized alongside the "categorical_crossentropy" loss function. The network architecture included 784 input neurons, a first hidden layer with 128 neurons activated by the "relu" function, a second hidden layer also consisting of 128 neurons with the "relu" activation, and an output layer featuring ten neurons with the "softmax" activation function. The training process spanned 30 epochs.

For the KNN algorithm, the number of neighbors (k) was set to five. In the case of the SVM and Decision Tree models, all hyperparameters were maintained at their default settings. The Random Forest model was constructed using 130 trees.

The performance of the models was evaluated by testing them on the dataset's test sample, with results measured using the "accuracy" metric, as shown in Table 3.

Table 3. Results of classification accuracy on the MNIST dataset.

Metric / Classifier	ACC	Training time/sec.	Time to predict test sample /sec.	Storage space/KB
ANN	0.897	204.015	0.71700	1.5300
KNN	0.793	0.08970	47.4330	373.00
SVM	0.769	302.024	106.432	84.600
DT	0.867	21.6012	0.18200	1.0030
RF	0.878	59.7898	0.61200	158.00

The best prediction accuracy was achieved using a neural network; however, this approach required more time for training compared to other algorithms. KNN required minimal training time since it stores the training data without explicit learning, but it requires substantial storage capacity. It was found that using an ANN for this specific task is beneficial, as long as the time needed for training the model isn't a major concern.

Random Forest is a viable alternative for reducing training time. It was anticipated that neural networks would outperform traditional machine learning algorithms in the task of digit recognition, given that each pixel in the image serves as a parameter. Neural networks are generally recommended for scenarios involving a high number of parameters, where conventional machine learning methods might fail to identify essential correlations. However, in practice, the performance of the algorithms was found to be roughly equivalent. This may be attributed to the simplicity of the digit images, which allows machine learning algorithms to effectively classify the data [31].

3.3. Fashion-MNIST Classification

The value of *k* for the KNN classifier was set to 5, representing the number of neighbors. All hyperparameters for both SVM and Decision Trees were used with their default values. The Random Forest model consisted of 130 trees. The results were reported by computing the accuracy metric from the developed model over the utilized dataset's test set, as shown in Table 4.

Table 4. Results of classification accuracy on the Fashion-MNIST dataset.

Metric / Classifier	ACC	Training time/sec.	Time to predict test sample /sec.	Storage space/KB
ANN	0.830	140.00	0.640	1.36
KNN	0.8388	0.17	42.560	363.00
SVM	0.8828	392.98	170.680	125.00
DT	0.7919	77.28	0.029	2.00
RF	0.8751	195.12	1.718	266.00

The best prediction accuracy was achieved by the SVM algorithm, although the neural network followed closely. The Decision Tree model achieved relatively low accuracy. However, it trained quickly and required minimal memory. Compared with other machine learning algorithms, the accuracy of the SVM algorithm was high, although it required significantly more time for both training and prediction.

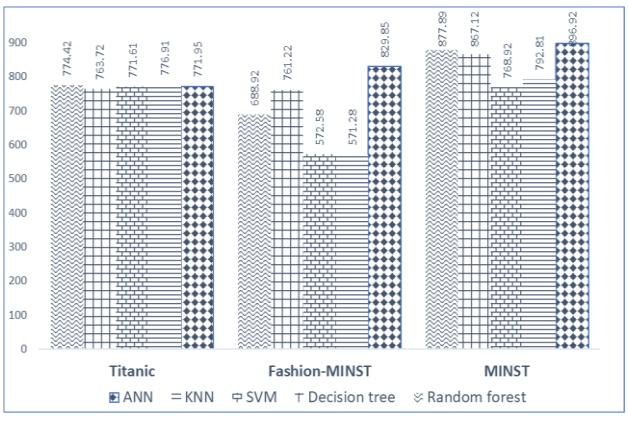


Figure 3. General diagram.

It was anticipated that the neural network would outperform traditional machine learning algorithms in clothing recognition. Although it was anticipated that the neural network would achieve the highest performance, the SVM algorithm slightly

outperformed it in terms of accuracy. Subsequently, we will summarize the study's results (see Figure 3).

Each approach is assigned a designated number of points according to a formula that involves multiplying each measurement by specific coefficients after all measurement data has been collected. The coefficient for accuracy is set at 1000, while the coefficients for training time, prediction time, and occupied space are -0.4, -0.6, and -0.4, respectively.

A substantial positive coefficient is selected for accuracy, as it is the most critical parameter and is evaluated in relative terms. In contrast, the coefficients for time and space utilization are assigned negative values, reflecting their inherently negative dimensions. A diagram is created for the Z-final scoring, illustrating that although certain models may exhibit high accuracy, they can receive low scores in other criteria, which can considerably influence the overall evaluation.

Consequently, it cannot be definitively asserted that neural networks possess a substantial advantage over machine learning algorithms. The findings indicate that for certain classification tasks, employing machine learning algorithms may be more appropriate, as the accuracy levels may remain relatively unchanged.

4. Conclusion

In this paper, we argue that the most effective method for assessing the strengths and weaknesses of various algorithms, including ANN, kNN, SVM, Decision Trees, and Random Forests, is to conduct a comprehensive evaluation across multiple datasets. Furthermore, the article examines the applicability of these algorithms to specific issues, highlighting that the ANN classifier demonstrated the highest accuracy in its results.

The researchers will persist in their efforts to evaluate the effectiveness of machine learning algorithms in comparison to neural networks across various tasks. They are particularly focused on areas such as traffic management, resource distribution, and the prediction of signal levels in 5G and 6G mobile networks.

Acknowledgments

I would like to express my sincere gratitude to my former supervisor Sanjar Abdoulaev for his invaluable support.

References

[1] Soori, M., Arezoo, B., & Dastres, R. (2023). Artificial intelligence, machine learning and deep learning in advanced robotics: A review. *Cognitive Robotics, 3*, 54–70. https://doi.org/10.1016/j.cogr.2023.04.001

[2] Salal, Y. K., Hussain, M., & Theodorou, P. (2021). Student next assignment submission prediction using a machine learning approach. In A. A. Radionov & V. R. Gasiyarov (Eds.), *Advances in Automation II. RusAutoCon 2020* (Lecture Notes in Electrical Engineering, Vol. 729). Springer. https://doi.org/10.1007/978-3-030-71119-1_38

[3] Hussain, S., Muhsion, Z. F., Salal, Y. K., Theodorou, P., Kurtoğlu, F., & Hazarika, G. C. (2019). Prediction model on student performance based on internal assessment using deep learning. *International Journal of Emerging Technologies in Learning, 14*(8), 4. https://doi.org/10.3991/ijet.v14i08.10001

[4] Kaggle. (n.d.). *Titanic – Machine learning from disaster*. Retrieved February 3, 2025, from https://www.kaggle.com/c/titanic/data

[5] Scikit-learn. (n.d.). *SVC*. Retrieved February 6, 2025, from https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html

[6] Scikit-learn. (n.d.). *KNeighborsClassifier*. Retrieved from https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsClassifier.html

[7] Scikit-learn. (n.d.). *DecisionTreeClassifier*. Retrieved February 5, 2025, from https://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeClassifier.html

[8] Scikit-learn. (n.d.). *RandomForestClassifier*. Retrieved February 3, 2025, from https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html

[9] Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The elements of statistical learning: Data mining, inference, and prediction* (2nd ed.). Publ. Springer New York. https://doi.org/10.1007/978-0-387-84858-7

[10] Breiman, L., & Ihaka, R. (1984). *Nonlinear discriminant analysis via scaling and ACE* (Technical Report No. 40). Department of Statistics, University of California, Berkeley. https://digitalassets.lib.berkeley.edu/sdtr/ucb/text/40.pdf

[11] McCloskey, M., & Cohen, N. J. (1989). Catastrophic interference in connectionist networks: The sequential learning problem. *Psychology of Learning and Motivation, 24*, 109–165. https://doi.org/10.1016/S0079-7421(08)60536-8

[12] Brown, W. M., Gedeon, T. D., Groves, D. I., & Barnes, R. G. (2000). Artificial neural networks: A new method for mineral prospectivity mapping. *Australian Journal of Earth Sciences, 47*, 757–770. https://doi.org/10.1046/j.1440-0952.2000.00807.x

[13] Rigol-Sanchez, J. P., Chica-Olmo, M., & Abarca-Hernandez, F. (2003). Artificial neural networks as a tool for mineral potential mapping with GIS. *International Journal of Remote Sensing, 24*, 1151–1156. https://doi.org/10.1080/0143116021000031791

[14] Boser, B. E., Guyon, I. M., & Vapnik, V. N. (1992). A training algorithm for optimal margin classifiers. In

Proceedings of the fifth annual workshop on Computational Learning Theory (pp. 144–152). https://doi.org/10.1145/130385.130401

[15] Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning*, *20*, 273–297. https://doi.org/10.1007/BF00994018

[16] Breiman, L. (2001). Random forests. *Machine Learning*, *45*, 5–32. https://doi.org/10.1023/A:1010933404324

[17] Rodriguez-Galiano, V. F., & Chica-Rivas, M. (2014). Evaluation of different machine learning methods for land cover mapping of a Mediterranean area using multi-seasonal Landsat images and digital terrain models. *International Journal of Digital Earth*, *7*, 492–509. https://doi.org/10.1080/17538947.2012.748848

[18] Breiman, L., Friedman, J., Olshen, R., & Stone, C. (1984). *Classification and regression trees* (Vol. 37). Wadsworth. https://doi.org/10.1201/9781315139470

[19] Al-Anazi, A., & Gates, I. D. (2010). A support vector machine algorithm to classify lithofacies and model permeability in heterogeneous reservoirs. *Engineering Geology*, *114*, 267–277. https://doi.org/10.1016/j.enggeo.2010.05.005

[20] Ben-Hur, A., Horn, D., Siegelmann, H., & Vapnik, V. N. (2001). Support vector clustering. *Journal of Machine Learning Research*, *2*, 125–137.

[21] Chen, W., Xie, X. S., Wang, J. L., Pradhan, B., Hong, H. Y., Bui, D. T., Duan, Z., & Ma, J. Q. (2017). A comparative study of logistic model tree, random forest, and classification and regression tree models for spatial prediction of landslide susceptibility. *Catena*, *151*, 147–160. https://doi.org/10.1016/j.catena.2016.11.032

[22] Gutiérrez, S. L. M., Rivero, M. H., Ramírez, N. C., Hernández, E., & Aranda-Abreu, G. E. (2014). Decision trees for the analysis of genes involved in Alzheimer's disease pathology. *Journal of Theoretical Biology*, *357*, 21–25. https://doi.org/10.1016/j.jtbi.2014.05.002

[23] Ghimire, B., Rogan, J., Galiano, V. R., Panday, P., & Neeti, N. (2012). An evaluation of bagging, boosting, and random forests for land-cover classification in Cape Cod, Massachusetts, USA. *GIScience & Remote Sensing*, *49*, 5, 623–643. https://doi.org/10.2747/1548-1603.49.5.623

[24] Rodriguez-Galiano, V. F., Ghimire, B., Rogan, J., Chica-Olmo, M., & Rigol-Sanchez, J. P. (2012). An assessment of the effectiveness of a random forest classifier for land-cover classification. *ISPRS Journal of Photogrammetry and Remote Sensing*, *67*, 93–104. https://doi.org/10.1016/j.isprsjprs.2011.11.002

[25] Pierdicca, R., Malinverni, E. S., Piccinini, F., Paolanti, M., Felicetti, A., & Zingaretti, P. (2018). Deep convolutional neural network for automatic detection of damaged photovoltaic cells. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, *42*, 893–900. https://doi.org/10.5194/isprs-archives-XLII-2-893-2018

[26] Funaya, H., & Ikeda, K. (2012). A statistical analysis of soft-margin support vector machines for non-separable problems. In *2012 International Joint Conference on Neural Networks (IJCNN)* (pp. 1–7). IEEE. https://doi.org/10.1109/IJCNN.2012.6252443

[27] Cohen, G., Afshar, S., Tapson, J., & Van Schaik, A. (2017). EMNIST: Extending MNIST to handwritten letters. In *2017 International Joint Conference on Neural Networks (IJCNN)* (pp. 2921–2926). IEEE. https://doi.org/10.1109/IJCNN.2017.7966217

[28] Meshkini, K., Platos, J., & Ghassemain, H. (2020). An analysis of convolutional neural network for fashion images classification (fashion-MNIST). In *Proceedings of the Fourth International Scientific Conference "Intelligent Information Technologies for Industry" (IITI'19)*, *4*, (pp. 85–95). Springer International Publishing. https://doi.org/10.1007/978-3-030-50097-9_10

[29] Bi, W., & Kwok, J. (2013). Efficient multi-label classification with many labels. In *Proceedings of the 30th International Conference on Machine Learning*, *28* (pp. 405–413). PMLR.

[30] Zhang, Z. (2018). Improved Adam optimizer for deep neural networks. In *2018 IEEE/ACM 26th International Symposium on Quality of Service (IWQoS)* (pp. 1–2). IEEE. https://doi.org/10.1109/IWQoS.2018.8624183

[31] Chauhan, N. K., & Singh, K. (2018, September). A review on conventional machine learning vs deep learning. In *2018 International Conference on Computing, Power and Communication Technologies (GUCON)* (pp. 347–352). IEEE. https://doi.org/10.1109/GUCON.2018.8675097

Computer Science and Interdisciplinary Research Journal
Volume 2, Issue 1, 2025. ISSN: 3033-1218, DOI: 10.70862/CSIR.2025.0201-13

Innovations Classification Model in Online Sportsbook Betting

Petko Iliev

University of Plovdiv Paisii Hilendarski, 236, Bulgaria Blvd, 4027 Plovdiv, Bulgaria

petko.iliev@uni-plovdiv.bg

Abstract – This paper examines the horizontal and vertical cuts of online bookmakers. On this basis, possible classes and subclasses of innovations in online sports betting are theoretically presented. To increase the accuracy of the model, the definition of the groups takes into account the analysis of multiple websites in the field. A classification model with classes, subclasses, and groups covering currently available innovations is proposed.

Keywords – classification model, innovations, sportsbook, sports betting.

1. Introduction

The first online bet was made in 1996. It was organized by Intertops, a licensed online betting company from the small Caribbean Island Antigua and Barbuda. Zion Market Research, a consulting company, has conducted a study that suggests the global online sports betting market could exceed $150 billion by 2024. It has also found that sports betting now covers more than 70% of the global gambling industry [1].

The growth of online sports betting is partly a consequence of the many innovations introduced during its evolutionary development, and partly an obligation to continue the flow of innovation. The challenge is to create both highly effective innovations and the ability to analyze existing and future innovations. This is made possible by a multi-level classification that describes all currently implemented innovations. Classification is a step towards digitalization and facilitates the management and control of innovation. Utilizing the Internet and big data technologies, sports enterprises can deliver intelligent and customized services through online platforms, thereby achieving more precise market positioning and optimized user experiences [2].

Adapting to the digital environment raises many issues [3]:

(1) Clarifying and systematizing terminology;
(2) Accurately classifying the data under consideration and identifying the peculiarities of their structure [4];
(3) Promoting adaptation to the digital environment;
(4) Identifying and analyzing risks as a consequence of digitalization.

This article aims to propose a method for classifying innovations in the field of online sports betting. No such method was found in the analyzed literature. An approach for decomposing the studied entities and conducting a structural analysis of existing objects for classification was employed.

The existence of a method for classifying innovations in online sports betting will help those interested in the field to assess what has been achieved, on the one hand, and to set development goals, on the other.

2. Innovations Classification

The bookmaker is a licensed company that accepts bets and pays out winnings when the events on which the bets are placed occur [5]. The types of bets, i.e. the types of products that the company offers, define the horizontal structure of the company [6]. It is basically divided into the following parts:

DOI: https://doi.org/10.70862/CSIR.2025.0201-13

Corresponding author: Petko Iliev,
University of Plovdiv Paisii Hilendarski
Email: petko.iliev@uni-plovdiv.bg

Received: 14 May 2025.
Revised: 5 June 2025.
Accepted: 6 June 2025.
Published: 24 June 2025.

© 2024 Petko Iliev. This work is licensed under the Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License.

The article is published with Open Access at http://csirjournal.com/

(1) Betting on sports games (Sportsbook):
  (1.1) Real sports;
  (1.2) Electronic sports (eSports);
  (1.3) Virtual games – football, horse racing, etc. and any other games based on random number generation.
(2) Casino betting (Casino).
(3) Other (prediction of results of political elections or other social activities).

2.1. Creating Classes

The vertical structure of the company looks at the activities which the company carries out in order to be able to offer the products which define the horizontal structure. These are the functionalities that make up each horizontal element. The subject of this paper is sports betting. A decomposition of the vertical structure of this type of betting in a bookmaker is carried out in order to find several levels of depth of the structure and, consequently, all the functional elements that can be classes of the classification.

The functional diagram of the bookmaker's business (Decomposition Level 1) is shown in Figure 1 using a BPMN diagram.

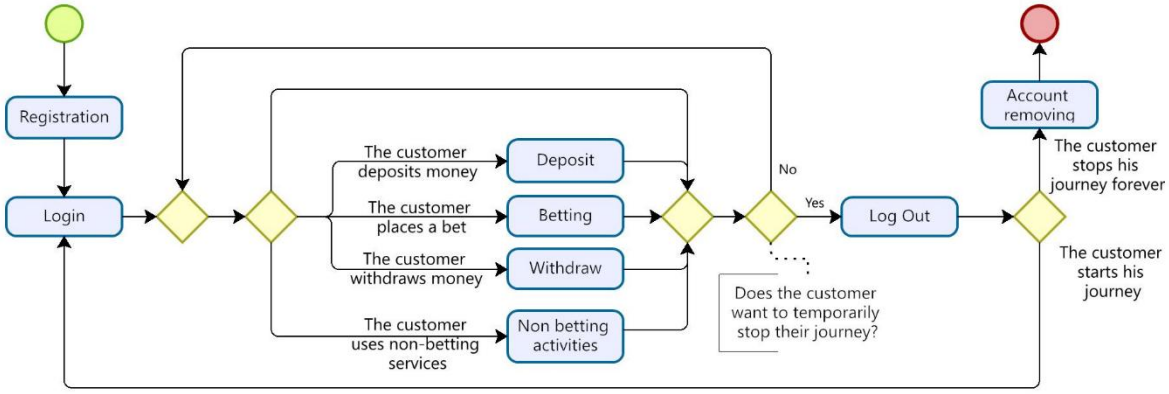


Figure 1. The details of the vertical structure of the horizontal element "Sportsbook" (Level 1)

In order for a person to become a customer of the bookmaker, they must first register and this is done in the "Registration" module. The next step is for the customer to access the bookmaker's online services and this is done via the "Log In" module. In order to be able to place a bet, the customer must deposit funds via the "Deposit" module. The actual betting takes place in the "Betting" module. After winning, the customer can withdraw funds via the "Withdraw" module. The "Non-Betting Activities" module includes activities for changing settings and participating in games that earn various bonuses, but the player is not involved in depositing and/or betting. If the customer no longer wishes to use the bookmaker's services, the "Log out" module will terminate the interaction with the website.

Betting innovations are usually promotions and bonuses designed to encourage activity around one or more features that generate positive results for the bookmaker. The only thing the bookmaker does not want is for the customer to stop betting forever. Withdrawals and stakes are neutral activities. Temporarily leaving the platform is undesirable but an unavoidable action. The other four functionalities ("Registration", "Login", "Deposit" and "Betting") are essential from a marketing and advertising perspective.

There are four classes defined on "Level 1": "Registration", "Log In", "Deposit" and "Betting".

2.2. Creating Subclasses

The decomposition of the Betting functionality is being deepened. "Level 2" is represented in Figure 2.

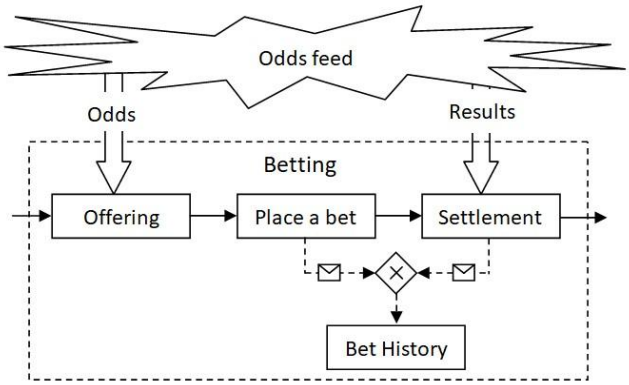


Figure 2."Betting" element functionalities (Level 2).

Odds are mostly fed into the online betting platform from an external source (Odds feed). If necessary, they are processed and offered to customers. This process is included in the “Offering” module. The functionality includes offering sports with all their markets and odds for each market outlet. Next is the actual betting process that is implemented in the “Place a Bet” module. Once the market is closed, bets are settled in the “Settlement” module. Each bet is checked after the market has closed. If it is profitable the payout amount is calculated and paid to the customer. “Bet History”

module stores detailed records of each bet, including all parameters and settlement information. This functionality offers limited potential for innovation.

Three subclasses of the Betting class are defined at level 2: “Offering”, “Placing bet” and “Settlement”. These are represented in Figure 3.

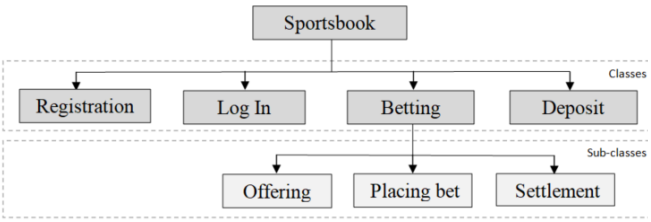


Figure 3. All classes and subclasses of sports betting.

2.3. Analysis of Existing Innovations and Creation of Groups

So far, the classes and subclasses for classification have been theoretically derived. In order to find all their possible groups to complete the classification model, a practical analysis of existing innovations in online sports betting websites will be carried out.

A search was conducted on Google using the keywords “top”, “bookmakers” and “online”. For analysis, the website top100bookmakers.com, which organically appeared in the first position, was selected [7]. From its Top 100 list, the 60 bookmakers with the highest number of daily visits were reviewed. These are listed below in alphabetical order: 8888, 10 Bet, 1xBet, 20 Bet, 22 Bet, Alpha Win, B Win, Batavia Bet, BC Game, Bet 365, Bet at Home, Bet Fair, Bet Fred, Bet Hard, Bet MGM, Bet Way, Bet Winner, Betano, Blue Chip, Borgata Online, Boyle Sports, Coral, Cash Point, Dafa Bet, Draft Kings, Ef Bet, Fan Duel, Fav Bet, Fon Bet, In Bet, KTO, Live Bet, Ladbrokes, Leon Bets, LV Bet, Mansion 88, Marathon Bet, Match Book, Mel Bet, Meridian Bet, Net Bet, Paddy Power, Palms Bet, Pari Match, Race Bet, Raja Bet, Rivalo, Sesame, Sky Bet, Sporting Index, Stake, TVG, Twin Spires, Unibet, V Bet, Virgin Bet, William Hill, Win Bet, Win Masters.

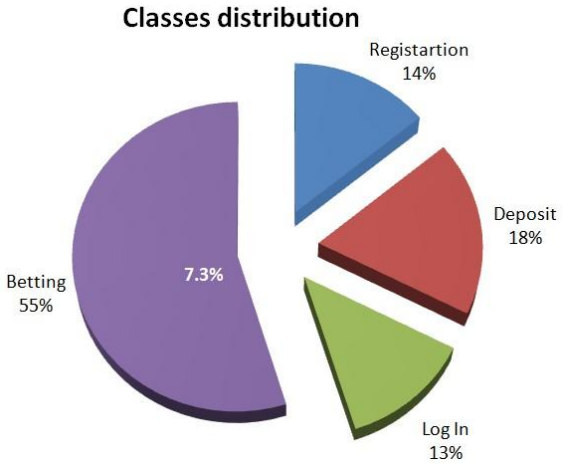


Figure 4. Distribution of innovations by class.

The distribution of innovations by class is shown in Figure 4. The absence of any class with zero innovations and the fact that all innovations are covered without the need for a new class ensures that the classes created are properly defined and sufficient.

The analysis found that the class "Registration" includes the actions "Welcome" and "Invite a friend". Two aspects can be considered in this class: the channel through which registration takes place and the fact of registration itself. There are no innovations related to the channel. Only the act of registration itself is relevant. This leads to this class having no groups.

All bonuses (free bet and lottery) that do not require a wager or deposit to receive can be classified as "Log In". There are two aspects that are the target of innovation: the channel used to log in and the frequency of use of the app. These create the two groups: the "Channel" and the "Sequence".

The Deposit class includes two types of bonuses: “Welcome”, which combines sign-up with a focus on the first deposit, and “Deposit”, which incentivizes each subsequent deposit as a consecutive one made on a day of the week or from a selected channel. These bonuses are categorized into three groups based on the focus of the innovation: “Amount” (deposit size), “Sequence” (frequency of deposits), and “Channel” (deposit method used).

The distribution of groups in the “Log In” and “Deposit” classes is highly imbalanced (Figure 5), but this does not change the fact that the groups are well defined and cover all innovations in a given class.

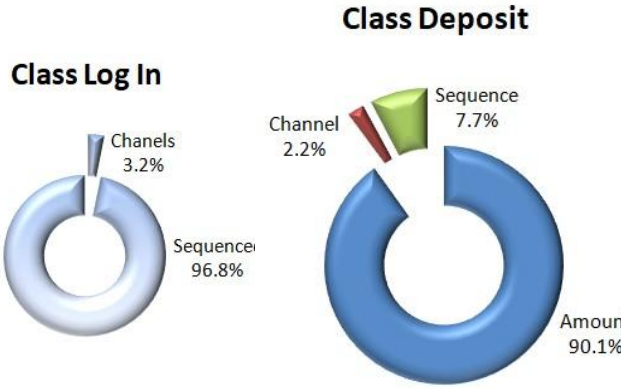


Figure 5. Distribution of the groups in the classes of the class "Log In" and "Deposit".

As expected, the class "Betting" attracts the most innovation and is therefore analyzed through its subclasses. The "Offering" subclass includes the "Bet Subjects" and "Odds" groups (Figure 6).

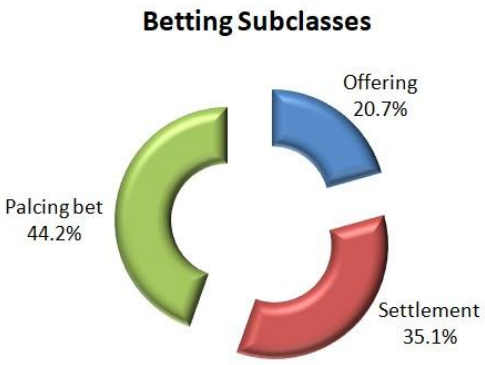


Figure 6. Distribution of innovations in the subclasses of the class "Betting".

Typically, when betting, the odds are offered by the bookmaker and used by the bookmaker's customer. With the development of the internet, person-to-person (P2P) betting has emerged. Here the odds are offered by the bookmaker's customers and used by other customers. The company only provides the technical support and charges accordingly. The first site to offer P2P betting was Flutter. It was followed by Bet Fair. Over time, Bet Fair absorbed Flutter and is now the most famous P2P betting site, but it is not the only one. The one who offers the odds and the one who uses them are the subjects of the transaction, and for this reason the group that classifies innovations that emphasize who is the subject of the bet is called "Bet Subjects".

The other subset is "Odds" and it includes all promotions that increase the odds of the bet. This can be a direct "Profit Boost" or by selecting leagues/games that increase the odds on markets offered by the bookmaker. Another approach involves lottery games in which one of the prizes is increased profit on a bet - effectively raising the odds. The "0% margin" promotion is relatively new and essentially increases the odds.

The largest number of groups is in the "Placing bet" subclass. The “Bet Type” group includes innovations that differentiate bets based on the number of markets and the number of games they involve One market from one game defines a "Single Bet". A bet with multiple markets and each from a different game is a "Multi Bet". A Bet Builder [8] is a bet on multiple markets from a single game. This is now offered as a promotion, but will probably soon become part of every bookmaker's standard. The same goes for the Multi Bet Bonus, which is now the most common bonus and almost every bookmaker offers it.It will likely soon become an integral part of the betting offer.

The "Kick-off" group includes all innovations relating to bet types according to when they were made relative to the start of the first game. If this happens before the start of the game (in the case of a single bet) or before the start of the first game (in the case of a multiple bet), the type is "Pre-match bet". If the bet involves only matches being played at the time of the bet, it is a “Live bet”. A multiple bet made during a match in progress but in combination with games not started defines a “Combined bet”. Currently, any promotions within this group are typically associated with “Live bets”

The "Channel" group includes innovations emphasizing the channel from which bets are made. These can be: “Website”, “Mobile Application”, “SMS”, “USSD” (Unstructured Supplementary Service Data). Currently, only innovations related to the mobile application are considered relevant [9].

The "Sports" group brings together all sports that are the target of promotions and bonuses.

The "Sequence" group was created to include innovations made to encourage more frequent betting. Typical examples include loyalty programs and various games linked to placing a bet. The group also covers bonuses awarded for specific betting actions.

The third subclass "Settlement" includes the "By Time", "Cash Back" and "Stake Refund" groups.

The "By Time" group combines "Usual payout", "Cash Out", and "Early payout". In "Usual payout", the payout occurs when the action in a market stops, i.e. the exits in the market are already known. This can be after the completion of a sports game, after the completion of a stage of the game or the occurrence of a market defining action. In a Cash out [10] where there is a multi-bet, there are completed sports games, the predictions are correct or there is a game in the bet (whether single or multiple) that is currently being played, the bookmaker may offer the option of prepaying some amount on the bet when the market is not closed. The customer decides whether to use the service, take the offered amount and risk no more, or not take it and risk waiting for the full bet amount, which they will take after the game is completed if the prediction is successful. Early payout [11] pays out the full amount as if the bet had won, but with the market not yet closed. This happens after certain conditions related to the game come true. The conditions may be different: reaching a certain goal (point) score or something else that classifies the game as very likely won by one player (team).

The "Cash Back" group includes innovations focusing on the return of an amount under certain circumstances. There are two cases: (1) where a percentage of the money lost is returned after a series of losses or over a period; (2) where a specific amount is returned when some event occurs that is part of the game but not directly related to the bet (e.g. every red card in the league on which the bets are placed).

The "Stake Refund" group is similar to the previous group, but returns the exact amount of the bet. This can happen (1) after a certain condition related to the game is met (for example, a 0:0 result) or (2) as a condition of an insurance-type promotion (for example, in the event of a failed pick on a single multiple bet game).

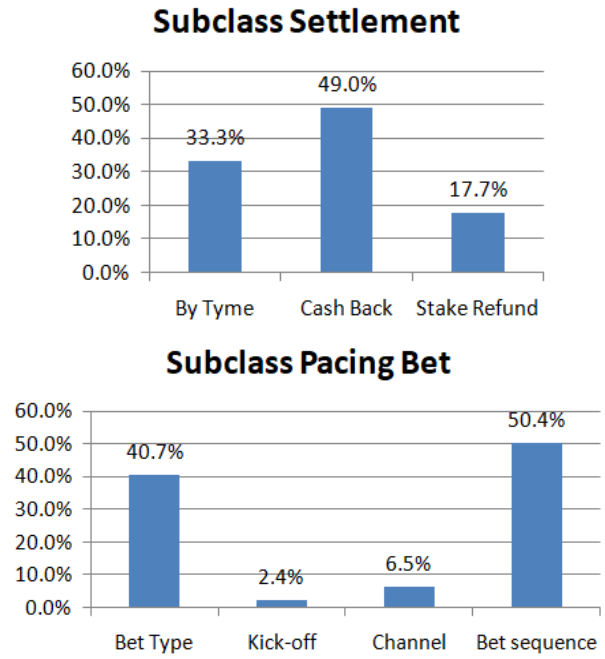


Figure 7. Distribution of groups in the subclasses "Settlement" and "Placing bet".

The final distribution of groups in the "Settlement" and "Placing bet" subclasses is shown in Figure 7. The "Offering" subclass groups are not shown because the innovations are almost 100% of the "Odds" group. The group "Bet Subjects" or the innovation "P2P" is present in only two websites. The group "Sport" is not included in the graph of the subclass "Placing bet" because it is a specific group that can be combined with the other groups of the same subclass. Almost always a promotion or bonus highlighting a "Bet Type", "Kick-off", "Channel" or "Bet Sequence" is associated with some type of sport.

3. The Result – Complete Classification Scheme

After the theoretical approach to create the classes and subclasses, and the practical analysis done to define the groups, the classification scheme can be considered complete. The effectiveness of the created classes and subclasses has been proven by the analysis of the innovations in place. The method used to create the groups inherently prevents redundancy or omissions. The model is open. If an innovation is created that is not classifiable with the existing classes, subclasses and groups, a new classification element can always be created to address the new requirement. This will enrich the proposed model. The final classification model is shown in Figure 8.

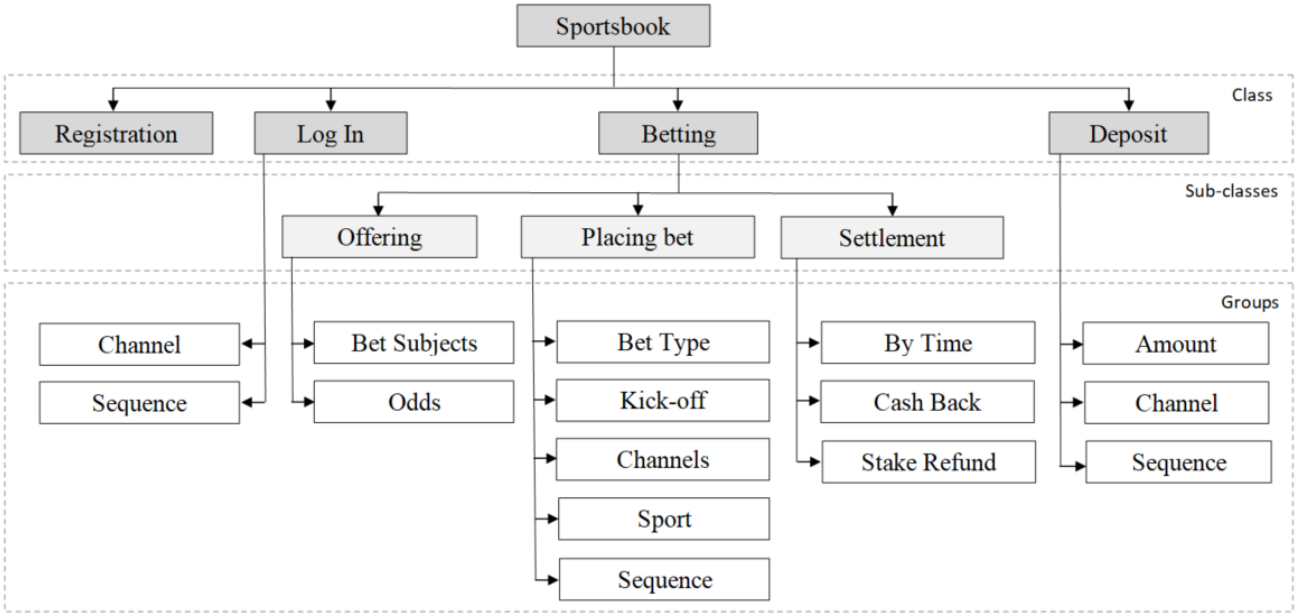


Figure 8. Classes, subclasses and groups in the classification scheme.

4. Conclusion

A model for classifying innovations in online sports betting is proposed. This helps each bookmaker to obtain an overview of the innovations they use to see which areas of potential they are covering and, if necessary, to build the right approach to increase the efficiency of their operations by incorporating new innovations.

Additionally, a weighting factor can be assigned to each group as part of a class or subclass, to each subclass as part of a class, and to each class. In this way, the proposed classification model can be

extended into a model for measuring the appropriateness of innovations. It will be based not on their outcomes, but on their necessity. The mathematical model should be developed using expert evaluation methods.

References

[1] Regulated United Europe. (n.d.). *History of sports betting*. Retrieved May 10, 2025, from https://rue.ee/blog/history-of-sports-betting/

[2] Li, Y., Kou, X., Li, Z., Ye, S. (2025) Dynamic mechanism and evolutionary game analysis of sports industry service transformation. *Scientific Reports 15*, 3381. https://doi.org/10.1038/s41598-025-88026-z

[3] Popchev, I., & Radeva, I. (2019). Risk analysis – an instrument for technology selection. *Engineering Sciences*, *56*(4), 5-29. https://doi.org/10.7546/EngSci.LVI.19.04.01

[4] Radeva, I. (2017). Multicriteria fuzzy sets application in economic clustering problems. *Cybernetics and Information Technologies*, *17*(3), 29-46. https://doi.org/10.1515/cait-2017-0028

[5] Chen, J. (2024). Bookie: Definition, meaning, duties, how they make money, and fee. *Investopedia*. Retrieved May 10, 2025, from https://www.investopedia.com/terms/b/bookie.asp

[6] Financial Dictionary. (n.d.). *Production structure*. Retrieved May 10, 2025, from https://credirect.bg/finansov-rechnik/struktura-na-proizvodstvoto

[7] Top 100 Bookmakers. (n.d.). Retrieved May 10, 2025, from https://www.top100bookmakers.com

[8] Play the Percentage. (n.d.). *What is a Bet Builder bet?* Retrieved May 10, 2025, from https://playthepercentage.com/blog/a-guide-to-bet-builders

[9] Goodin, Ch. (2024) Technological Innovations in Sports Betting: The Impact of Technological Advancements, Such As Online Betting Platforms, Mobile Apps, and Live Betting Features, *SSRN*. https://doi.org/10.2139/ssrn.4899419

[10] Maher, T. (2024). Sportsbook early cash-out betting: How does it work? *Forbes Magazine*. Retrieved May 10, 2025, from https://www.forbes.com/betting/guide/early-cash-out

[11] Fournier, J. (2024). Early payout explained – Full guide. *The Punters Page*. Retrieved May 10, 2025, from https://www.thepunterspage.com/early-payout-explained



https://csirjournal.com