Funded by the European Union

FALCON is funded under the Horizon Europe Framework Program
Grant Agreement ID 101121281

FALCON
Fighting Corruption & Organised Crime

Deliverable D4.2
Title: Corruption data acquisition and analysis toolset (R1.0)

Dissemination Level:	PU - Public
Nature of the Deliverable:	R
Date:	31/08/2024
Work Package:	WP4 - Corruption data acquisition and analysis tools
Editors:	CERTH
Reviewers:	ICCS, ENG, UCSC, C&T
Contributors:	UPV, GTI, UCSC, C&T, BPTI, SPH, VICOM

Abstract: This document presents the first version of the "Corruption Data Acquisition and Analysis Toolset (R1.0)," detailing the initial FALCON tools designed for extracting indicators, detecting anomalies, and analysing trends. It outlines the work completed in Work Package 4 (WP4) through various tasks, emphasizing the alignment of these tools with the project's objectives as specified in the Description of Action (DoA). The report includes a description of data management tools, tailored indicator extraction tools for specific corruption-related use cases, and a comprehensive overview of methodologies for anomaly detection and trend analysis.

Disclaimer

Revision History

Date	Rev.	Description	Partner
29/09/2023	0.1	Document template	IOSB
14/05/2024	0.11	Table of Contents	CERTH
28/06/2024	0.12	Anomaly detection	CERTH
31/07/2024	0.2	Company risk indicators	UCSC, C&T
01/08/2024	0.3	Car visual recognition tool	CERTH
02/08/2024	0.4	Trends analysis	CERTH
05/08/2024	0.5	Social network data/public websites indicators	SPH
09/08/2024	0.6	Procurement indicator descriptions	GTI
21/08/2024	0.7	Border corruption indicators	BPTI
22/08/2024	0.8	Executive summary, introduction and conclusions	CERTH
26/08/2024	0.9	Cryptocurrency indicators	VICOM
26/08/2024	0.95	Submitted for internal review	CERTH
29/08/2024	0.96	Final version	CERTH
31/08/2024	1.0	Review by the Coordinator - Submission	ICCS

List of Authors

Partner	Author
CERTH	Fotini Dougali, Vassilis Solachidis, Nicholas Vretos, Kostas Loumponias, George Koutidis, Rania Theodosiadou, Giorgos Gkarmpounis, Kostas Marthoglou, Christos Vranis
UPV	Francisco Perez
GTI	Bence Tóth
UCSC	Caterina Paternoster
C&T	Michele Riccardi, Andrea Carenzo
BPTI	Kostas Griska, Alfonsas Juršėnas
SPH	Kostas Tripolitis, Marios Zacharias

Internal Reviewers

Partner	Reviewer(s)
ICCS	Evgenia Adamopoulou, Theodoros Alexakis, Emmanouil Daskalakis, Nikolaos Peppes
ENG	Marco San Biaggio
UCSC	Caterina Paternoster
C&T	Michele Riccardi

Table of Contents

Index of Figures

Index of Tables

Glossary

AI	Artificial Intelligence
BADB	Blockchain Anomaly and Detection Benchmark
BCP	Border Control Points
BO	Beneficial Owners
BTC	BitCoin
COCO	Common Objects in COntext
CRM	Common Representational Model
DoA	Description of Action
DTW	Dynamic Time Warping
EM	Expectation-Maximization
ETL	Extract, Transform, Load
EU	European Union
GBM	Gradient Boosting Machine
GMM	Gaussian Mixture Models
IOU	Intersection Over Union
KDE	Kernel Density Estimation
KNN	K-Nearest Neighbours
LEA	Law Enforcement Agency
LOF	Local Outlier Factor
ML	Machine Learning
OCR	Optical Character Recognition
OSINT	Open Source INTelligence
PCA	Principal Component Analysis
REST	REpresentational State Transfer
RF	Random Forest
SoA	State of the Art
SVM	Support Vector Machine
UC	Use Case
UI	User Interface
URL	Uniform Resource Locator

VOC	Visual Object Classes
WP	Work Package
YOLO	You Only Look Once

Executive Summary

This document is the first version of the "Corruption data acquisition and analysis toolset (R1.0)" the report on the 1st release of FALCON tools for extraction of indicators from underlying data, anomaly detection, and trends analysis. The objective of this document is to provide a description of the tools developed in WP4 and present the work performed in Tasks T4.1, T4.3, T4.4 and T4.5 until month 12 of the project.

The tools outlined in this document have been meticulously developed to address the specific needs of the FALCON project, as articulated in the Description of Action (DoA). These tools are informed by a comprehensive analysis of all project deliverables produced to date, with particular emphasis on Deliverables D3.1 (Use cases and requirements), D3.2 (FALCON framework architecture) and D2.2 (Comprehensive list and definitions of corruption risk indicators). By aligning our tools with the project's objectives and requirements, we aim to enhance efficiency, facilitate collaboration, and ensure the successful implementation of the FALCON's goals.

Initially, the tools for ensuring and managing access to datasets within the overall FALCON framework architecture are described. These tools include all the required adapters for the FALCON heterogeneous data sources, a dataflow manager that will take care of data injection and data modelling that covers FALCON's corruption crime description needs. Furthermore, the dataflow management is described as well as the FALCON middleware. Finally, a short description of the datasets used for data collection is provided.

This document also reports on the work conducted in extracting indicators. A suite of tailored indicator extraction tools has been developed to meet the specific needs of end-users and data availability. These tools focus on border corruption indicators and car recognition (UC3), public procurement risks (UC1 and UC4), cryptocurrency and the analysis of social data networks encompassing all FALCON use cases.

Finally, a comprehensive State of the Art (SoA) is presented regarding the methodologies for anomaly detection and trend analysis that will be integrated into the upcoming versions of the FALCON tools. This detailed overview encompasses various innovative techniques, including machine learning algorithms, statistical analysis methods, and data mining approaches. The focus will be on enhancing the accuracy and efficiency of detecting anomalies in large datasets, as well as identifying emerging trends that can provide valuable insights for decision-making processes.

1. Introduction

1.1. Purpose of the Deliverable

This document presents a comprehensive report on the tools developed for the first version of the Corruption Data Acquisition and Analysis Toolset within the FALCON project. This toolset is designed to facilitate the extraction of key indicators from various underlying data sources, enabling effective anomaly detection and trend analysis.

The primary objective of this toolset is to enhance our understanding of corruption dynamics by providing robust analytical capabilities. By integrating advanced data processing techniques, the toolset will allow users to identify irregular patterns and potential red flags in data, which are critical for timely intervention.

Additionally, the data sources related to corruption are comprehensively catalogued, along with the specific information that are utilized for extracting indicators and creating visualizations. By systematically organizing these resources, we aim to enhance the clarity and accessibility of the data, facilitating more effective analysis and interpretation.

1.2. Relevance of D4.2 and Connections with Other Work Packages

The tools described in this deliverable are strictly related with the work performed in WP2, WP3 and WP5 as well as to the work in T4.2.

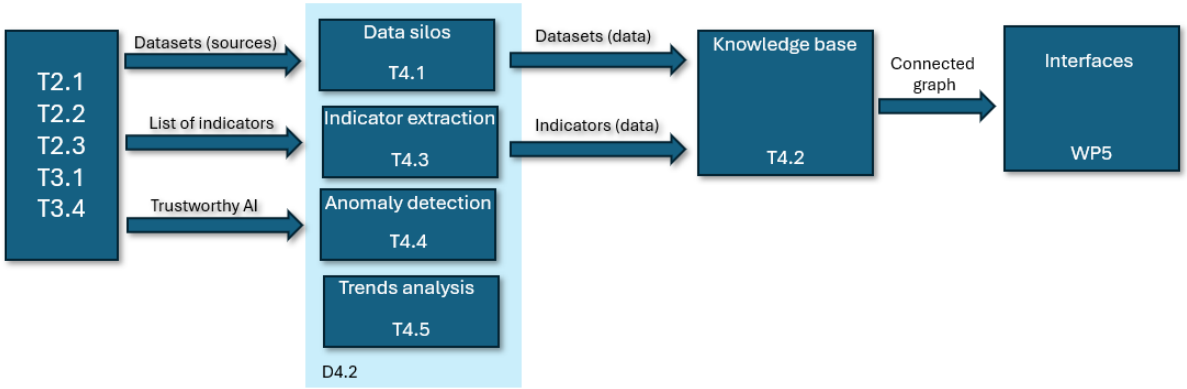

Figure 1. Relation of D4.2 with other tasks and WPs

The preliminary work conducted in WP2 and T3.1 has laid the foundation for identifying the specific datasets required for the research and development activities of the FALCON project. This initial phase has been crucial in ensuring that the data collected aligns with the project's objectives. Furthermore, T3.4 establishes comprehensive rules and procedures that must be adhered to in the development of trustworthy AI components. These guidelines are essential for fostering transparency, accountability, and ethical considerations in AI applications.

In T4.3, the indicators to be extracted and analysed have been meticulously defined in T2.2. This definition is vital for ensuring that the data collected is relevant and useful for subsequent analyses.

Finally, the data utilized and extracted in WP4 will be systematically stored in the FALCON knowledge base, as outlined in T4.2 and D4.1. This knowledge base will serve as a central repository for acquiring corruption-related data, encompassing both raw and processed data, along with the relevant indicators. These resources will facilitate effective visualization and analysis in WP5, ultimately contributing to the project's overarching goals.

1.3. Structure of the Deliverable

This deliverable is structured to present in detail the tools developed for corruption data acquisition and analysis. More specifically:

In Section 2, the developed approaches for the smooth collection, standardization, and organization of data are presented. Additionally, the various types of datasets are catalogued and linked to their corresponding use cases within the FALCON framework.

In Section 3 the tools for the indicators' extraction from different individual data sources and data types are presented. More specifically, the border corruption indicators are presented along with the car recognition tool, the company and public procurement risk indicators as well as the indicators extracted from cryptocurrency transactions and social network data and public web sites.

In Section 4, a comprehensive overview of state-of-the-art anomaly detection methods and applications is presented,

Section 5 is dedicated to trend analysis, where related work is initially introduced and an outline of the preliminary design of FALCON's trend detection tool is provided.

Section 6 concludes the document by summarizing the key findings and contributions of the deliverable.

2. Datasets and Data Silos

The success of the FALCON platform in fighting corruption relies heavily on its ability to gather, connect, and analyse a wide variety of datasets. These datasets often exist in separate, isolated silos, making it difficult to see the full picture needed for effective analysis. To combat corruption effectively, FALCON needs to bring all this data together, allowing for a more complete understanding and actionable insights.

The challenge is that these datasets come from different sources—like financial records, corporate databases, social media, and satellite images—and are often stored in ways that make them hard to access or combine. This separation of data creates barriers to analysis, making it harder to spot connections and patterns that could indicate corrupt activities.

To address this, FALCON is designed to break down these silos, integrating data from multiple sources into a single, cohesive data warehouse. This is not just about linking different databases; it involves creating an architecture that supports the smooth collection, standardization, and organization of data so it can be effectively analysed. The proposed architecture will also entail also the establishment of web-services and API connectors. FALCON uses advanced tools such as the FALCON interoperability enablers to ensure data from various sources is compatible, reliable, and ready for use.

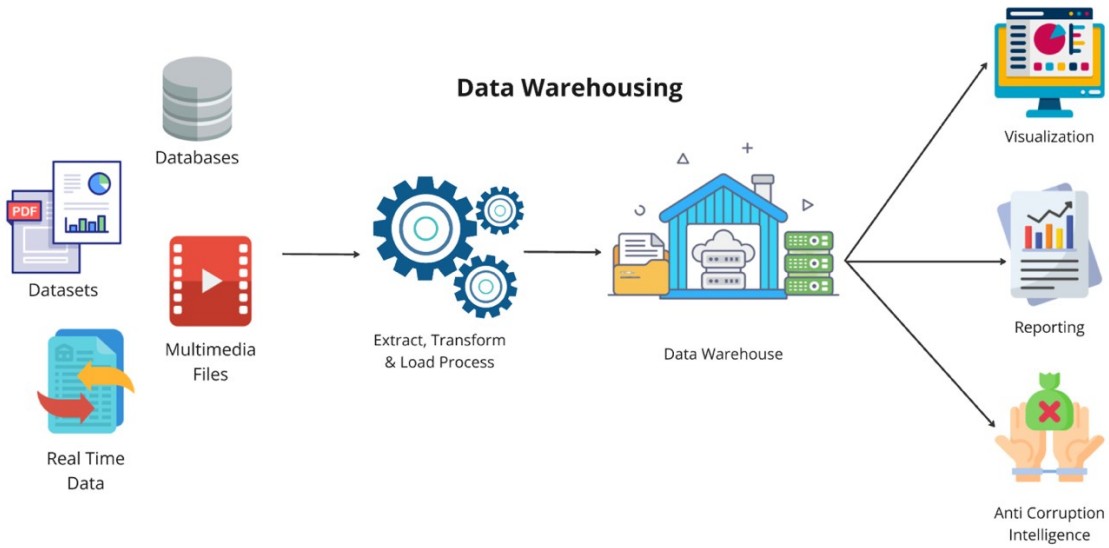

Figure 2. Data Warehousing concept

Moreover, FALCON's approach is flexible and scalable, meaning it can adapt to new data sources and changing conditions over time. This flexibility is key to ensuring the platform continues to provide valuable insights in the fight against corruption.

The following sections will first provide an overview of the identified data sources and their integration into the FALCON platform. After that, we will detail the strategies and tools used to connect, ingest, and process these datasets, ensuring that the platform can deliver the insights necessary to combat corruption effectively. By integrating and harmonizing these diverse sources of information, FALCON transforms scattered data into a cohesive and powerful resource for fighting corruption on a global scale.

2.1. Identified Data Sources and Integration Overview

A key element of the FALCON project is the effective integration and use of diverse datasets, which are crucial for developing and validating the tools aimed at detecting and combating corruption. These datasets, contributed by various project partners, cover a wide range of domains, including financial transactions, corporate records, open-source intelligence (OSINT), social media activity, and satellite imagery.

Each dataset has been carefully selected to align with the specific needs of the project's use cases. This ensures that the tools being developed are grounded in real-world data, making them more effective and reliable when applied to actual corruption scenarios. By using these datasets, FALCON can build robust analytical tools capable of addressing the complex challenges associated with detecting and preventing corruption.

The datasets play a central role in the project by providing the necessary data for training and validating the analytical tools. These tools will be used to process and analyse the data, uncovering patterns and insights that are critical in identifying corrupt activities. The datasets are used across multiple use cases, maximizing their utility and ensuring that the tools developed are versatile and adaptable.

Table 1. FALCON datasets linked to responsible partner and Relevant UC

Dataset Name	Partner	Relevant UC(s)
Opentender.eu[1] - Procurement data	GTI	UC1, UC4
Company –related indicators	UCSC	UC1, UC2, UC4
Bitcoin, Litecoin & Monero transactions	VICOM	UC2, UC4
Interest declarations	GTI	UC2, UC4
Sanctions Lists	UCSC and C&T	UC2
Vessel data	UCSC and C&T	UC2
PEP data	UCSC and C&T	UC2, UC4
Real Estate Data	MINT, MUP	UC2, UC4

[1] https://opentender.eu/start

Car detail (model, make, year, license plate)	CERTH	UC3
Copernicus Sentinel	TBC	UC2
IMF – Corruption Perception Index	ABI	TBC
International Consortium of Investigative Journalism (Panama papers)	UCSC and C&T	TBC
AIS Log	BPTI	UC2
OSINT / social media and Websites - news	SPH	UC3, UC4
BITCOIN Transactions	CERTH	UC2, UC4
World-bank: Global Public Procurement Database	ABI	UC1, UC4
International border transit records	BPTI/VSAT	UC3
Border Guard Deployment and Schedule VSATIS	BPTI/VSAT	UC3
Car model recognition	BPTI/VSAT	UC3
Criminal History and Identification	BPTI/VSAT	UC3
Vehicle Registration and ownership	BPTI/VSAT	UC3
Road cameras data	BPTI/VSAT	UC3

The integration of these datasets into the FALCON platform is essential for developing tools that are not only effective but also validated against real-world data. The collaboration of project partners in providing these datasets highlights the collective effort to address corruption comprehensively. By using a wide range of data sources, FALCON ensures that its tools are well-equipped to tackle various aspects of corruption, making the platform a powerful resource in the global fight against this pervasive issue.

2.2. Data Acquisition Strategy

The FALCON platform's data acquisition strategy is designed to gather, integrate, and harmonize a wide range of data sources, ensuring that the platform has access to the most relevant and comprehensive information for its analysis. This strategy is crucial because the effectiveness of FALCON's tools depends on the quality and diversity of the data they process. By pulling data from various domains—such as financial transactions, corporate records, open-source intelligence (OSINT), satellite imagery, and blockchain networks—FALCON creates a robust foundation for detecting and analysing corruption.

Each type of data is meticulously chosen not only for its relevance to uncovering corruption patterns but also for its compatibility with FALCON's analytical workflows. Financial

transaction data, for example, can reveal suspicious activities that might indicate money laundering or bribery. Corporate records help trace complex ownership structures that could hide illicit financial flows. OSINT, including social media and news reports, offers real-time insights into public perception and emerging corruption cases, while satellite imagery provides a bird's-eye view of physical assets and land use that could be linked to corrupt activities. Blockchain networks, with their transparent and immutable records, are invaluable for tracking cryptocurrency transactions that may be used in illegal activities.

To effectively collect, process, and integrate these diverse data sources, FALCON employs specialized tools such as Apache NiFi and Apache StreamSets. These tools act as powerful connectors and adapters, handling the unique technical requirements of each data source. Whether the data is coming from a structured financial database, an unstructured social media feed, or complex satellite data, these tools ensure that it is securely and efficiently transferred into the platform. They are designed to work in both real-time and batch processing modes, providing flexibility depending on the data's nature and the immediacy required by the analysis. Additionally, these tools are capable of transforming the data into formats that are optimal for FALCON's needs, ensuring that the information is not only collected but also ready for immediate use in analysis.

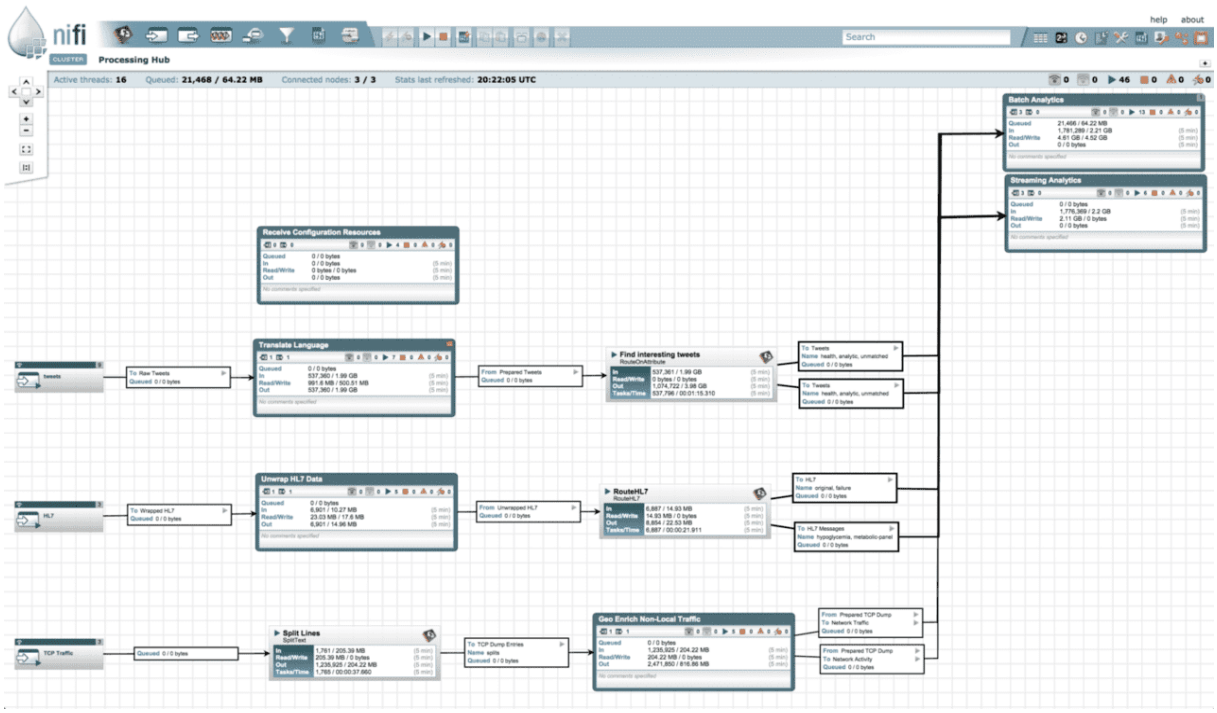

Figure 3. APACHE NiFi configuration

Figure 4. Apache STREAMSETS configuration

Once the data is collected, it enters FALCON's ingestion pipeline. This pipeline is responsible for the initial processing of the data, including its standardization. Standardization is a critical step because it ensures that all data—regardless of its source or original format—can be consistently and effectively integrated into FALCON's analytical models. By standardizing the data, FALCON can merge and analyse diverse data sets together, providing a comprehensive, unified view of potential corruption activities. This process makes the data more reliable and easier to interpret, which is essential for generating accurate, actionable insights that can support decision-making and investigative efforts.

2.3. Breaking Data Silos

Data silos are a significant challenge in large-scale data integration projects, particularly when dealing with sensitive and diverse information like corruption-related data. These silos can prevent different datasets from being combined effectively, limiting the ability to see the full picture needed for thorough analysis. FALCON tackles this issue head-on by implementing a comprehensive strategy to break down these silos, allowing for seamless data integration and a more complete understanding of corruption patterns.

2.3.1. Curated and Proprietary Datasets

FALCON brings together a variety of datasets provided by consortium partners, including third party corporate related indicators and carefully curated research datasets. These datasets are essential because they contain detailed, high-quality information that might not be available through public sources. Access to these datasets is tightly con-

trolled and managed through secure APIs, ensuring that sensitive information is protected and used in compliance with privacy and security regulations, as well as commercial constraints. This approach allows FALCON to tap into rich sources of data while maintaining the integrity and confidentiality of the information.

2.3.2. Open Source and Public Data

In addition to proprietary datasets, FALCON also leverages publicly available data. This includes open-source intelligence (OSINT) from social media platforms, news websites, and other online sources. Public data adds valuable context to the analysis, helping to build a broader and more nuanced understanding of corruption-related activities. For example, social media can offer real-time insights into public sentiment or highlight emerging issues, while news reports can provide background information and historical context. By combining these public data sources with proprietary information, FALCON enhances the depth and richness of its analysis.

2.3.3. Specialized Data Collection

For more complex and specialized data types, FALCON employs targeted data collection methods. For example, satellite imagery from Copernicus services is used to monitor land use and other physical changes that may be linked to corrupt practices. Similarly, blockchain transactions are tracked using dedicated connectors, allowing FALCON to follow the flow of cryptocurrencies that could be involved in illegal activities. These specialized datasets are integrated into FALCON's analytical workflows, providing unique insights that are otherwise difficult to obtain. By including these types of data, FALCON ensures that its analysis is comprehensive, covering a wide range of corruption indicators and providing a more complete view of potential issues.

2.4. Data Harmonization and Standardization

Given the diversity of data sources integrated into the FALCON platform, harmonization and standardization are critical to ensure seamless data integration and effective analysis. These processes are essential for transforming raw, disparate data into a consistent, unified format that can be efficiently analysed and interpreted across the platform. FALCON employs several key strategies and technologies to achieve this. In the next subparagraphs, different techniques will be presented.

2.4.1. Data Normalization

The first step in the harmonization process is data normalization. All incoming data is standardized to fit into a common representational model (CRM) that has been specifically designed for corruption-related analysis. This model ensures that data from various sources—whether it's structured financial data, unstructured social media content, or complex satellite imagery—can be interpreted and analysed uniformly within the platform. By normalizing the data, FALCON enables the integration of different datasets, making it possible to conduct comprehensive, cross-domain analyses that are crucial for uncovering complex corruption patterns.

2.4.2. Data Quality and Enrichment

Ensuring high data quality is paramount for the reliability of FALCON's analytical outputs. The platform incorporates processes for cleaning, enriching, and verifying the data. Data cleaning involves identifying and correcting errors or inconsistencies in the datasets, such as missing values, duplicates, or inaccuracies. Data enrichment adds context or additional layers of information to the raw data, making it more useful and insightful for analysis. For instance, financial transaction data can be enriched with metadata about the entities involved, geographical locations, or temporal patterns. Verification processes ensure that the data is accurate, complete, and relevant, which is essential for maintaining the integrity of FALCON's analyses. These quality assurance steps enhance the value and usability of the data, ensuring that the platform's outputs are both reliable and actionable.

2.4.3. Extract, Transform and Load Processes

To systematically prepare data for analysis, FALCON employs Extract, Transform, Load (ETL) processes. These processes are managed by advanced tools like Apache NiFi and Apache StreamSets. Apache NiFi handles the extraction of data from various sources, transforming it into a structured format that aligns with the platform's analytical requirements. Apache StreamSets further processes this data, adapting it to the specific formats and structures needed for effective analysis. Once transformed, the data is loaded into the appropriate storage solutions within the FALCON data warehouse.

The data warehouse in FALCON is a versatile storage system that supports various types of data, including binary files stored in MinIO, JSON data stored in MongoDB, and other formats stored in databases optimized for specific data types. This centralized storage system ensures that data is well-organized and easily accessible for a range of analytical tasks, from real-time data processing to historical data analysis. The data warehouse

serves as the backbone for FALCON's analytical processes, providing a reliable repository for both ongoing and future analyses.

2.4.4. Real-Time Data Integration with RedPanda

Although primarily a part of dataflow management, RedPanda also plays a role in maintaining the consistency and accessibility of data across FALCON. As a high-performance message broker, RedPanda ensures that standardized and enriched data can be distributed in real-time to various components of the platform. This integration allows for immediate use of newly ingested data, whether it's for live monitoring, model training, or historical trend analysis. The seamless flow of data facilitated by RedPanda further supports the harmonization efforts by ensuring that all components of FALCON can access the most current and consistent data available.

2.5. Integration and Accessibility

Ensuring that data is not only integrated but also accessible to the various analytical tools within FALCON is a key objective of the platform's architecture. The ability to efficiently manage and access data across different components is crucial for providing timely and actionable insights in the fight against corruption.

2.5.1. Centralized Data Storage

At the core of FALCON's data architecture is a central Data Lake, which supports the storage of structured, semi-structured, and unstructured data at scale. This centralized repository is designed to handle large volumes of diverse data types, making it possible to store everything from raw text and binary files to complex JSON structures. The Data Lake serves as the primary storage solution where all the collected data is securely housed, ensuring that it is readily available for analysis, model training, and historical data review.

2.5.2. Dataflow Management and Integration

The integration and flow of data within FALCON are managed to ensure that data moves smoothly and efficiently from one component to another. Tools like Apache NiFi and Apache StreamSets play a crucial role in this process. These tools are responsible for orchestrating the data as it flows from various sources into the platform. They handle the ETL processes, ensuring that data is properly formatted and ready for use as soon as it enters the system.

Once the data is processed, it is managed by RedPanda, a high-performance message broker that ensures real-time data accessibility across the platform. RedPanda acts as the central hub for all data communication within FALCON, allowing different components to publish, subscribe to, and consume data streams as needed. Whether the data is required for live analysis, ongoing model training, or historical data queries, RedPanda ensures that it is delivered in real-time to the appropriate components. This real-time dataflow capability is essential for maintaining the platform's responsiveness and ensuring that users can act on the most current information available.

RedPanda's role as a broker is critical for enabling asynchronous data processing, where different components of the platform can operate independently yet remain connected through a seamless data exchange mechanism. This setup allows for a highly modular and scalable architecture, where new components can be easily integrated into the platform without disrupting existing workflows. By providing a robust and flexible dataflow management system, FALCON ensures that all parts of the platform can efficiently access and use the data they need to perform their tasks.

2.5.3. Knowledge Base Database

In addition to the central Data Lake, FALCON includes a Knowledge Base Database (KB) built on Apache Jena. This Knowledge Base stores the results of various analyses conducted across the platform, following a CRM specifically designed for FALCON. The CRM ensures that data from different analyses is uniformly structured and easily accessible for further use. The Knowledge Base DB plays a vital role in storing, organizing, and making sense of the vast amount of information processed by FALCON. The data within the Knowledge Base is available for visualization and querying through the FALCON dashboard, allowing users to interact with the results of the analyses, track trends, and generate reports. This integration enables FALCON to deliver actionable insights that are not only based on raw data but also enriched by comprehensive analytical processing.

2.5.4. Security and Compliance

Given the sensitive nature of the data handled by FALCON, robust security measures are in place to protect data both at rest and in transit. The platform employs advanced encryption techniques to safeguard data, ensuring that unauthorized access is prevented. Additionally, FALCON adheres to international data protection regulations, such as GDPR, to ensure that all data handling processes comply with the highest standards of privacy and security. This includes implementing strict access controls, maintaining comprehensive audit logs, and ensuring that all data transfers are secure.

2.6. Addressing Future Challenges

The FALCON platform (Figure 5) has been designed with a forward-looking architecture that emphasizes scalability and adaptability, ensuring that it can effectively respond to future challenges as data volumes grow and new types of data sources emerge. This design was crucial for maintaining the platform's relevance and effectiveness in the dynamic and evolving landscape of corruption analysis.

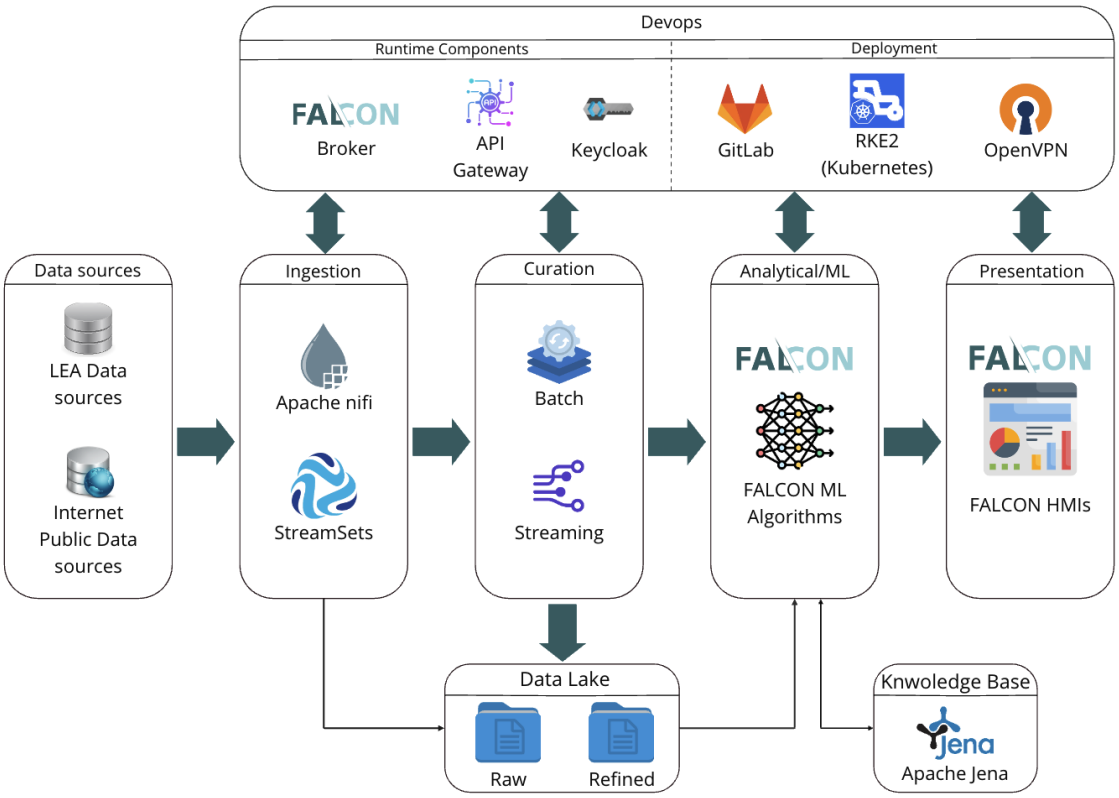

Figure 5. FALCON platform architecture

2.6.1. Scalability

As the volume and complexity of data continues to increase, FALCON's architecture is designed to scale seamlessly. The platform leverages cloud-based solutions (also deployable on-premises) and a microservices architecture, which allows it to expand its capacity without compromising performance. This scalability ensures that FALCON can handle larger datasets, more complex analyses, and an increasing number of concurrent users as the platform evolves. Whether integrating new types of data sources, processing more extensive data sets, or running more sophisticated analytical models, FALCON's infrastructure is equipped to grow and adapt as needed. This flexibility is key to ensuring that the platform remains at the forefront of corruption detection and analysis, capable of addressing both current and future demands.

2.6.2. Modular Design and Integration

FALCON's modular design allows for the easy integration of new connectors, data processing tools, and analytical components. As new technologies emerge or as the requirements of corruption analysis evolve, these modules can be added or upgraded without disrupting the existing system. This modularity is a cornerstone of FALCON's adaptability, allowing the platform to incorporate the latest advancements in data science, machine learning, and data processing. It also ensures that the platform can quickly respond to new forms of corruption and emerging threats, integrating new data sources or analytical techniques to stay ahead of these challenges.

2.6.3. Continuous Improvement:

The FALCON platform is built with continuous improvement in mind. It includes capabilities for regularly updating and refining data models, connectors, and analytical tools. This ensures that FALCON can adapt to new types of data and analytical methods, staying current with the latest developments in technology and corruption analysis. Continuous improvement processes are embedded within the platform, allowing it to evolve in response to user feedback, changes in the regulatory environment, or advances in technology. By continually refining its components, FALCON not only enhances its current capabilities but also ensures long-term sustainability and effectiveness.

2.6.4. Future-Proofing the Platform

FALCON is designed to be future-proof, meaning it can incorporate future technologies and methodologies with minimal disruption. This future-proofing involves regular updates to its core infrastructure, ongoing integration of cutting-edge technologies, and a commitment to maintaining high standards of security and data integrity. As new threats emerge and the tactics used in corruption evolve, FALCON's architecture is prepared to adapt, ensuring that the platform remains a vital tool in the fight against corruption for years to come.

3. Indicators Extraction

3.1. Company Risk Indicators

Company risk indicators are employed in the FALCON project in relation to three use cases (see Deliverable D3.1): Early detection of public corruption and public procurement fraud (UC1); Tracing of assets owned by "kleptocrats" and/or oligarchs in cases of corruption, money laundering and sanction circumvention (UC2); and Tracing conflicts of interest and asset self-declarations of politically exposed persons (PEPs) (UC4).

The risk indicators discussed in this section and integrated into the FALCON tool have been developed by Transcrime – Università Cattolica del Sacro Cuore in previous research[2], are delivered by its spin-off Crime&Tech and reflect risk factors highlighted by relevant regulations and guidelines at the international and national level, as well as by the relevant literature in the corporate and financial crime domain. The present subsection provides a description of the risk indicators, outlining the data sources used for their computation and referencing their previous applications (for details on the indicators, their rationale and background please see Deliverable D2.2). Indicators will be provided by Crime&Tech to the FALCON tools in real-time via API/web-services, upon a variety of search criteria (e.g. name of the firm, ID of the firm, name of director/owner, etc).

3.1.1. Risk Indicators

Ownership structure risk

These indicators indicate whether a company has an extremely complex and opaque ownership structure which may be employed for concealing illicit financial flows, including corruption and bribery transactions.

Complexity

Vertical complexity is a categorical risk indicator that measures the complexity of the ownership structure operationalized as number of layers of a company's ownership structure compared to its peers (size and sector), to detect possible anomalies not justified by its economic sector and dimension.

Horizontal complexity is a categorical risk indicator that measures the extent of a company's ownership structure – i.e., the number of nodes included between a company and its BOs compared to its peers (size and sector).

[2] Click **here** for more information.

Opacity

Opacity is a categorical risk indicator that measures the extent of ultimate beneficiaries of a company which are trusts, fiduciaries, foundations, pension funds, and other opaque legal arrangements that do not allow identification of beneficial owners/individuals.

Unavailability

Unavailability is a binary risk indicator which takes maximum value when a company does not have information on its ownership structure from official registers.

Anomalous share distribution

Anomalous share distribution is a binary risk indicator which takes maximum value when a company is controlled by shareholders holding shares just below the typical beneficial owner identification threshold (25%).

Territorial risk

These indicators indicate whether a company or its owners/directors come from high-risk territories, including secrecy jurisdictions (which do not allow full transparency of corporate ownership) or countries/regions characterised by high level of organised and financial crimes.

Country risk

Country risk is a categorical risk indicator that measures the level of risk of the jurisdiction(s) (a) in which a company is located and (b) their owners are located. The idea is to measure the exposure of a company's ownership structure to secrecy or high-risk jurisdictions. The score takes maximum value when at least one entity in the ownership chain is linked to a high-risk jurisdiction. By default, high-risk jurisdictions are those included in:

- the EU list of non-cooperative jurisdictions for tax purposes.
- FATF *Call for action* and *Other monitored jurisdictions* (known as FAT blacklist and grey list).

However, alternative lists of high-risk countries may be produced depending on the activity domain or the specific needs of the user.

Municipality risk

Municipality risk is a composite indicator measuring the risk of criminal infiltration for 8,000 Italian municipalities, combining a variety of information: crime and administrative statistics, intelligence information and evidence of infiltration at the economic and political level by Italian mafia groups. The indicator highlights the risk that a company could be involved in various criminal activities (e.g. organized crime, corruption, money laundering, fraud, financial crime).

Political exposure

Political exposure is a binary risk indicator indicating whether there are one or more Politically Exposed Persons (PEPs) or local administrators among all individuals involved in a company's network, i.e., administrators, shareholders, and beneficial owners. It includes:

- Politically Exposed Persons (PEP): Individuals who fall within the definition of PEP provided by FATF/GAFI. This includes a) National PEPs (current or former members of parliament and government, judges, military leaders, diplomats); b) International PEPs (Managers of international organizations); c) Other: family members of the previous categories.
- Local administrators: Local administrators in office at the municipal, provincial and regional level in selected countries (Italy, France, Spain, Czech Republic, Lithuania).

Adverse events

Adverse events is a binary risk indicator which indicates whether the presence of adverse events is detected for subjects (individuals or legal persons) included in the ownership structure, namely shareholders, directors, and beneficial owners. It includes:

- Adverse Media: companies and individuals associated with adverse events (e.g., investigations, crime allegations) reported by at least two independent media sources.
- Enforcement: companies or individuals subjected to enforcement provisions (e.g., arrests, judgments) and court filings around the world from various sources including national law enforcement reports, press releases and other statements from public authorities.
- Sanctions: companies and individuals included in one or more of the global screening and sanction lists issued by the following institutions: the United Nations, the European Union, the OFAC (Office of Foreign Assets Control) of the United States, the United Nations, the Bank of England, the US Federal Bureau of Investigation, the US Bureau of Industry and Security (BIS) and others.
- Offshore leaks: companies or individuals included in the 'Offshore leaks' journalistic investigations (e.g. Panama Papers, Paradise Papers and other ICIJ-related investigations).

Other risks

Anomalous age

Anomalous age is a risk indicator which takes maximum value when a company has owners and/or directors who are younger than 20 or older than 80 years of age.

Overall risk

The overall risk of a company is a categorical composite indicator which summarizes the previously mentioned risk dimensions.

3.1.2. Data Sources

Table 2 reports the data sources collected and processed to compute the company risk indicators presented in the previous section.

Table 2. Data sources processed for company risk indicators

Information	Data sources	Details
Company registers and BO registers	Moody's Analytics Orbis	WW coverage – 400+ million companies in 200+ countries. It includes company information, economic sector of activity, ownership structure, directors/managers, relevant legal events (merges, acquisitions), financial information from balance sheet.
Sanctions, enforcement, adverse media, PEPs	LexisNexis WorldCompliance	WW coverage – 2.5+ million detailed profiles.
Country black/greylists	EU lists of non-cooperative jurisdictions for tax purposes; FATF lists of non-cooperative jurisdictions (*Call for action* and *Other monitored jurisdictions*)	Latest information on jurisdictions included in an official anti-money laundering or tax blacklist or greylist from the European Union or FATF.
Offshore leaks	Crime&tech elaboration of Offshore leaks ICIJ	Pandora Papers (2021), Paradise Papers (2017), Bahamas Leaks (2016), Panama Papers (2016), Offshore Leaks (2013).
Local administrators	Ministero dell'Interno, Rèpertoire national des élus, Base de datos de Alcaldes y Concejales, Czech Statistical Office, Seimas Lietuvos Respublikos	Coverage: Italy, France, Spain, Czech Republic, Lithuania.

3.1.3. Previous Applications

The company risk indicators presented in Section 3.1.1 have been employed and tested in multiple studies, including [1],[2],[3],[4],[5]. For more information, consult the website

of **TOM – the Ownership Monitor**, the R&D hub for the analysis of business ownership structure founded by Transcrime and its spin-off Crime&Tech.

3.2. Public Procurement Indicators

Public procurement risk indicators are used in the FALCON project in relation to Use Case 1 'Early detection of public corruption and public procurement fraud' and Use Case 4 ‘Tracing conflicts of interest and asset self-declarations of politically exposed persons (PEPs)’.

The risk indicators summarised in this section have been developed in the process of preparing Deliverable 2.2 and will be integrated into the FALCON tool. They are based on risk factors put forward by the literature. Some of these indicators were already validated by the Government Transparency Institute, yet others were not yet tested within FALCON (with the intention to test them at the later stages of the project). This subsection provides a description of the risk indicators, outlining the definitions of the indicators, data sources and previous applications (see more details in Deliverable 2.2).

3.2.1. Risk Indicators

Tender Design

No Call for Tender

No call for tender is a categorical risk indicator that shows whether a call for tender was published. It flags high risk if no call for tender was published, based on call for tender information or URL.

Length of Submission Period

Length of submission period is a categorical risk indicator that assesses whether the length of the submission period for tenders restricts competition. It flags risk if the period is too lengthy or too short, using the call for tender publication date and submission deadline date.

Relative Length of Eligibility Criteria

Relative length of eligibility criteria is a numeric risk indicator that is based on the relative length of the eligibility criteria of a tender compared to its average length per market.

Call for Tender Modification

Call for tender modification is a categorical risk indicator that measures whether a call for tender was modified after publication. It flags risk if the tender was modified, based on the date of modification.

Weight of Non-Price Evaluation Criteria

Weight of non-price evaluation criteria is a numeric risk indicator that evaluates the percentage of non-price criteria in the tender evaluation process. A higher percentage indicates higher risk, assessed using the tender description with evaluation criteria

High Share of Missing Information About Tender

High share of missing information about tender is a categorical risk indicator taking value based on the share of missing information in tender documents. It flags risk if the missing information rate is higher than average for the country or market.

Tender Description Length

Tender description length is a categorical risk indicator that assesses if the tender description is too lengthy or too short. It flags risk if the description deviates from the average for the country or market, based on the tender description text and is associated with lower levels of competition.

Incomplete Information About Bidding Requirements

Incomplete information about bidding requirements is a categorical risk indicator that measures if bidding requirements are shorter than average for the market or missing from the tender description. It flags risk if the information is incomplete, based on the bidding requirements text.

Excessive Technical Specifications

Excessive technical specifications is a categorical risk indicator that evaluates if technical specifications are more lengthy than usual for the market. It flags risk if specifications are excessive, based on technical specifications for the tender.

Exclusive Prequalification Criteria

Exclusive prequalification criteria is a categorical risk indicator that measures if prequalification criteria are too lengthy or detailed than usual for the market. It flags risk if criteria are exclusive, based on the tender description text with prequalification criteria.

Evaluation phase

Single Bidding

Single bidding is a categorical risk indicator that measures whether only one bid was submitted for a tender. It flags risk if only one bid was submitted, based on the number of bidders.

Benford's Law

Benford's law is a categorical risk indicator that measures whether the distribution of the first digits of bid prices follows Benford's law. It flags risk if the distribution does not follow Benford's law, based on bid prices. Specifically, around 30% of numbers should begin with the digit 1, while fewer than 5% start with the digit 9. This means that the digit 1 should appear as the leading digit 6.5 times more often than the digit 9.

Relative Price

Relative price is a numeric risk indicator that measures whether the contract price is higher than the original tender estimated price by extracting the estimated price from

the final one. It flags risk if the contract price is higher, based on the contract price and tender estimated price.

Length of Decision Period

Length of decision period is a categorical risk indicator that assesses the appropriateness of the period taken to decide on the tender. It flags a risk if the decision period is too lengthy or too short, using the submission deadline date and contract signature date.

Suppliers Risks

Winner's Share of Issuer's Contracts

Winner's share of issuer's contracts is a numeric risk indicator that measures the percentage of contracts won by a supplier from the same buyer. A higher percentage indicates higher risk, based on supplier ID, buyer ID, and contract ID.

Tax Haven

Tax haven is a categorical risk indicator that measures whether a company's address is in a state considered a tax haven or with a high Financial Secrecy Index (FSI). It flags risk if the company is located in such a state, using bidders' city/country/postcode and a list of tax havens.

Remote Supplier

Remote supplier is a categorical risk indicator that measures whether the supplier's address is far from the implementation address. It flags risk if the supplier's address is remote, based on the bidder's address and implementation address.

Extreme Growth of Public Procurement Income After Ownership Change

Extreme growth of public procurement income after ownership change is a categorical risk indicator that measures whether a company's public procurement income significantly increased after an ownership change. It flags risk if there is extreme growth, based on bidder ID, ownership change date, and contract price.

Change in Public Procurement Income After Political Change

Change in public procurement income after political change is a categorical risk indicator that measures whether a company's public procurement income significantly increased after a government or mayor change. It flags risk if there is growth, based on bidder ID, company annual income, and changes in government/mayor/etc.

Large Contract Size Compared to Company Size

Large contract size compared to company size is a categorical risk indicator that measures whether the public procurement contract size is disproportionately high compared to the company's average yearly turnover. It flags risk if the contract size is large, based on bidder ID, contract price, and the company's average yearly turnover.

Extreme Profit Rate

Extreme profit rate is a categorical risk indicator that measures whether a company's profit rate is significantly higher than the market average. It flags risk if the profit rate is extreme, using market ID, bidder ID, and company annual profit.

Change in Ownership Before Winning Public Procurement Contracts

Change in ownership before winning public procurement contracts is a categorical risk indicator that measures whether there was a change in ownership before the company won public procurement contracts. It flags risk if ownership changed before winning, using bidder ID, contract signature date, and company ownership information.

Distinct Markets

Distinct markets are a categorical risk indicator that measures whether a supplier has a high ratio of the number of distinct markets to the number of contracts. It flags risk if this ratio is high, based on market ID and supplier ID.

Buyers risks

Buyer Concentration

Buyer concentration is a numeric risk indicator that measures the percentage of contracts awarded by a buyer to the same supplier. A higher percentage indicates higher risk, based on buyer ID, bidder ID, and contract ID.

Politically-exposed persons risks

PEP Judicial

PEP judicial is a categorical risk indicator that measures the presence of politically exposed persons in high-level judicial bodies. It includes members of higher courts, such as Supreme Courts, constitutional courts, or other high-level judicial bodies.

Political Party Officials

Political party officials is a categorical risk indicator that measures the presence of members from significant political parties, particularly those present in the parliament.

Members of Parliament or European Parliament

Members of Parliament or European Parliament is a categorical risk indicator that measures the presence of politically exposed persons who are members of national parliaments or the European Parliament.

Heads of State or National Government

Elected: elected heads of state or national government is a categorical risk indicator that measures the presence of elected politically exposed persons such as Presidents and Prime Ministers.

Non-Elected: non-elected heads of state or national government is a categorical risk indicator that measures the presence of non-elected politically exposed persons such as Ministers.

Regional and Local Government Officials

Elected: elected regional and local government officials is a categorical risk indicator that measures the presence of elected politically exposed persons such as Mayors and Governors.

Non-Elected: non-elected regional and local government officials is a categorical risk indicator that measures the presence of non-elected politically exposed persons who are members of government at the regional or local level.

State Owned Enterprise

SOE Supplier

SOE supplier is a categorical risk indicator that measures whether a company participating in public procurement is a state-owned enterprise (SOE).

Supplier Connected to SOE Company

Supplier Connected to SOE Company is a categorical risk indicator that measures whether the company participating in public procurement has board connection to another SOE.

3.2.2. Data Sources

Table 3 presents the data sources used for public procurement and PEP indicators calculation presented in the previous section.

Table 3. Data sources used for public procurement and PEP indicators

Information	Data sources	Details
Public procurement	OpenTender.eu portal	The portal presents contract-level public procurement data, which includes information from various national portals and validated corruption risk indicators developed under the Horizon 2020 project DIGIWHIST. OpenTender relies on sophisticated data collection and processing software that sources procurement data from the Tenders Electronic Daily (TED) and official national public procurement sources, which vary significantly in quality, scope, and accessibility. In some countries, data is collected from multiple sources, while in others, TED is the sole data source. In cases where open data is unavailable, information is extracted from thousands of web pages to create a structured database.
Company and BO register	Moody's Analytics Orbis	Data coverage – 400+ million companies in 200+ countries. It includes company information, economic sector of activity, ownership structure,

		directors/managers, relevant legal events (merges, acquisitions), financial information from balance sheet.
Politically exposed persons	Official national online repositories of politically exposed persons list	Information on national positions considered politically exposed with respective officials' names
Asset and interest declaration data	Official national online repositories of public officials' declarations	Financial assets (e.g. financial holdings and investments, properties, securities and stocks, trusts) and interest disclosures (e.g. memberships, positions and outside activities, spouse or partner's functions) of politicians and public office holders.

3.2.3. Previous Applications

Some of the public procurement risk indicators presented in Section 3.2.1 have been employed and tested in multiple studies, including [6], [7], [8], [9], [10], [11], [12], [13].

3.3. Social Network Data, Public Websites

3.3.1. Indicator Description

Search in social network data and public websites, from now on called OSINT Search, will be performed under the scope of the FALCON project to provide indication of financial corruption about a person or organization. The search will be combined with specified keywords and key phrases, from now on called risk terms (e.g. related to public, financial or government corruption, see more examples in Section 3.3.3), which will be related to corruption and conducted by the OSINT Tool of the FALCON toolset. Currently there are no OSINT indicators explicitly defined in D2.2, but they will be added in future deliverables after OSINT Search is evaluated by the end users.

3.3.2. Related Use Cases

OSINT Search will be used in Use Case 1 to support the tracing of public corruption, as well as in Use Case 4 to indicate conflicts of interest of politically exposed persons.

It is also applicable to Use Case 3, where it will be used to demonstrate the usage of social network data (Facebook) or assess a person's social status and identify an inconsistent lifestyle compared to one's legitimate income.

Due to the sensitivity of the data originating from social networks, in terms of privacy and ethics, Use Case 3 will be demonstrated with the use of synthetic data (see the Datasets chapter for more details).

3.3.3. Datasets

Two types of datasets will be created by the OSINT search: a dataset containing information from social networks (synthetic Facebook data) and one with information from public websites (news, public procurement, open sanctions list and others). Both datasets will contain information about persons or organizations that could be related to public corruption.

Although OSINT Search will be applied only to public posts, the Facebook dataset will be synthetic to respect data privacy and ethics related to the usage of personal information. It will contain text in JSON format and occasionally images (if required) in jpeg format. It will be used to demonstrate how data from social networks can help assess a border guard's status in terms of inconsistent lifestyle (Use Case 3). There will be no use of this data beyond FALCON and data will be accessed only by FALCON Analysis tools (if applicable) and FALCON authorized users. Access will be provided via REST endpoint, message broker or UI.

The Websites dataset will be comprised of actual data collected from publicly available websites. Such sites could be the following or any other sites with information that is publicly available and does not require login:

https://www.eprocurement.gov.cy/epps
https://ted.europa.eu
https://www.transparency.org/en/news
https://www.opensanctions.org
https://star.worldbank.org/asset-recovery-watch-database/

The Websites dataset will also contain text in JSON format and occasionally images (if required) in jpeg format. It will be used to demonstrate the use of open-source data in Use Cases 1 and 3. There will be no use of this data beyond FALCON and data will be accessed only by FALCON Analysis tools (if applicable) and FALCON authorized users. Access will be provided via REST endpoint, message broker or UI.

To create the above datasets, a series of risk terms will be required that will relate persons or organizations under search with corruption or luxurious lifestyles. Some examples in English are given below:

- Public corruption
- Financial corruption
- Government corruption
- Bribery
- Embezzlement
- Fraud
- Money laundering
- Kickbacks

- Misappropriation of funds
- Corruption scandals
- Cases of public corruption
- Financial corruption in government
- High-profile corruption scandals
- Anti-corruption measures
- Impact of corruption on economy
- Corruption in public sector
- Investigations into financial corruption
- Corruption prevention strategies
- Examples of government fraud
- Money laundering schemes
- Five-star amenities
- Luxury living
- World-class service
- Private jet
- Designer brands
- Gourmet dining
- Custom-built
- Prime location
- Exclusive access
- Tailored experiences

The sites and terms given above can be used as a starting point for the OSINT Search, but the actual definition of sites and terms should be done by LEAs and FALCON business experts, for OSINT Search to produce valuable results.

3.3.4. Methodology

The targeted URLs that will be searched, as well as the risk terms that will be used, will be mainly defined by the LEAs, although some URLs and terms will be provided to get them started.

The sources that will be considered for the FALCON project will be the public Web, from now on called Clearnet, as well as Facebook from social networks.

Regarding Clearnet, the URLs, which will be searched by OSINT web crawlers, will be publicly available websites containing information that could be used to trace public corruption (Use Case 1) and conflicts of interest of politically exposed persons (Use Case 4).

As far as Facebook is concerned, the sources will be synthetic Facebook accounts that will be populated with content related to Use Case 3. This approach will be followed to

address privacy/ethics restrictions associated with the consent of the data subjects to process their social network accounts. It is noted that end users will be able to add or modify the content of these mock accounts, to test the efficiency of the OSINT Search.

The results of the OSINT Search will then serve as indicators to LEAs for identifying potential suspects of public corruption. These search results cannot be considered as hard evidence though, since the search may return false positives. Therefore, they will always be assessed by an authorized official (human-in-the-loop principle).

To strengthen the output of the OSINT Search as evidence of corruption, the search results could be further analysed by the FALCON analytics tools, using state-of-the-art ML (machine learning) techniques. For that matter, search results will also be stored in the Knowledge Base.

3.3.5. Experimental Results

An example follows of how data from social networks will be stored in the OSINT Tool's repository and how it could be reflected in the Knowledge Base. The content given in the example is out of context, just for demonstrations purposes, because at the time of writing no corruption-related data was collected yet:

```
[
	{
		"content": "Has anyone realized yet that US soldiers in Iraq continue
to get depressed & end up committing suicide because they serve a dishonorable
cause that is a lie to the world? Isn't it clear that 'Democratic' France &
Belgium's actions to ban the niqab have proved that they harbor tremendous revulsion
for Islam itself? Obama's silent con-sent evidently conveys his 'new relationship'
with the Muslims. What major catastrophic event upon the Western world will its
leaders need in order to start actively listening to Usamah bin Ladin's demands?
Why are Americans afraid of racial profiling in Arizona? Is it because the false
sentiment of racism being vanquished is beginning to collapse? Why hasn't anyone
tried George W. Bush yet? Are one's crimes against human-ity detestable only during
their administration?  Why does the vast majority of Western media outlets refer
to jihadi media as 'propaganda' when every media outlet in the world has an agenda
to propa-gate for the purpose of altering mindsets in one way or another?",
		"entitytype": "post",→SocialNetworkPost-1
		"site": "https://www.facebook.com",→Website-1
		"source":
"https://www.facebook.com/Q9PY1UnX",→SocialNetworkAccount-1
		"creatorId": "Q9PY1UnX",
		"creatorName": "Ryley Secombe",
		"createdAt": "2023-07-31T09:43:22Z",→DateTime-1
		"id": "100059601672932",
		"friendOf": [
			"https://www.facebook.com/VDenis2000"→SocialNetworkAccount-2
		],
		"searchTerms": [
			"suicide",
			"Islam",
			"catastrophic",
			"crime"
		]
	}
]
```

Figure 6. Storage example the OSINT Tool's repository

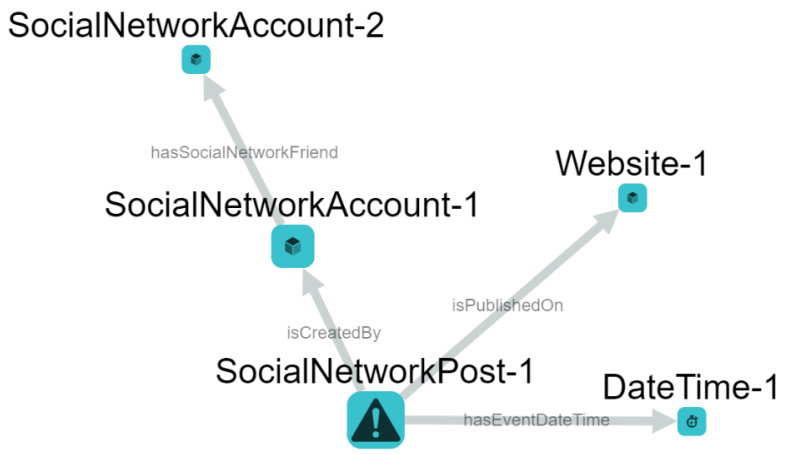

Figure 7. Social networks' data in the Knowledge Base

3.3.6. Future Work

Current work on OSINT Search includes the definition of the targets and risk terms that will be used in the search, the implementation of synthetic data for Facebook searches, the actual search (crawling) on the web and the storage of the search results in the Knowledge Base.

The work that needs to be done later is the definition of the FALCON tools that will analyse the data from the OSINT Search, the implementation of the integration points for interoperability with other FALCON tools and the provision of a UI for browsing the raw OSINT Search results and setting search targets and terms.

3.4. Border Corruption Indicators

In this subsection, we explore the extraction of key indicators that are crucial for identifying illicit activities at border checkpoints. The indicators discussed—Frequent Crossing, Collaborative Crossing, Short Visits, and Border Guards Collusion—are integral to detecting potential smuggling operations or corruption among border officials. These indicators will be applied within the FALCON framework in the context of Use Case 3. The following sections provide a detailed overview of each indicator, including their description, methodology and data employed for their extraction, and potential future work. Some of the indicators below can be extracted using the Car Visual Recognition tool (section 3.5).

3.4.1. Risk Indicators

Frequent Crossing

Frequent crossing refers to the repeated entry and exit of vehicles or individuals across the border within a short timeframe. This indicator highlights suspicious behaviour of vehicles or drivers who cross the border more often than would typically be expected, suggesting possible involvement in smuggling activities.

The methodology for identifying anomalously frequent crossings involves analysing the frequency of border crossings over a specified period. By setting thresholds that define what constitutes "frequent" crossing, the system flags vehicles and drivers whose crossing patterns deviate significantly from the norm. This data is then cross-referenced with other indicators to provide a comprehensive analysis.

Collaborative Crossing

Collaborative crossing involves a group of vehicles or individuals crossing the border in a coordinated manner. This indicator is often associated with organized smuggling operations, where multiple vehicles are used to distribute risk and cargo.

The system identifies groups of vehicles crossing the border together by analysing the time gaps between their entries. Typically, this involves detecting vehicles that cross within a 3-hour window of each other, at least once a month. This coordinated behaviour is flagged for further investigation.

Short Visits

Short visits refer to brief stays in the country, typically less than a day. This indicator is particularly relevant for identifying individuals who may enter the country solely for illicit activities, such as unloading smuggled goods, before quickly exiting.

The system calculates the duration between entry and exit times, flagging any visits shorter than the given threshold. These short visits are then analysed in conjunction with other indicators to determine if they warrant further investigation.

Border Check Post Clearance through a Single Border Guard

This indicator focuses on identifying patterns where specific vehicles or individuals are consistently cleared by the same border guard. Such patterns may suggest a corrupt relationship, where the guard allows illegal activities to pass unchecked.

To identify potential collusion, the system analyses border crossing records to find cases where there is a 100% correlation between a specific border guard and certain vehicles or individuals. These patterns are flagged for closer scrutiny.

3.4.2. Objectives

The methodologies for each indicator involve analysing patterns in the data available to law enforcement agencies (LEAs), whether it be the frequency of crossings, the timing and coordination of group crossings, the duration of stays, or the consistency of border guard involvement. While each indicator focuses on different anomalies encountered by authorities, they cannot individually identify corrupt border guards. Therefore, by analysing these indicators collectively, we can build a comprehensive model that enhances our ability to detect and address illegal activities at border checkpoints.

Future work will focus on refining the thresholds and algorithms used for detecting these anomalies, expanding the data sources to include external factors like social media analysis, and automating suspicious car model identification by employing a video stream analysis using artificial intelligence. These modules will be then integrated into a unified detection system.

3.5. Car Visual Recognition Tool

3.5.1. Related Use Case and Indicator Description

The necessity for creating the tool discussed in this section has emerged within the context of Use Case 3 (UC3), aimed at tracking and investigating corruption activities at Border Control Points (BCPs). Organized smugglers are utilizing cars to illegally transport illicit goods through border checkpoints, presenting a significant challenge for law enforcement agencies in tracking them due to the extensive data analysis required. FALCON can provide a solution by flagging suspicious activities during the "car-crosses-border" process for further investigation. More specifically, potential patterns that may indicate corrupt practices can be identified by comparing features across datasets. The development of a Car Visual Recognition Tool that utilizes computer vision and image processing for analysing cars based on their visual attributes, such as model and license plate details, is crucial for extracting such features. Moreover, it is proved that capturing also the arrival time of cars at the Border Control Point presents another valuable indicator that can be extracted through the Car Visual Recognition Tool since factors signalling a high likelihood of corruption in the context of Use Case 3 (UC3) include:

1. frequent crossings of a vehicle at the Border Control Point (Car Visual Recognition Tool outputs that are used: *car license plate, time*)
2. vehicle crossings during the same border guard's shift (Car Visual Recognition Tool outputs that are used: *car license plate, time*)
3. use of specific car models commonly associated with smuggling (Car Visual Recognition Tool outputs that are used: *car model*)
4. extended duration of a vehicle's stay in the Schengen zone (Car Visual Recognition Tool outputs that are used: *car license plate, time*)

It is noteworthy to mention here that the duration of time each car spends waiting at the BCP could serve as an additional indicator signalling a low or high likelihood of corruption. By extracting the arrival and departure time of the car regarding the "car-crosses-border" process, we can compute the aforementioned indicator and therefore the list of the factors that indicate a red flag for corruption can be updated as follows:

1. frequent crossings of a vehicle at the Border Control Point (BCP) (Car Visual Recognition Tool outputs that are used: *car license plate, time*)

2. vehicle crossings during the same border guard's shift (Car Visual Recognition Tool outputs that are used: *car license plate, time*)
3. use of specific car models commonly associated with smuggling (Car Visual Recognition Tool outputs that are used: *car model*)
4. extended duration of a vehicle's stay in the Schengen zone (Car Visual Recognition Tool outputs that are used: *car license plate, time*)
5. short period of time staying at BCP (Car Visual Recognition Tool outputs that are used: *car license plate, time*)

3.5.2. Datasets

The datasets utilized within the framework of the Car Visual Recognition Tool can be divided into two different categories:

Datasets used for deep learning model training:

1. Stanford Car Dataset[3] : A publicly available academic dataset for vehicle visual imagery that comprises 16,185 images across 196 car classes. The dataset is divided into 8,144 training images and 8,041 testing images. The classes within the dataset are structured around Year, Make, and Type (e.g., 2012 Tesla Model S), encompassing a total of 49 Car Make classes and 18 Car Type classes.

2. License Plate Recognition Dataset[4]: A publicly available academic dataset for vehicle visual imagery that encompasses 24,312 images of license plates, with a distribution of 21,174 images for training, 2048 images for validation, and 1090 images for testing.

Datasets used for evaluation:

1. Road camera data for border crossing vehicle recognition: Specifically, footage from a local Border Crossing Point (BCP) was used to analyse the environment and the quality of car images. The camera footage was provided by VSAT and BPTI, allowing us to tailor our algorithm in order to meet the specific requirements.

3.5.3. Methodology

This section delineates the comprehensive development of the Car Visual Recognition Tool. The tool comprises two distinct sub-modules, namely:

1. The Car Detection and Model Classification Module, and

2. The Car Label Detection and Recognition Module.

Car detection and model classification module

The "Car Detection and Model Classification module" comprises of two main components: the car detection component and the car model classification component. The algorithm processes a recorded video by analysing each frame individually passing

[3]https://www.kaggle.com/datasets/jutrera/stanford-car-dataset-by-classes-folder
[4]https://universe.roboflow.com/utech-susnq/license-plate-detection-wtqh2

it initially to the Car Detector to detect cars crossing the Border Control Point. When a car is detected, the algorithm returns the bounding box of the car which then is forwarded to the Car Model Classifier for the identification of the car model.

Regarding **Car detection**, the state-of-the-art YOLO[5] (You Only Look Once) algorithm is employed at the Border Control Point. YOLO is a real-time object detection network that efficiently identifies objects and returns their bounding boxes in a single pass. By segmenting the image into a grid, YOLO predicts bounding boxes and probabilities for each grid cell. YOLO has been trained on diverse datasets such as COCO (Common Objects in Context) and VOC (Visual Object Classes) to recognize a wide array of objects in various scenarios. Specifically tailored for our application, YOLOv5 has been trained on the COCO dataset, renowned for its comprehensive object categories and annotations, enabling precise object detection in different environments.

To fulfil the requirements of the UC3 objectives, our algorithm has been adjusted to accept as input only a specific region of the frame, namely the area that captures the car passing through the BCP. This adjustment effectively filters out any extraneous car detections that may appear in the background, allowing the algorithm to concentrate solely on the single car with the greatest visibility. This targeted approach enhances the algorithm's speed and robustness by prioritizing the detection of the closest and most discernible car, as distant detections tend to yield less reliable information. Figure 8 depicts the aforementioned adjustment with the yellow rectangle indicating the specific region of the frame that we consider.

[5]https://github.com/ultralytics/yolov5/releases

Figure 8. Frame region where the car detection is applied

Regarding the **car model classification** aspect, a pre-trained model[6] was employed as an initial reference to identify the model of each car. The model receives the bounding box of each car, as derived from the car detector, and subsequently outputs its model that constitutes of three parts: make, type, and year. To elaborate, the cutting-edge deep learning architecture ResNeXt50 was utilized within a multi-task-learning framework, train on Car Model (196 classes), Car Type (18 classes) and Car Make (49 classes) classification tasks. The training data for this model was sourced from the Stanford Cars dataset.

Tracking mechanism

To meet the UC3 objectives, we argued that the integration of a tracking mechanism into our system constitutes a requirement for extracting the requested indicators. This addition is necessary to monitor the identified region for each car across frames and consolidate the extracted data. We have opted for the Deep-Sort tracker[7] to incorporate tracking into our algorithm, as it strikes a balance between speed and accuracy. The tracker receives the detections provided by the Car Detector as input, assigns a unique identifier to each detected car, and monitors their movements across frames while maintaining consistent ID assignments. This approach enhances our understanding of the detections by associating additional identity information throughout the frames.

Since the objective of this algorithm is to be used on a specific use case (UC3) where the camera is set in a predefined location and the car positions are defined, a need for adjusting the tracker's parameters has arisen. In the first experiments it is realized that in some cases the tracking algorithm could not distinguish the tracks of two successive cars, since they seem to be located at the same place especially when they are close to each other. Namely, the last bounding box of car A at the time *t* has a small Intersection over Union (IoU) with the first bounding box of car B at the time *t+1*. For this reason, we have adjusted the tracker's parameters preventing the tracker from incorrectly assigning the same identifier to two distinct cars that traverse the same area of the frame, as an IoU distance is calculated between different bounding boxes to assign an identifier to each car.

Voting process

Having tracked the designated detected region for each car across multiple frames, we proceed to the next stage, which entails post-processing of the extracted model detection results to generate the final output of our algorithm. Since the model output for each instance of the specific car is not always the same, a voting process that will result in a unique model output for all the detections of the tracked trajectory is involved.

[6]https://github.com/kamwoh/Car-Model-Classification

[7]https://github.com/nwojke/deep_sort

More specifically, throughout the "car-crosses-border" process, the algorithm's results are recorded for subsequent post-processing. As also mentioned above, a decision must be made among the various predictions generated by the Car Model Classifier for each frame. Our approach employs a voting mechanism whereby the final output of the Car Model Classifier is determined by the prediction that prevails for each car throughout the entire detection process. The voting process must be expanded to incorporate the output of the car label detection and recognition module. Detailed information is provided in the following section regarding the car label detection and identification module along with a JSON file that illustrates the output of the entire process (Figure 10).

Car label detection and recognition module

In a manner akin to the Car Detection and Model Classification module, the "Car Label Detection and Recognition module" also comprises two distinct components: the license plate detection component and the license plate recognition component. The **license plate detector** operates by taking a frame as its input and detecting the license plates of the cars that are crossing the BCP. We chose to apply the car label detection to the entire frame instead of just to the region of the car detection output, since the car label detection model was trained on a dataset of various vehicles, where some of them are partially visible while others cover only a small part of the image. This detection process yields the car label bounding box coordinates, which are subsequently fed into the license plate OCR recognition component. Notably, a pre-trained license plate detector[8] based on the YOLO algorithm was integrated into our process for the car label detection segment. The YOLO algorithm can be utilized for the detection of car license plates within the framework of an object detection task. Through training the YOLO model on a dataset comprising images of vehicles with visible license plates, the algorithm can be trained to identify and localize license plates within an image. The YOLO algorithm processes the entire image simultaneously and generates bounding boxes around identified objects, including license plates, along with associated confidence scores. Specifically tailored to our application, the model was trained using the YOLOv8 on the License Plate Recognition Dataset. Given that we have refined the algorithm to accept as input solely the area that captures the car passing through the BCP, thereby eliminating any extraneous car detections that may occur in the background, we achieve the detection of a single car in each frame. Consequently, there is no requirement for the association of the car and its corresponding license plate, as each car is automatically linked to the uniquely detected license plate with every instance. However, we have developed such a function that associates the detected license plates with the detected cars in each frame, particularly in scenarios where multiple cars are detected, for

[8]https://github.com/Muhammad-Zeerak-Khan/Automatic-License-Plate-Recognition-using-YOLOv8?tab=readme-ov-file

experimental purposes outlined in Section “Car detection and model classification module”.

In reference to the car license plate recognition component, the EasyOCR algorithm[9] is used to recognize the detected license plate of each car passing through the BCP. This algorithm conducts optical character recognition (OCR) that enables machines to identify and interpret printed or handwritten text characters from images. The specific output includes the license plate string sequence along with an associated confidence score. To make it easier for the OCR technology to read license plates, we applied image processing (filters) on the cropped license plates. Specifically, we converted the images to grayscale and then applied a threshold to convert the image to black and white. Once the easyOCR reader has been applied, we convert the detected text to uppercase and remove all the white spaces. As an initial step, we selected a particular format of license plates to work with. The United Kingdom license plate format was selected. Having observed the difficulty in distinguishing between numbers and letters when utilizing OCR technology (e.g., discerning between the number 5 and the letter S), we have developed an additional algorithm. This algorithm converts letters resembling numbers and vice versa, based on the specific region of the license plate being localized. To elaborate, in cases where the OCR technology identifies a character as an "S" but we are certain that it should be interpreted as the number "5" based on its position within the license plate (where a number is expected), we will convert the character "S" to the number "5." Concerning the voting process and utilizing the tracking information obtained for each car, we selected as the final output of the car label detection and recognition module the output with the highest corresponding confidence score. In the next version of the tool car label format from more countries will be supported.

Time in, time out, duration of stay at BCP – extra indicators

The indicators extracted so far include the model (make, type, year) and the license plate of each car that crosses the BCP. As also mentioned in Section 3.4.1, the arrival time of cars and their duration of stay at the BCP could serve as valuable information in the context of addressing corruption within UC3. The extraction of the aforementioned information can be feasible by identifying the frame in which the car is firstly detected (“frame_in”) and the frame in which the car is lastly detected (“frame_out”) during the "car-crosses-border" process. The detected frames can be then converted into timestamps providing thus the arrival time and the duration of stay at BCP for each car. Consequently, the integration of tracking within our algorithm proves essential not only for the post-processing of the extracted indicators so far, but also for the extraction of these additional indicators.

[9]https://github.com/JaidedAI/EasyOCR

Upon post-processing the algorithm's results we can derive the final output for each car that crosses the BCP as illustrated below:

1. Model - Make, Type, Year
2. License Plate Number
3. Arrival time at BCP (time in)
4. Duration of stay at BCP (time in, time out)

All the above extracted elements furnish a comprehensive compilation of information pertaining to each car and they will be stored in the Knowledge Base (KB) for further processing, in order to find a relation to factors that could indicate a corruption case. The figures below (Figure 9 and Figure 10) illustrate the entire procedure to enhance understanding.

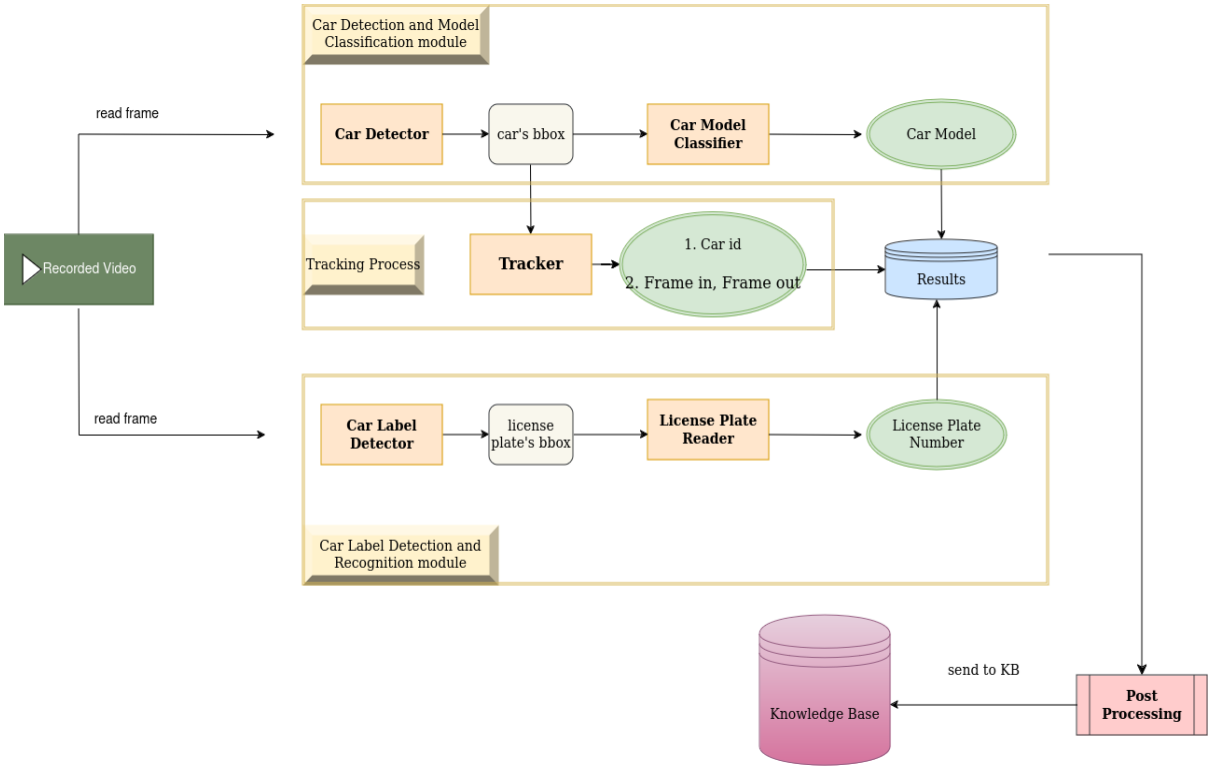

Figure 9. The Car Visual Recognition Tool


```
{"1": {"CAR_ID": 1, "FRAME_IN": "{32}", "FRAME_OUT": "{900}",
"CAR_MODEL": "BMW", "LICENSE_PLATE_NUMBER": "[GX15UGJ]"}, "3":
{"CAR_ID": 3, "FRAME_IN": "{910}", "FRAME_OUT": "{924}",
"CAR_MODEL": "Mercedes-Benz", "LICENSE_PLATE_NUMBER": "[APO5JEO]"},
"4": {"CAR_ID": 4, "FRAME_IN": "{955}", "FRAME_OUT": "{1665}",
"CAR_MODEL": "smart", "LICENSE_PLATE_NUMBER": "[KHO5ZZK]"}}
```


Figure 10. Tool's output in a JSON format

3.5.4. Experimental Results

In this section, we present the experimental results generated by our tool. Our experiments utilized a recorded video sourced from a local Border Control Point, provided by VSAT and BPTI, alongside a sample video for a traffic camera downloaded from the web. We present the two illustrative examples that demonstrate the tool's versatility in analysing varied situations, showcasing its output and the insights it can deliver. To clarify, the reason we chose to employ the sample video was that the resolution of the recorded video was too small, preventing us from detecting or identifying the car's license plate. Consequently, we opted to utilize an alternative video in which the license plates are sufficiently visible. This adjustment allowed us to effectively test the Car Label Detection and Recognition module. Figure 12 illustrates the results of this procedure, demonstrating the capabilities of our system in identifying and recognizing car license plates. As also mentioned earlier, regarding the sample video, we adapted the initial algorithm by developing a function that matches detected cars with their corresponding license plates, as such correspondence is essential when the number of detected cars are more than one.

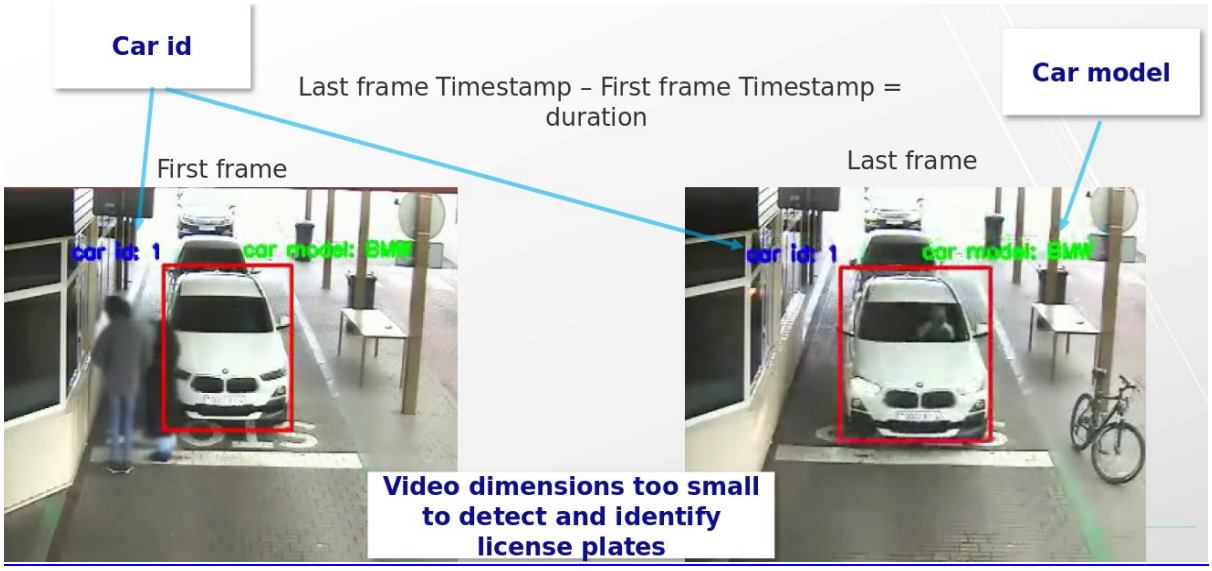

Figure 11. Tool implementation using a recorded video from a local BCP

Figure 12. Tool implementation using a sample video featuring clearly license plates

It is noteworthy to mention here that our algorithm can be adjusted in order to receive as input both a recorded video and a video stream in order to support a real-time application. If this is the case, the voting process is modified to align with real-time demands. Last but not least, we conducted experiments both on CPU and GPU platforms and we verified that both set ups fit the UC3.

3.5.5. Future Work

The end-users consider useful the system to send a trigger when the car model belongs to the suspicious model class. Instead of classifying the car into one of the 196 car make/type/year classes of the Stanford dataset which achieves poor performance, we instead intend to create a binary classifier that will categorize a car as either suspicious or non-suspicious. A binary classifier may be more effective in our case since trying to identify all the car models would be quite challenging. BPTI has furnished a roster of suspicious cars which will serve as the foundation for our algorithm. Regarding the Car label detection and recognition module, we are going to enhance the current methodology by refining our algorithm and expanding its capabilities to accommodate the recognition of additional license plate formats. Moreover, a docker environment is going to be developed for the integration of the algorithm regarding Pilot's purposes. Last but not least, we have communicated with our partner BPTI, and we are actively seeking a higher-quality video to evaluate the Car Visual Recognition Tool using superior image resolution, as the tool is intended for high quality close-distance car images in a controlled environment. We expect such a high-quality video to enhance the results of the algorithm and yield optimal results.

3.6. Cryptocurrency

3.6.1. Indicator Description

The various indicators established for different crypto assets are defined. These indicators have been selected based on the list of definitions concerning corruption risk indicators (detailed in Deliverable D2.2).

Crypto Address Property

This is a binary indicator that determines whether the selected address has available properties or not.

Crypto Transfer Amount

This is a numerical indicator that calculates the total number of transactions conducted by the requested wallet.

Crypto Transfer Currency

This is a categorical indicator that returns the currency type associated with a known wallet.

Crypto Transfer Balance

This is a numerical indicator that returns the total economic balance of a selected wallet, considering all money sent and received.

Crypto Transfer Input Transactions

This indicator is returned in a list format and provides information on each of the incoming transactions involving the requested wallet.

Crypto Transfer Output Transactions

This indicator is also returned in a list format and provides information on each of the outgoing transactions involving the requested wallet.

Crypto Account Label

This is a categorical indicator that returns the categorical value of the Label associated with a requested Account.

Crypto Account Prediction

This is a categorical indicator responsible for making a prediction about the selected wallet, returning the Account to which it belongs.

Crypto Account Of

This is a binary indicator that determines whether an Account belongs to a selected Label or not.

3.6.2. Related Use Case

We align with UC2 indicators (see more details in Deliverable D3.1) as defined in the FALCON project by conducting a comprehensive analysis of cryptocurrency addresses. Through the construction of address transaction graphs, we analyse the economic values associated with these addresses, identifying behavioural patterns that could be linked to illicit activities such as corruption, money laundering, or sanction evasion. Additionally, we group these addresses based on detected behaviours and predefined ground truth data, as well as categorizing them into specific labels. This approach facilitates the identification and tracking of assets that may be under the control of kleptocrats, oligarchs, or other sanctioned individuals.

3.6.3. Datasets

The entire Bitcoin blockchain data until the block 830,000 are downloaded, i.e., all the transactions until February 11th, 2024 (more than 900M transactions). In addition, to have more information about real- world entities, labelled (tagged) addresses are gathered from multiple reliable sources, such as WalletExplorer[10] and the tagpacks provided by Graphsense[11]. Indeed, these sources allowed us to gather more than 38M addresses of almost 400 entities labelled as Exchanges, Gambling, Marketplaces, Mining Pools, Mixers, Services, Trading platforms, eWallet, Ransomware, Sextortion, and Extremist.

3.6.4. Methodology

This methodology defines the approach to analyse the impact of sanctions on the cryptocurrency ecosystem, specifically within the Bitcoin (BTC) transaction network. The key steps of the methodology are outlined as follows:

Starting Point

This methodology begins by identifying BTC addresses included in the sanctions list (SDN list). These addresses serve as the starting point for the investigation.

Construction of the Address-Transaction Graph

An address-transaction graph is constructed using information available in the BTC blockchain. The elements of the graph are described as follows:

- **Nodes**: The nodes of the graph represent BTC addresses and transactions.
- **Directed Edges**: Directed edges connect addresses to incoming transactions and transactions to outgoing addresses, thus representing the flow of BTC.

[10] https://www.walletexplorer.com/

[11] https://graphsense.info/

- **Additional Information**: Edges may incorporate additional information such as the amount of BTC, fees, and timestamps.

Definition of the n-Step Graph

The methodology defines an n-step graph for a sanctioned address X1X1, which includes all paths originating from X1X1involving a maximum of n transactions. The maximum path length in this graph is *2n*.

Proposed Analysis

Two types of analysis are proposed:

1. **Flow Analysis**:
    - This analysis introduces a temporal aspect into the address-transaction graph.
    - Multiple 1-step graphs are created for each address of a sanctioned entity, considering four different time ranges:
        a) Transactions prior to the sanction (pre-sanction).
        b) Transactions within 7 days post-sanction (7 post-sanction).
        c) Transactions within 30 days post-sanction (30 post-sanction).
        d) All activities up to February 11th, 2024 (up-to-date).
    - **Extracted Metrics**: Metrics such as the number of input and output transactions, the overall balance of the entity after each time range, and the amount in USD of money sent and received (using the BTC/USD value on the transaction date) are extracted.
    - **Data Aggregation**: If an entity possesses multiple sanctioned addresses, the metrics from all its addresses are aggregated to provide a comprehensive view of the entity's behaviour.
2. **Behavioural Analysis**:
    - This analysis focuses on a single address-transaction graph for each entity, covering data from immediately after the sanctions until the end of the dataset (up-to-date).
    - **Enrichment with Real-World Data**: The graph is enriched with real-world entity information, using labels gathered from external sources.

- **Identification of Relationships**: This approach enables the identification of whether the sanctioned entity engages in transactions with other known entities that could be involved in illicit activities (e.g., other sanctioned entities, ransomware, etc.), or if it attempts strategies such as money laundering, on-ramps and off-ramps operations, or fundraising campaigns.
- **1-Step and 2-Step Analysis**: Both 1-step and 2-step analyses are conducted. The 1-step analysis provides information on the entity's direct relationships, while the 2-step analysis includes transactions reached in two steps, offering a deeper insight into the entity's strategy.

3.6.5. Experimental Results

This section presents the results obtained from the two proposed analyses. First, the results of the analysis of transactions and money flow of the entities before and after the sanctions are described. Then, the relationships that sanctioned entities have maintained with other known types of entities are detailed. Finally, key observations are discussed, and the study's limitations are identified.

Flow Analysis

The analysis revealed that only half of the sanctioned entities were effectively discouraged from engaging in transactions post-sanctions. Specifically, out of all the entities analysed, only 21 stopped receiving money, and 25 stopped sending funds. Despite these sanctions, some entities (7) continued moving funds within 7 days after the sanction. The distribution of sanctioned entities' transactional behaviour over different post-sanction intervals shows that even in the long term, a significant number of entities continued to receive and send funds. As time progressed, the persistence of these transactions became evident, highlighting the resilience of certain entities to sanctions.

The BTC (Bitcoin) balance analysis before and after the sanctions revealed a general trend of maintaining minimal balances in sanctioned addresses. Most entities did not move large amounts of money, preferring to keep small balances. Interestingly, the number of entities with a zero-balance decreased over time, while those with balances ranging from greater than 0 to up to 0.1 BTC increased. Additionally, only a small number of entities with a balance of 50 BTC or more chose to adopt an off-ramp strategy, withdrawing significant amounts to external wallets or other means. This pattern indicates a cautious approach by many sanctioned entities, perhaps to avoid detection or further penalties.

In terms of the volume of transactions and funds moved, cybercrime-related violations accounted for the highest number of transactions and USD volume before sanctions

were imposed, with over 150,000 transactions and approximately 8.3 billion USD moved. Although post-sanctions activity decreased, some transactions persisted, showing resilience in certain sectors. In contrast, entities involved in drug trafficking and illicit activities in North Korea showed significant pre-sanction transaction volumes but experienced notable reductions after the sanctions were enforced.

Lastly, entities associated with more severe violations, such as terrorism and arms proliferation, were largely deterred by the sanctions. These entities conducted only minimal transactions post-sanctions, with negligible fund movements, reflecting the effectiveness of the imposed restrictions.

Behavioural Analysis

The behavioural analysis focused on studying the patterns of behaviour of sanctioned entities through 1-step and 2-step address-transaction graphs. The 1-step analysis reached approximately 4,000 addresses, of which only a small fraction was related to known entities. Extending the analysis to 2 steps identified over 10 million addresses, but again, only a small portion was labelled.

This analysis revealed that although the 1-step approach provided more precise information, the 2-step analysis significantly enriched the investigation by identifying a greater number of behaviours and connections among entities. Most labelled addresses in both analyses belonged to exchanges, suggesting that these continue to be the primary intermediaries in transactions involving sanctioned entities.

Additionally, the 2-step analysis uncovered connections with addresses associated with serious crimes such as sextortion, ransomware, and extremism, illustrating the complexity and breadth of the networks in which these entities operate.

3.6.6. Future Work

This study has provided a detailed insight into the behaviour of sanctioned entities within the cryptocurrency ecosystem. However, several areas could be explored:

- **Exploration of Other Crypto Assets**: Expanding the analysis to include transactions in other crypto assets would provide a more comprehensive view of the strategies used by sanctioned entities to evade sanctions.
- **Development of Predictive Models**: New approaches to predictive models could be developed and tested based on the behavioural patterns identified. These models could help anticipate the actions of sanctioned entities following the imposition of sanctions, utilizing advanced machine learning techniques.
- **Long-Term Study**: Conducting a long-term study would be valuable to understand how sanctioned entities adapt over time. This would involve tracking

the emergence of new addresses and connections, as well as the evolution of their evasion methods.

- **Impact of New Regulations**: Investigating how new regulations and policies in different countries affect the behaviour of sanctioned entities in the cryptocurrency ecosystem would be insightful. Evaluating the effectiveness of these regulatory frameworks could provide key information for improving prevention strategies.

4. Anomaly Detection

Anomaly detection involves identifying patterns in data that deviate from expected behaviour. These deviations are known as anomalies, outliers, discordant observations, exceptions, aberrations, surprises, peculiarities, or contaminants, depending on the application domain. The terms anomalies and outliers are most frequently used and are often interchangeable in the context of anomaly detection. This technique is widely used in numerous applications, including credit card fraud detection, insurance and healthcare fraud detection, cybersecurity intrusion detection, fault detection in safety-critical systems, and military surveillance for enemy activities. The significance of anomaly detection lies in its ability to uncover crucial and actionable information across various fields.

Over time, numerous anomaly detection techniques have been developed by different research communities. Some of these techniques are tailored for specific application domains, while others are designed to be more general-purpose.

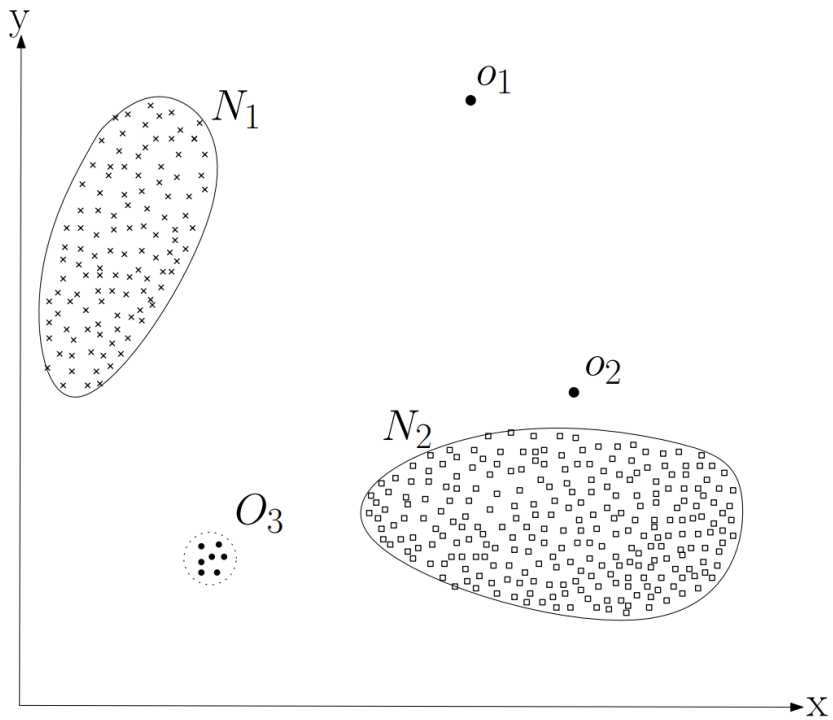

Figure 13. A simple example of anomalies (o1,o2 and O3) in a dataset [14]

At an abstract level, an anomaly is defined as a pattern that does not conform to expected normal behaviour. A straightforward anomaly detection approach, therefore, is to define a region representing normal behaviour and declare any observation in the data which does not belong to this normal region as an anomaly (Figure 13). However, several factors make this apparently simple approach very challenging. Most of the existing anomaly detection techniques solve a specific formulation of the problem. The

formulation is induced by various factors such as the nature of the data, availability of labelled data, type of anomalies to be detected, etc.

4.1. Types of Anomalies

An important aspect of an anomaly detection technique is the nature of the desired anomaly. Anomalies can be classified into the following three categories:

4.1.1. Point Anomalies

If an individual data instance can be considered as anomalous with respect to the rest of data, then the instance is termed a point anomaly. This is the simplest type of anomaly and is the focus of the majority of research on anomaly detection. As a real-life example, consider credit card fraud detection. Let the data set correspond to an individual's credit card transactions. For the sake of simplicity, let us assume that the data is defined using only one feature: the amount spent. A transaction for which the amount spent is very high compared to the normal range of expenditure for that person will be a point anomaly.

4.1.2. Contextual Anomalies

If a data instance is anomalous in a specific context (but not otherwise), then it is termed as a contextual anomaly (Figure 14). The notion of a context is induced by the structure in the data set and has to be specified as a part of the problem formulation. Each data instance is defined using the following two sets of attributes: contextual attributes and behavioural attributes.

The *contextual attributes* are used to determine the context (or neighbourhood) for that instance. For example, in spatial data sets, the longitude and latitude of a location are the contextual attributes. In time series data, time is a contextual attribute which determines the position of an instance on the entire sequence.

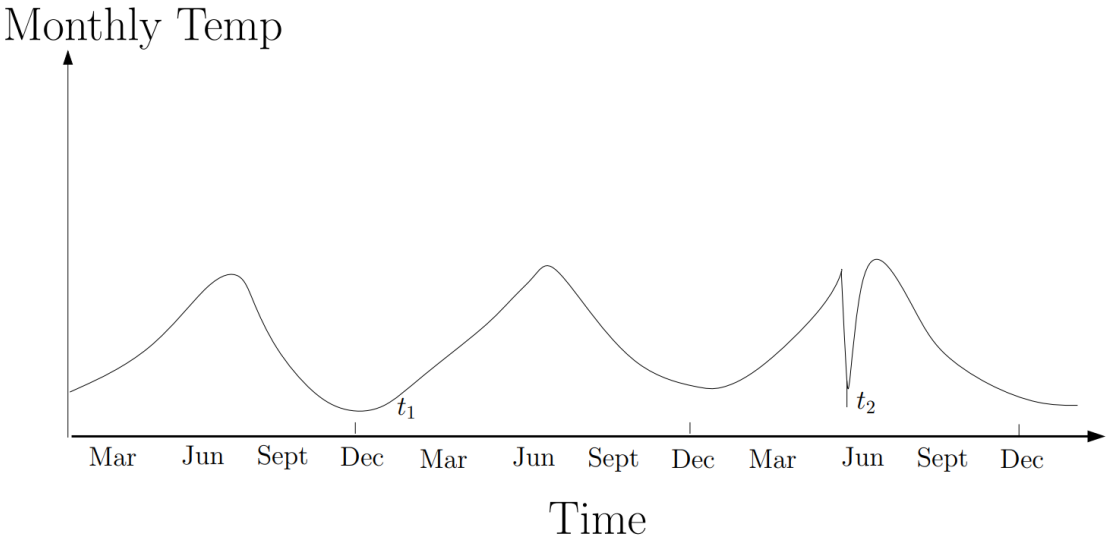

Figure 14. Contextual anomaly in a temperature time-series [14]

The *behavioural attributes* define the non-contextual characteristics of an instance. For example, in a spatial data set describing the average rainfall of the entire world, the amount of rainfall at any location is a behavioural attribute.

The anomalous behaviour is determined using the values for the behavioural attributes within a specific context. A data instance might be a contextual anomaly in a given context, but an identical data instance (in terms of behavioural attributes) could be considered normal in a different context. This property is key in identifying contextual and behavioural attributes for a contextual anomaly detection technique.

4.1.3. Collective Anomalies

If a collection of related data instances (sequential, spatial or graph data) is anomalous with respect to the entire data set, it is termed as a collective anomaly. The individual data instances in a collective anomaly may not be anomalies by themselves, but their occurrence together as a collection is anomalous.

4.2. Data Labels

The labels assigned to a data instance indicate whether it is normal or anomalous. It's important to highlight that obtaining labelled data that is both accurate and representative of all types of behaviours can be very costly. Labelling usually requires a human expert to manually tag the data, which demands a significant amount of effort. Generally, acquiring a labelled set of anomalous data instances that encompass all possible types of anomalies is more challenging than obtaining labels for normal behaviour. Furthermore, anomalous behaviour is often dynamic; new types of anomalies can emerge for which there is no labelled training data.

Anomaly detection techniques can function in one of three modes, depending on the availability of labels.

4.2.1. Supervised Anomaly Detection

Techniques trained in a supervised manner assume that a training dataset with labelled instances for both normal and anomalous classes is available. The typical method involves creating a predictive model to distinguish between normal and anomalous classes. Any new data instance is then evaluated against this model to classify it.

4.2.2. Semi-Supervised Anomaly Detection

Techniques operating in a semi-supervised mode assume that the training data contains labelled instances only for the normal class. Because they do not require labels for the anomaly class, these techniques are more broadly applicable than supervised ones.

Typically, these techniques build a model based on normal behaviour and use it to detect anomalies in the test data.

4.2.3. Unsupervised Anomaly Detection

Techniques operating in an unsupervised mode do not require training data, making them highly versatile. These methods assume that normal instances are much more common than anomalies in the test data. If this assumption is incorrect, these techniques may produce a high rate of false alarms.

Many semi-supervised techniques can be adapted for unsupervised use by treating a sample of the unlabelled dataset as training data. This adaptation relies on the assumption that the test data contains very few anomalies and that the model developed during training is resilient to these anomalies.

4.3. Applications of Anomaly Detection

In this section, we discuss some of the most prominent real-world applications of anomaly detection. Project FALCON will build on these applications and use cases, so as to design an effective anomaly detection mechanism.

4.3.1. Intrusion Detection

Intrusion detection refers to the detection of malicious activity (break-ins, penetrations, and other forms of computer abuse) in a computer related system. These malicious activities or intrusions are interesting from a computer security perspective. An intrusion is different from the normal behaviour of the system, and hence anomaly detection techniques are applicable in intrusion detection domain. The key challenge for anomaly detection in this domain is the huge volume of data. The anomaly detection techniques need to be computationally efficient to handle these large sized inputs. Moreover, the data typically comes in a streaming fashion, thereby requiring online analysis. Another issue that arises because of the large sized input is the false alarm rate. Since the data amounts to millions of data objects, a few percent of false alarms can make analysis overwhelming for an analyst. Labelled data corresponding to normal behaviour is usually available, while labels for intrusions are not. Thus, semi-supervised and unsupervised anomaly detection techniques are preferred in this domain.

4.3.2. Fraud Detection

Fraud detection refers to the detection of criminal activities occurring in commercial organizations such as banks, credit card companies, insurance agencies, cell phone companies, stock market, and so on. The malicious users might be the actual customers of the organization or might be posing as customers (also known as identity theft). The

fraud occurs when these users consume the resources provided by the organization in an unauthorized way.

4.3.3. Medical and Public Health Anomaly Detection

Anomaly detection in the medical and public health domains typically works with patient records. The data can have anomalies due to several reasons, such as abnormal patient condition, instrumentation errors, or recording errors. Anomaly detection is a very critical problem in this domain and requires a high degree of accuracy.

4.3.4. Image Processing

Anomaly detection techniques dealing with images are either interested in any changes in an image over time (motion detection) or in regions that appear abnormal on the static image. The anomalies are caused by motion, or insertion of a foreign object, or instrumentation errors. The data has spatial as well as temporal characteristics. Each data point has a few continuous attributes such as colour, lightness, texture, and so on. The interesting anomalies are either anomalous points or regions in the images (point and contextual anomalies).

One of the key challenges in this domain is the large size of the input. When dealing with video data, online anomaly detection techniques are required.

4.4. Methods

In this section, we provide a broad overview of the different anomaly detection methods available.

4.4.1. Statistical

Statistical methods for anomaly detection utilize statistical techniques to identify data points that significantly deviate from the expected distribution within a dataset. These methods can be broadly divided into parametric and non-parametric approaches.

Parametric methods often assume that the data follows a specific distribution, such as the Gaussian (normal) distribution. In this approach, anomalies are identified as data points that fall outside a certain number of standard deviations from the mean. Alternatively, the t-distribution can be used, particularly in cases involving smaller sample sizes or data with heavier tails. Another parametric approach involves using mixture models, where a combination of several probability distributions models different clusters within the data. Anomalies are detected as data points that do not fit well into any of the established clusters, with Gaussian Mixture Models (GMM) being a common example.

Non-parametric methods do not assume a specific distribution for the data. Histogram-based methods involve dividing the data into bins and analysing the frequency of data

points within each bin. Data points in bins with significantly low frequencies are considered anomalies. Kernel Density Estimation (KDE) [14] is another non-parametric approach that estimates the probability density function of the data. Anomalies are those data points that lie in regions with low estimated density.

Other statistical methods for anomaly detection include the box plot method, the z-score method, Grubb's test and Principal Component Analysis (PCA) [15].

4.4.2. Density

Density anomaly detection methods are techniques used to identify patterns in data that deviate from expected behaviour based on the density of data points in a given space.

Density-Based Spatial Clustering of Applications with Noise (DBSCAN) [16], which is a clustering algorithm that groups together points that are closely packed, while marking points that lie alone in low-density regions as outliers. It relies on two parameters: epsilon (ε), which defines the radius of a neighbourhood around a point, and the minimum number of points required to form a dense region.

Local Outlier Factor (LOF) [17] measures the local density deviation of a data point with respect to its neighbours. The anomaly score is based on the ratio of the local density of the point to the local densities of its neighbours. A point is considered an outlier if its local density is significantly lower than that of its neighbours.

Isolation Forest [18] isolates observations by randomly selecting a feature and then randomly selecting a split value between the maximum and minimum values of the selected feature. The process is repeated recursively, creating a tree structure. Anomalies are identified as points that require fewer splits to be isolated, as they are less frequent and lie in sparse regions of the data.

k-Nearest Neighbours (k-NN) [19] based anomaly detection uses the distance to the k-th nearest neighbour as the anomaly score. Points with a high k-th nearest neighbour distance are considered anomalies since they are far from their neighbours, indicating low-density regions.

4.4.3. Cluster-based

Cluster-based anomaly detection methods identify anomalies by analysing how data points are grouped into clusters. The fundamental idea is that normal data points belong to large, dense clusters, while anomalies are located in small or sparse clusters, or do not belong to any cluster.

K-means clustering [20] partitions the data into a predefined number of clusters, assigning each point to the nearest cluster centroid. Anomalies are identified as points

that are far from their cluster centroids, indicating that they do not fit well within the cluster structure.

Expectation-Maximization (EM) [21] clustering uses a probabilistic approach to model the data as a mixture of Gaussian distributions. Points with low probabilities of belonging to any of the Gaussian components are considered anomalies.

Hierarchical clustering [22] builds a tree of clusters by either iteratively merging smaller clusters into larger ones (agglomerative) or splitting larger clusters into smaller ones (divisive). Anomalies are detected as points that form small clusters at the leaves of the tree or as outliers that do not merge into larger clusters until late in the process.

4.4.4. Deep Neural Networks

In [23] a hierarchical taxonomy of deep learning anomaly detection methods is introduced, where the methods are classified into three main categories, as in Figure 15.

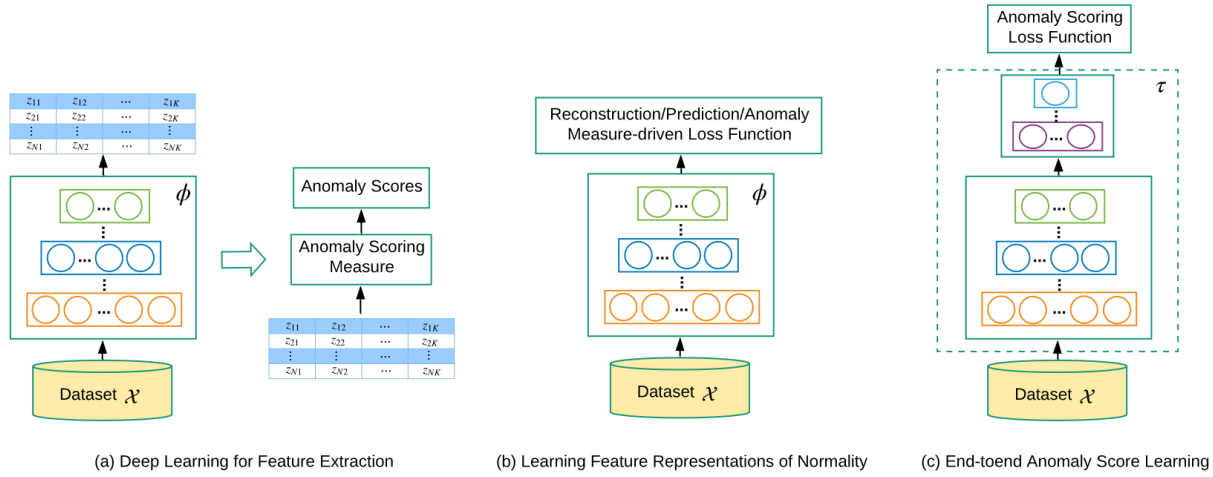

Figure 15. Conceptual Frameworks of Three Main Deep Anomaly Detection Approaches [23]

The first category is Deep Learning for Feature Extraction. This category of methods focuses on using deep learning to derive low-dimensional feature representations from high-dimensional or non-linearly separable data for subsequent anomaly detection. The processes of feature extraction and anomaly scoring are completely separate and independent. Consequently, the deep learning components serve solely for dimensionality reduction.

The second category is Learning Feature Representations of Normality, with methods that generally fall into two groups: i) generic feature learning and ii) anomaly measure-dependent feature learning.

In generic feature learning the algorithms learn the representations of data instances by optimizing a generic feature learning objective function that is not primarily designed

for anomaly detection, but the learned representations can still empower the anomaly detection since they are forced to capture some key underlying data regularities. The deep learning models most often used are Autoencoders, Variational Autoencoders [24] and Generative Adversarial Networks [25]. In anomaly measure-dependent feature learning the algorithms aim at learning feature representations that are specifically optimized for one particular existing anomaly measure. Here more standard machine learning methods are used such as SVMs and GMMs.

The third category is End-to-End Anomaly Score Learning. This approach focuses on directly learning scalar anomaly scores through an end-to-end process. Unlike methods that rely on pre-existing anomaly measures for feature learning, this approach uses a neural network to learn the anomaly scores independently. This typically necessitates the development of new loss functions to train the anomaly scoring network effectively.

5. Trends Analysis

The FALCON project aims to strengthen the EU's capacity to understand and combat corruption. Within this context, trend analysis, which involves examining data to identify patterns, shifts, and emerging phenomena over time, aims to contribute to the identification of potential trends that could be linked to corruption phenomena. Towards this direction, the FALCON's trends detection tool will analyse data from heterogeneous sources, combining temporal and spatial information (if possible), to gain deeper insights and provide reliable detections. The trends detection component will exploit the obtained intelligence from tasks 4.1 – 4.3. Data of interest for the identification of trends include, for instance, transactions with cryptocurrencies and public procurement. Cryptocurrencies, such as Bitcoin [26] and Ethereum [27], characterized by their decentralized nature and anonymity, have transformed the financial landscape, offering innovative opportunities for investment and transactions. However, these features also make it a prime target for fraudulent activities. To this end, the initial design of the trends detection tool focuses on the analysis of cryptocurrency transactions relevant to FALCON's Use Case 2 "Tracing of sanction circumvention schemes and assets owned by 'kleptocrats'/oligarchs" to detect trends that could potentially be related to corruption schemes, exploiting the transparency that public blockchains offer to enhance the detection of corruption.

Next, Section 5.1 presents a literature review of trends analysis on cryptocurrency transactions related to illicit activities; unsupervised (Section 5.1.1), supervised (Section 5.1.2), and hybrid methods (Section 5.1.3) are reported. Section 5.2 provides an overview of the initial design of FALCON's trends detection tool. Finally, this section concludes with a summary (Section 5.3).

5.1. Related Work

This section reviews the latest and most advanced techniques in the field of trend analysis of cryptocurrency transactions to identify illicit activities. It includes unsupervised, supervised AI-based methods using labelled datasets, and hybrid methods showcasing how they contribute to a deeper understanding of fraudulent patterns and enhance the ability to prevent and mitigate crime.

5.1.1. Supervised Techniques

In the realm of cryptocurrency, the missing and scarce availability of labelled data for fraud detection has led to a reliance on unsupervised learning techniques to identify and analyse fraudulent activities. More in detail various methodologies exist to detect illicit activities categorized by the used method (partitional, graph-based, density-based, probabilistic) [28], the blockchain layer (data layer, network layer, Incentive layer, contract layer), and the type of fraud that is related [29].

Advanced partitioning techniques are employed across various studies to analyse transaction patterns and behaviours at different layers of the blockchain network. For instance, [30] applied trimmed K-means clustering to detect anomalies in Bitcoin transactions, effectively identifying financial fraud and associated activities such as money laundering. Similarly, [31] used Affinity Propagation and K-medoids to label Ethereum smart contracts and identify malicious ones. In [32] an enhanced K-means clustering algorithm with Dynamic Time Warping (DTW) was used to detect double-spending attacks in blockchain nodes where fraudulent users spent the same digital currency more than once to gain financial advantages and facilitate illegal operations, posing a significant threat to the blockchain.

Graph-based approaches are also used to detect suspicious users or transactions by representing users and transactions as vertices and edges, respectively. In [33] key properties of the Bitcoin network graph were explored, focusing on classical graph properties such as densification, distance analysis, degree distribution and clustering coefficient. Densification examines how the network becomes denser over time, while distance analysis assesses the reachability within the network. Degree distribution identifies patterns in node connections, and the clustering coefficient measures the tendency of nodes to form tightly knit groups. Moreover, [34] presents a method to detect money laundering in Bitcoin transactions. The authors introduce Guilty Walker, a tool that uses random walks on the Bitcoin transaction graph to calculate features based on the distance to known illicit nodes. These new features, when combined with existing transaction-specific features, enhance machine learning models' ability to identify illicit transactions, particularly during events like black market shutdowns.

In crypto transactions related to illicit activities, temporal aspects are crucial as they reveal patterns over time, helping to identify anomalies that static data might miss. Incorporating time-series analysis enhances the detection of suspicious activities, capturing the dynamic nature of fraud. For instance, [35] demonstrated the effectiveness of combining temporal and graph-based features using K-means clustering, Mahalanobis distance, and Unsupervised Support Vector Machine (SVM) to achieve this goal. Similarly, [36] focused on detecting anomalies in Bitcoin transactions by extracting numerical features through sliding windows and calculating anomaly scores to pinpoint significant events, such as changes in investment rules or scheme collapses. [37] employed rolling window aggregation to extract features over various time frames, from seconds to 90 days, tailoring anomaly detection models to individual addresses for timely and accurate identification of suspicious activities. In [38], this approach was extended to Ethereum transactions, using various algorithms to emphasize temporal dependencies and multivariate time series analysis. Key features analysed include payment amounts, destination addresses, gas limits, and gas prices, all contributing to high accuracy and rapid response in detecting fraud.

5.1.2. Labelled Datasets and Supervised Techniques

While labelled data for fraudulent cryptocurrency transactions is sparse and rare, some valuable datasets exist that enable supervised learning approaches. One such dataset is the "Elliptic Dataset,"[12] from Kaggle, a transaction graph collected from the Bitcoin blockchain, which contains 203,769 nodes and 234,355 edges, with 2% (4,545) labelled as illicit and 21% (42,019) as licit, leaving the rest unlabelled. It features 94 local transaction attributes (e.g., time step, transaction fee) and 72 aggregated attributes derived from one-hop neighbours, including metrics like maximum, minimum, standard deviation, and correlation coefficients. In [39] a Random Forest (RF) algorithm was applied to the "Elliptic Dataset" to effectively identify fraudulent transactions by training on known examples of both licit and illicit activities. Building on this research, [40] demonstrated that active learning could also effectively detect money laundering in Bitcoin transactions, even with minimal labelled data. The proposed active learning method achieved performance comparable to fully supervised models while using only 5% of the labels.

Additionally, the Blockchain Anomaly and Detection Benchmark (BADB-13)[13] dataset, introduced by [41], is another important resource in this context. The BADB-13 dataset contains 13 types of Bitcoin addresses, 5 categories of indicators with 148 features, and 544,462 labelled instances of various types of fraudulent activities, including Ponzi schemes, phishing, and scam transactions. It includes features such as transaction times, amounts, recipient addresses, and other transaction metadata, enabling comprehensive analysis and detection of fraudulent activities. In [42] goodness-of-fit tests (Kolmogorov-Smirnov, Anderson-Darling, and Cramér-von Mises) were utilized to analyse time intervals between Bitcoin transactions, effectively identifying suspicious addresses from the BABD-13 dataset.

Lastly, in [43] the Ethereum Fraud Detection[14] dataset from Kaggle was used, which comprises 9841 Ethereum accounts, with 7662 marked as non-fraudulent and 2179 as fraudulent. The dataset includes 49 features, such as the sender's address, receiver's address, and transaction value to detect fraudulent transactions using RF, SVM, and K-Nearest Neighbors (KNN) in 10-fold cross validation set.

5.1.3. Hybrid Techniques

To develop a robust method for detecting fraudulent activities on the blockchain, a hybrid approach integrating multiple anomaly detection techniques can be employed. These techniques can be applied sequentially (cascade methods), in parallel (majority

[12] https://www.kaggle.com/datasets/ellipticco/elliptic-data-set

[13] https://www.kaggle.com/datasets/lemonx/babd13

[14] https://www.kaggle.com/datasets/vagifa/ethereum-frauddetection-dataset

vote methods), or in a semi-supervised setup to enhance the accuracy and reliability of the detection process.

The cascade method involves the sequential application of multiple models to refine the detection results progressively. An application of this methodology was demonstrated in [44], where the focus was on anomaly detection specific to the Bitcoin transaction network. They aimed to detect suspicious users and transactions, where anomalous behaviour serves as a proxy for suspicious activity. Initially, a clustering algorithm such as k-means was used to group the data into clusters. This step helped in organizing the data into meaningful subsets. Following this, the Local Outlier Factor (LOF) technique was applied to these clusters to identify outliers based on local density variations within the clusters. The LOF method was effective in detecting anomalies by comparing the local density of a data point to that of its neighbours, making it easier to spot deviations from norm users and transactions that could indicate fraudulent activities for further analysis. This sequential application of clustering followed by outlier detection ensured a more accurate anomaly detection process.

As described in [45], the majority vote method involves executing multiple anomaly detection models in parallel, where each model independently analyses the data and outputs its detection results. The final decision on whether a data point is anomalous is made based on a majority vote mechanism. This means that if the majority of models flag a data point as an anomaly, it is considered fraudulent. By leveraging the collective intelligence of multiple models, this method increases the robustness and reliability of the detection process. The majority vote approach reduces the risk of false positives and negatives, as the combined decision is less likely to be swayed by the limitations or biases of a single model, thereby enhancing the overall detection performance.

The semi-supervised method combines elements of both supervised and unsupervised learning to improve detection accuracy; this methodology was applied in [20]. In particular, at first, data points that are known to be similar, such as those involved in Ponzi schemes, were clustered using a normalized Levenshtein distance. This distance metric measures the similarity between sequences, making it suitable for clustering similar contracts. After clustering, a Gradient Boosting Machine (GBM) was used to classify the clusters. GBM is a powerful ensemble learning technique that builds a series of decision trees sequentially, where each tree corrects the errors of the previous ones. This method effectively distinguished between Ponzi and non-Ponzi contracts, providing a nuanced detection capability that adapts to the complexities of blockchain transactions.

5.2. Trends Detection Design

This section provides an overview of the initial design of FALCON's trends detection tool. The initial version of the tool will focus on the analysis of cryptocurrency transactions to detect trends that could potentially be related to corruption phenomena.

The main input will be formulated from the outcomes of tasks 4.1 – 4.3; in particular:

- Preliminary data collection, preprocessing, and basic exploratory analysis related to cryptocurrency transactions. This data can include enriched datasets with additional features derived from raw transaction data, such as transaction frequency and volume patterns.
- Extracted indicators of corruption related to cryptocurrency transactions

The foreseen processes to extract useful intelligence for the identification of trends will include the use of AI-based techniques focusing on the exploitation of temporal features from the transactions' history of addresses/wallets of interest. The analysis of temporal features may rely on time series analysis techniques such as point-by-point Poisson models and change point analysis or clustering methods to refine the detection of trends potentially linked to corruption.

Finally, the output of the initial version of the trends detection tool is foreseen to include identified trends in cryptocurrency transactions of interest that may correlate with corruption indices. These trends could include unusual transaction volumes, high-frequency trading between certain addresses, or significant deviations from expected behaviour. The outcome of FALCON's trends detection component will be used as input to WP5 analysis and visualization tools. Figure 16 depicts the foreseen workflow in the initial version of the trend's detection tool.

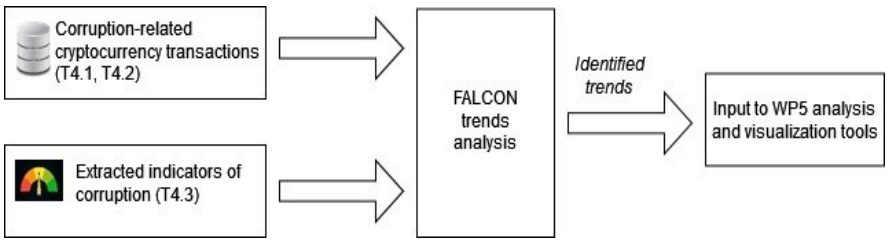

Figure 16. Foreseen workflow of the initial version of FALCON's trends detection tool

Regarding the backend implementation, the communication of the trends detection tool with the other components will be supported via a message broker. The module will consume messages of interest published on the message broker, then it will process the requested data from FALCON's knowledge base, and finally, it will produce a message upon the completion of the relevant analysis task. Figure 17 depicts a description of the

foreseen pipeline in a sequence diagram; this is also included in deliverable 3.2 “FALCON Framework Architecture”.

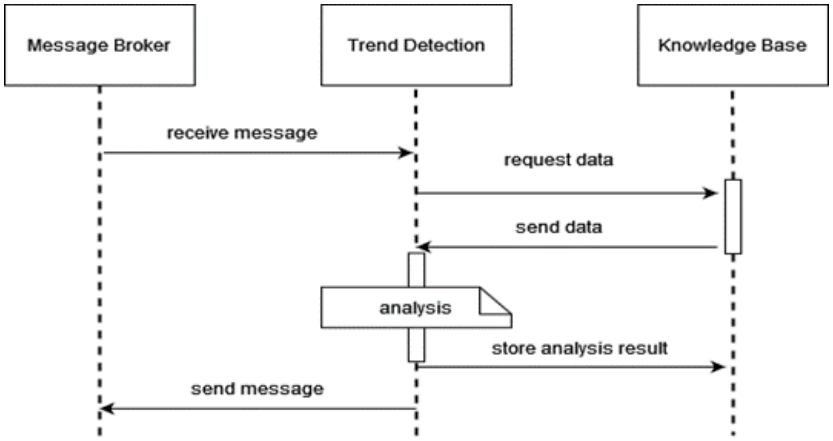

Figure 17 Sequence diagram for FALCON’s trends detection tool

5.3. Summary

This section presented the initial design for the development of FALCON’s trends detection tool. Specifically, at first, the focus is foreseen to be on the analysis of cryptocurrency transactions to identify trends that could be related to corruption. Moreover, related work was detailed about the use of AI-based techniques for the detection of trends in fraudulent cryptocurrency transactions. In the next iterations, the development of the tool as well as the results of the analysis will be reported.

6. Summary and Conclusions

This document has provided an extensive overview of the tools developed for the first version of the Corruption Data Acquisition and Analysis Toolset within the FALCON project. Designed to facilitate the extraction of key indicators from various underlying data sources, this toolset enables effective anomaly detection and trend analysis. Its primary objective is to enhance our understanding of corruption dynamics by offering robust analytical capabilities that allow users to identify irregular patterns and potential red flags in the data, crucial for timely interventions.

Additionally, the document systematically catalogues data sources related to corruption, detailing the specific information utilized for extracting indicators and creating visualizations. This organization aims to improve the clarity and accessibility of data, thereby facilitating more effective analysis and interpretation.

The report further outlines the mechanisms for managing access to datasets within the FALCON framework, including necessary adapters for heterogeneous data sources and a dataflow manager that supports data injection and modelling tailored to corruption crime descriptions. Specialized indicator extraction tools focusing on border corruption, public procurement risks, cryptocurrency transactions and social data networks relevant to all FALCON use cases are also highlighted.

A State-of-the-Art review is included, presenting methodologies for anomaly detection and trend analysis that will be integrated into future iterations of the FALCON tools. This overview encompasses innovative techniques such as machine learning algorithms and statistical methods aimed at enhancing the accuracy and efficiency of anomaly detection in extensive datasets.

In conclusion, this document establishes a foundational step for the FALCON project, laying the groundwork for ongoing development and refinement of tools that support anti-corruption efforts through advanced technological solutions. As the project progresses, these tools and methodologies will evolve to remain aligned with the project's goals and stakeholder expectations.

7. References

[1] Bosisio, Antonio, and Maria Jofre. 2022. "Investigating High-Risk Firms." European Law Enforcement Research Bulletin 22 (November). https://bulletin.cepol.europa.eu/index.php/bulletin/article/view/547.

[2] Bosisio, Carbone, Jofre, Riccardi, Guastamacchia, Antonio, Carlotta, Maria, Michele, Stefano. 2021. "Project DATACROS - Developing a Tool to Assess Corruption Risk Factors in Firms' Ownership Structure." https://www.transcrime.it/wp-content/uploads/2021/09/Datacros_report.pdf.

[3] Bosisio, Nicolazzo, Riccardi, Antonio, Giovanni, Michele. 2021. "I Cambi Di Proprietà Delle Aziende Italiane Durante l'emergenza Covid-19: Trend e Fattori Di Rischio." https://www.transcrime.it/wp-content/uploads/2021/05/Ownership-changes-report-1.pdf.

[4] Jofre, Maria. 2022. "Network Analysis for Financial Crime Risk Assessment: The Case Study of the Gambling Division in Malta." Global Crime 23 (2): 148–70. https://doi.org/10.1080/17440572.2022.2077330

[5] Riccardi, Michele. 2021. Money Laundering Blacklists. 1st ed. London: Routledge. https://doi.org/10.4324/9781003212867.

[6] Mineva, D., Kostova, T., Fazekas, M., Poltoratskaia, V. (2023). Bridges to Nowhere: State Capture and Corruption Risks in Fiscal Transfers and Public Procurement at the Local Level in Southeast Europe. Center for the Study of Democracy. Sofia, Bulgaria.
5

[7] Ortega Nieto, D., Fazekas, M., Vaz Mondo, B., Tóth, B., Braem Velasco, R. A. (2023). Governance Risk Assessment System (GRAS): Advanced Data Analytics for Detecting Fraud, Corruption, and Collusion in Public Expenditures (English). Equitable Growth, Finance and Institutions Insight. Washington, D.C.: World Bank Group.

[8] Mineva, D., Fazekas, M., Poltoratskaia, V. and Tsabala, K. (2023). Rolling Back State Capture in Southeast Europe. Implementing Effective Instruments for Asset Declaration and Politically Exposed Companies. Center for the Study of Democracy.

[9] Desislava Nikolova, Mihaly Fazekas, Bence Tóth, Viktoriia Poltoratskaia, Marc Schiffbauer, et al. (2023) Bulgaria; Country Economic Memorandum: A Path to High Income (English). Washington, D.C. : World Bank Group.

[10] OECD (2021), Countering Public Grant Fraud in Spain: Machine Learning for Assessing Risks and Targeting Control Activities, OECD Public Governance Reviews, Paris: OECD Publishing, https://doi.org/10.1787/0ea22484-en.

[11] Cocciolo, S., Samaddar, S. and Fazekas, M. 2023. Government Analytics Using Procurement Data. in Rogger, D. and Schuster, C. (editors) 2023. The Government

Analytics Handbook: Leveraging Data to Strengthen Public Administration. Washington, DC: World Bank, Chapter 12.
[12] Fazekas, Mihály, István János Tóth, and Lawrence Peter King. "An objective corruption risk index using public procurement data." European Journal on Criminal Policy and Research 22 (2016): 369-397
[13] Czibik, Á. and Fazekas, M (2021): Measuring regional quality of government: the public spending quality index based on government contracting data. Regional Studies. 55(8), pp. 1459-1472, 10.1080/00343404.2021.1902975
[14] Chen, Y. C. (2017). A tutorial on kernel density estimation and recent advances. *Biostatistics & Epidemiology*, *1*(1), 161-187.
[15] Maćkiewicz, A., & Ratajczak, W. (1993). Principal components analysis (PCA). *Computers & Geosciences*, *19*(3), 303-342.
[16] Ester, M., Kriegel, H. P., Sander, J., & Xu, X. (1996, August). A density-based algorithm for discovering clusters in large spatial databases with noise. In *kdd* (Vol. 96, No. 34, pp. 226-231).
[17] Breunig, M. M., Kriegel, H. P., Ng, R. T., & Sander, J. (2000, May). LOF: identifying density-based local outliers. In *Proceedings of the 2000 ACM SIGMOD international conference on Management of data* (pp. 93-104).
[18] Liu, F. T., Ting, K. M., & Zhou, Z. H. (2008, December). Isolation forest. In *2008 eighth ieee international conference on data mining* (pp. 413-422). IEEE.
[19] Peterson, L. E. (2009). K-nearest neighbor. *Scholarpedia*, *4*(2), 1883.
[20] Kanungo, T., Mount, D. M., Netanyahu, N. S., Piatko, C. D., Silverman, R., & Wu, A. Y. (2002). An efficient k-means clustering algorithm: Analysis and implementation. *IEEE transactions on pattern analysis and machine intelligence*, *24*(7), 881-892.
[21] Moon, T. K. (1996). The expectation-maximization algorithm. *IEEE Signal processing magazine*, *13*(6), 47-60.
[22] Nielsen, F., & Nielsen, F. (2016). Hierarchical clustering. *Introduction to HPC with MPI for Data Science*, 195-211.
[23] Pang, G., Shen, C., Cao, L., & Hengel, A. V. D. (2021). Deep learning for anomaly detection: A review. *ACM computing surveys (CSUR)*, *54*(2), 1-38.
[24] Doersch, C. (2016). Tutorial on variational autoencoders. *arXiv preprint arXiv:1606.05908*.
[25] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., ... & Bengio, Y. (2020). Generative adversarial networks. *Communications of the ACM*, *63*(11), 139-144.
[26] Nakamoto, S. (2008). Bitcoin. A peer-to-peer electronic cash system, 21260.
[27] Buterin, V. (2014). A next-generation smart contract and decentralized application platform. white paper, 3(37), 2-1

[28] Cholevas, C., Angeli, E., Sereti, Z., Mavrikos, E., & Tsekouras, G. E. (2024). Anomaly Detection in Blockchain Networks Using Unsupervised Learning: A Survey. Algorithms, 17(5), 201.
[29]Hassan, M. U., Rehmani, M. H., & Chen, J. (2022). Anomaly detection in blockchain networks: A comprehensive survey. IEEE Communications Surveys & Tutorials, 25(1), 289-318.
[30] Monamo, P., Marivate, V., & Twala, B. (2016). Unsupervised learning for robust Bitcoin fraud detection. In 2016 Information Security for South Africa (ISSA) (pp. 129-134). IEEE.
[31]Norvill, R., Pontiveros, B. B. F., State, R., Awan, I., & Cullen, A. (2017). Automated labeling of unknown contracts in ethereum. In 2017 26th International Conference on Computer Communication and Networks (ICCCN) (pp. 1-6). IEEE.
[32]Kumari, R. A. C. H. A. N. A., & Catherine, M. O. N. I. C. A. (2018). Anomaly detection in blockchain using clustering protocol. International Journal of Pure and Applied Mathematics, 118(20), 391-396.
[33] Di Francesco Maesa, D., Marino, A., & Ricci, L. (2018). Data-driven analysis of bitcoin properties: exploiting the users graph. International Journal of Data Science and Analytics, 6, 63-80
[34] Oliveira, C., Torres, J., Silva, M. I., Aparício, D., Ascensão, J. T., & Bizarro, P. (2021). GuiltyWalker: Distance to illicit nodes in the Bitcoin network. arXiv preprint arXiv:2102.05373.
[35] Chaudhari, D., Agarwal, R., & Shukla, S. K. (2021, December). Towards Malicious address identification in Bitcoin. In 2021 IEEE international conference on blockchain (Blockchain) (pp. 425-432). IEEE.
[36] Toyoda, K., Ohtsuki, T., & Mathiopoulos, P. T. (2018). Time series analysis for bitcoin transactions: The case of pirate@ 40's hyip scheme. In 2018 IEEE International Conference on Data Mining Workshops (ICDMW) (pp. 151-155). IEEE.
[37] Podgorelec, B., Turkanović, M., & Karakatič, S. (2019). A machine learning-based method for automated blockchain transaction signing including personalized anomaly detection. Sensors, 20(1), 147.
[38] Kaufman, E., & Iaremenko, A. (2022). Anomaly Detection for Fraud in Cryptocurrency Time Series. arXiv preprint arXiv:2207.11466.
[39] Weber, M., Domeniconi, G., Chen, J., Weidele, D. K. I., Bellei, C., Robinson, T., & Leiserson, C. E. (2019). Anti-money laundering in bitcoin: Experimenting with graph convolutional networks for financial forensics. arXiv preprint arXiv:1908.02591.
[40] Lorenz, J., Silva, M. I., Aparício, D., Ascensão, J. T., & Bizarro, P. (2020). Machine learning methods to detect money laundering in the bitcoin blockchain in the presence

of label scarcity. In Proceedings of the first ACM international conference on AI in finance (pp. 1-8).
[41] Xiang, Y., Lei, Y., Bao, D., Li, T., Yang, Q., Liu, W., ... & Choo, K. K. R. (2023). Babd: A bitcoin address behavior dataset for pattern analysis. IEEE Transactions on Information Forensics and Security.
[42] Maheshwari, R., VA, S. P., Shobha, G., Shetty, J., Chala, A., & Watanuki, H. (2023). Illicit activity detection in bitcoin transactions using timeseries analysis. International Journal of Advanced Computer Science and Applications, 14(3).
[43] Raheem, M., & Abubacker, N. F. (2023, December). Ethereum Fraud Detection Using Machine Learning. In 2023 IEEE 21st Student Conference on Research and Development (SCOReD) (pp. 219-224). IEEE.
[44] Pham, T., & Lee, S. (2016). Anomaly detection in bitcoin network using unsupervised learning methods. arXiv preprint arXiv:1611.03941.
[45] Kaufman, E., & Iaremenko, A. (2022). Anomaly Detection for Fraud in Cryptocurrency Time Series. arXiv preprint arXiv:2207.11466.