Techniques and Frameworks for Classification of Open-Source Information

Călin Ioan JULAN, Alexandru IVANA, and Cristian-Alexandru VINATORU

***Abstract*—his paper reviews several modern frameworks and techniques for classification of open-source information. For a better understanding of how these technologies work, basic principles of functionality are explained.**

***Index Terms*—SVM-Support Vector Machine.**

I. INTRODUCTION

Open-source information, such as data, documents, and media, is widely available and can be a valuable resource for various purposes. However, the vast amount of information available can make it challenging to find and use relevant information efficiently. Classification, the process of organizing information into categories, can help to overcome this challenge by enabling the identification and retrieval of specific types of information [1].

There are various techniques and frameworks that can be used for the classification of open-source information, including:

1. Manual classification: This involves manual evaluation of open-source information by a person, who labels it according to predefined categories.[2] This method is slow and can be subjective, but can be useful in cases where the amount of information is small or if a careful evaluation of the quality of the information is needed.
2. Automatic classification algorithms: These use machine learning models, such as logistic regression or decision trees, to automatically classify open-source information according to predefined categories. This can be done by training a classification model with labeled examples of data, so that it can predict labels for new sets of data. [3]Automatic classification algorithms are faster and can process large amounts of information, but may be less accurate than manual evaluation.
3. Semi-automatic classification algorithms [5]: These combine elements of both of the above methods, allowing people to automatically label a large amount of information, but also allowing for manual evaluation of a portion of it. Examples of semi-automatic classification algorithms include active learning and collaborative filtering. This method can be useful when a combination of accuracy and speed is desired.
4. Content analysis: This involves using natural language processing algorithms, such as support vector machines or naive Bayes classifiers [4], to analyze the content of open-source information and classify it according to theme, sentiment, or another defined characteristic. This method can be useful for analyzing large volumes of text-based information and getting an overview of the sentiments or themes being discussed.
5. Data mining frameworks: Data mining frameworks, such as WEKA or Orange [6], are collections of algorithms and tools that can be used to discover patterns and trends in large amounts of data. These frameworks can be used to classify open-source information by identifying patterns and relationships within the data that can be used to categorize it.
6. Ontology-based approaches: Ontologies are structured representations of knowledge that can be used to classify information according to its meaning and relationships to other concepts. Ontology-based approaches to classification involve creating an ontology that reflects the categories or concepts relevant to the information being classified, and then using this ontology to categorize the information. Tools such as Protégé or TopBraid can be used to create and manipulate ontologies [6].

II. TYPES OF CLASSIFICATIONS

Manual classification of open-source information using frameworks refers to the process of categorizing and organizing data obtained from open-source intelligence (OSINT) using predefined frameworks or structures. This approach helps analysts to efficiently manage and analyze large volumes of information and extract valuable insights.

Here are some key steps involved in the manual classification of open-source information using frameworks [7]:

1. Define the Objective: Clearly articulate the purpose of the classification process. Determine the specific information requirements and goals, such as identifying potential security threats, monitoring online discussions, or gathering market intelligence.

C. I. JULAN is with the Military Technical Academy “Ferdinand I”, Bucharest, Romania (e-mail: julan_calin@yahoo.com).

A. IVANA is with the Military Technical Academy “Ferdinand I”, Bucharest, Romania.

C. VINATORU is with the Military Technical Academy “Ferdinand I”, Bucharest, Romania.

Digital Object Identifier 10.32754/JMT.2023.1.08

2. Select a Framework: Choose a classification framework that aligns with the objectives and types of information being analyzed. A framework provides a structured system for categorizing data, ensuring consistency and enabling easy retrieval. Examples of frameworks commonly used in intelligence analysis include the Analysis of Competing Hypotheses (ACH), the Intelligence Cycle, and the Joint Military Intelligence Analysis Process (JMIAP).
3. Create Categories: Within the chosen framework, establish categories or labels that will be used to classify the information. These categories should be mutually exclusive and collectively exhaustive, meaning that each piece of information can fit into only one category and all information falls under at least one category. The categories should reflect the specific needs of the analysis and allow for meaningful differentiation between different types of information.
4. Train Analysts: Provide training to analysts who will be responsible for manually classifying the information. Ensure they have a clear understanding of the classification framework and categories. Training may involve examples, case studies, and practical exercises to enhance their classification skills.
5. Apply Classification: Analysts review individual pieces of open-source information and assign them to appropriate categories based on the established framework. This step requires careful analysis, critical thinking, and knowledge of the subject matter. Analysts may use tools like spreadsheets, databases, or specialized software to manage and organize the classified information.
6. Review and Quality Control: Implement a review process to validate the accuracy and consistency of the classification. This step helps identify any errors, inconsistencies, or ambiguities in the classification and ensures the quality of the classified data [8].
7. Analyze and Extract Insights: Once the information has been classified and organized, analysts can start performing deeper analysis and extracting insights from the data. They can identify patterns, trends, relationships, and anomalies within and across different categories, leading to actionable intelligence.
8. Maintain and Update: Open-source information is dynamic and constantly evolving. It is essential to regularly review, update, and maintain the classification framework to accommodate new information and changing requirements.

By following these steps, manual classification of open-source information using frameworks enables analysts to effectively manage and extract valuable insights from large volumes of data, enhancing decision-making processes across various domains such as security, business intelligence, and research.

Automatic classification algorithms are computational methods designed to categorize data into predefined classes or categories without human intervention. These algorithms leverage machine learning and artificial intelligence techniques to automatically learn patterns and relationships within the data, enabling them to make predictions or assign labels to new, unseen instances [10].

Here are some commonly used automatic classification algorithms:

1. Naive Bayes: This algorithm is based on Bayes' theorem and assumes that the presence of a particular feature in a class is independent of the presence of other features. Naive Bayes classifiers are simple and efficient, making them particularly useful for text classification tasks.
2. Decision Trees: Decision tree algorithms construct a tree-like model of decisions and their possible consequences. They partition the data based on features and recursively split the data into smaller subsets until a stopping criterion is met. Decision trees are easy to interpret and can handle both categorical and numerical data.
3. Random Forest: A random forest is an ensemble learning method that combines multiple decision trees. Each tree is trained on a random subset of the data, and the final prediction is obtained by averaging or voting across all the individual tree predictions. Random forests are known for their robustness and ability to handle high-dimensional data.
4. Support Vector Machines (SVM): SVM is a powerful algorithm for both classification and regression tasks. It aims to find the optimal hyperplane that maximally separates data points belonging to different classes. SVMs can handle complex decision boundaries and are effective even in cases where the data is not linearly separable by transforming it into higher-dimensional space.
5. k-Nearest Neighbors (k-NN): The k-NN algorithm classifies new instances by assigning them the label of the majority of their k nearest neighbors in the training set. The value of k, which represents the number of neighbors considered, is a parameter that can be tuned. k-NN is a simple yet effective algorithm that can handle both classification and regression tasks.
6. Neural Networks: Neural networks are powerful models inspired by the structure and function of the human brain. They consist of interconnected nodes (neurons) organized in layers. Neural networks learn through a process called backpropagation, adjusting the weights between neurons to minimize the prediction error. Deep learning, a subfield of neural networks, utilizes architectures with multiple hidden layers and has achieved state-of-the-art performance in various classification tasks.

These are just a few examples of automatic classification algorithms, and there are many other variants and techniques available. The choice of algorithm depends on the characteristics of the data, the complexity of the classification problem, and the available computational resources. Additionally, preprocessing steps such as feature selection, normalization, and dimensionality reduction may be applied to enhance the performance of these algorithms [9].

Semi-automatic classification algorithms, also known as interactive classification algorithms, combine the power of

 Digital Object Identifier 10.32754/JMT.2023.1.08

automatic classification algorithms with human input and guidance. These algorithms involve an iterative process where human experts actively participate in the classification process, refining and validating the results produced by the algorithm. Here's an overview of semi-automatic classification algorithms:

1. Active Learning: Active learning algorithms aim to reduce the human effort required for annotation by selectively choosing the most informative instances for labeling. These algorithms iteratively query the human expert to label instances that the algorithm is uncertain about. The expert labels these instances, and the algorithm incorporates this feedback to improve its performance. Active learning helps optimize the use of human resources by focusing on the most relevant instances for labeling.
2. Co-training: Co-training algorithms are suitable when the available labeled data is limited, but there are multiple views or sets of features available for each instance. The algorithm trains multiple classifiers, each using a different subset of features, and each classifier labels unlabeled instances. The instances on which the classifiers agree are considered confidently labeled and are added to the training set. Co-training iterates this process, incrementally increasing the labeled data and improving classification performance [11].
3. Interactive Clustering: Interactive clustering algorithms involve user interaction to guide the clustering process. The algorithm initially performs an unsupervised clustering on the data and presents the results to the user. The user can then provide feedback by assigning or modifying cluster labels, merging or splitting clusters, or specifying constraints. The algorithm incorporates this feedback and updates the clustering accordingly. Interactive clustering enables the user to incorporate domain knowledge and preferences into the classification process.
4. Human-in-the-Loop Classification: Human-in-the-loop algorithms combine the strengths of automatic classification algorithms with human experts in an iterative loop. The algorithm performs an initial classification, and the results are presented to the human expert for review and refinement. The expert can modify the classification results, correct errors, or add new labels. The algorithm then incorporates this feedback and updates the classification. This iterative loop continues until satisfactory results are achieved.
5. Rule-based Systems: Rule-based systems involve defining a set of rules or heuristics that guide the classification process. These rules can be based on domain knowledge, expert guidelines, or patterns identified from labeled data. The classification algorithm applies these rules to classify instances, but human experts have the flexibility to modify or add new rules as needed. Rule-based systems provide transparency and interpretability in the classification process.

Semi-automatic classification algorithms leverage the expertise of human analysts while harnessing the efficiency and scalability of automatic algorithms. They are particularly useful in scenarios where labeled data is limited, complex domain knowledge is required, or human judgment is crucial for accurate classification. The iterative nature of these algorithms allows for continuous improvement and adaptation to changing data and requirements [13].

III. Examination of the Most Important Algorithms

Some of the most important ones are represented and categorized by type and name, and all of them will be represented in a picture as a proof of working, e.g. “in Table I”.

TABLE I. THE ALGORITHMS

Title	Type
Random Forest	Automatic
Neural Network	Automatic
SVM	Automatic
Interactive Clustering	Semi-automatic

1. Random Forest Classifier is an ensemble learning method that combines multiple decision trees to make predictions. It is a popular and powerful machine learning algorithm used for classification tasks [3].

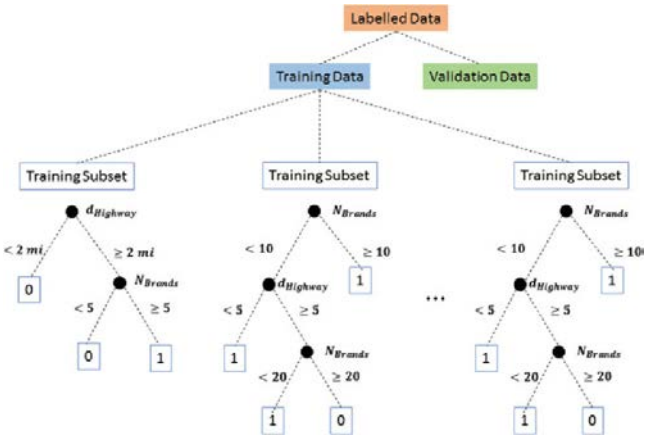

Figure 1. Random Forest Classification

2. A neural network, also known as an artificial neural network (ANN) or simply a neural net, is a computational model inspired by the structure and functioning of the human brain. It is a powerful machine learning algorithm that can learn from data, recognize patterns, and make predictions or decisions. Neural networks have gained significant popularity due to their ability to solve complex problems across various domains [12].

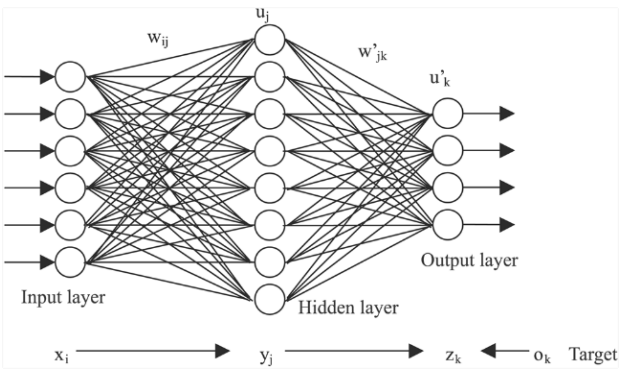

Figure 2. Neural Network

3. Support Vector Machines (SVMs) are powerful and versatile machine learning algorithms used for both classification and regression tasks. They are particularly effective in solving complex problems with a clear separation between classes. SVMs have been widely used in various domains, including image recognition, text categorization and bioinformatics. SVMs offer several advantages, including their ability to handle high-dimensional data, their resistance to overfitting, and their ability to capture complex decision boundaries; however, they can be computationally expensive for large datasets and require careful selection of the hyperparameter.

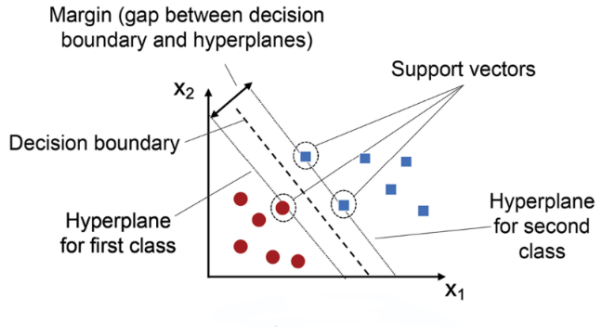

Figure 3. Support Vector Machine

4. Interactive clustering is a clustering approach that involves user interaction to guide and refine the clustering process. It combines the power of automated clustering algorithms with human expertise, allowing users to provide input, constraints, or feedback to improve the clustering results. The goal of this approach is to incorporate human knowledge, domain expertise, and preferences into the clustering process to achieve more accurate and meaningful clusters [14].
Interactive clustering allows users to inject their domain knowledge, expertise, and preferences into the clustering process, making it more flexible and adaptable to specific requirements. It bridges the gap between automated clustering algorithms and the user's understanding of the data, enabling more meaningful and accurate clustering results.
Interactive clustering has applications in various domains, such as data exploration, pattern discovery, recommendation systems, and information retrieval. It helps users uncover hidden patterns, gain insights, and make informed decisions based on the clustering results that align with their domain expertise.

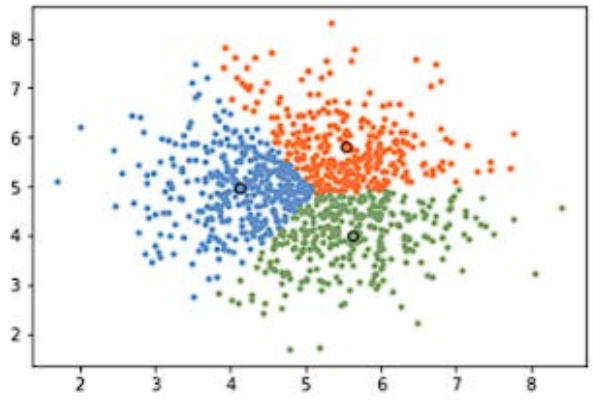

Figure 4. Interactive clustering

IV. MATH

Mathematical equations which are behind each of the algorithms are a very important part of understanding how these are working.

Gini impurity is a measurement used to build decision trees to determinate how the features of a dataset should split nodes to form the three as follows:

$$\sum_{i=1}^{C} f_i\left(1-f_1\right) \tag{1}$$

f_i is the frequency of label i at a node and C is the number of unique labels.

Entropy is an information theory metric that measures the impurity or uncertainty in a group of observations. It determines how a decision tree chooses to split data as follows:

$$\sum_{i=1}^{C} -f_i \log\left(f_i\right) \tag{2}$$

f_i is the frequency of label i at a node and C is the number of unique labels.

Let's consider Fig. 3, in order to separate red dots from the blue ones a hyperplane was created, so, a vector $\vec{w}$ was drawn such that it is perpendicular to our line while the vector $\vec{u}$ is a point vector.

So the dot product of $\vec{w}$ and $\vec{u}$ decides whether the point u is red or blue, if it is greater than a certain value - it is blue. Else it is red

$$\vec{w}\times\overrightarrow{xb}+b\geq 1\overrightarrow{w\times xr}+b\leq 1 \tag{3}$$

$$y\left(\vec{w}\times\overrightarrow{xr}+b\right)-1\geq 0 \tag{4}$$

When the maximization of the margin is required, we can use a constraint optimization technique called Lagrange Multiplier technique, to find optimal value of any parameters as follows:

$$L\left(x,y,\lambda\right)=R\left(x,y\right)-\lambda\left(B\left(x,y\right)-b\right) \tag{5}$$

R – function to be maximized; B – constraints; λ – Lagrange multiplier; x, y – parameters.

To maximize the width we need to prove that $\Delta L=0$ and after doing some calculations the following interesting thing will result:

$$\vec{w}=\sum_i a_i y_i x_i \sum_i a_i y_i = 0 \tag{6}$$

Now, we can put the corresponding values in the Lagrange equation and we must get the following equation:

$$L=\sum a_i-\frac{1}{2}\sum_i\sum_j a_i a_j y_i y_j \overrightarrow{x_i x_j} \tag{7}$$

 Digital Object Identifier 10.32754/JMT.2023.1.08

V. CONCLUSION

In conclusion, automatic and semi-automatic classification algorithms have revolutionized the field of data analysis and decision-making processes across various industries. These algorithms have proven to be effective in handling large volumes of data and extracting meaningful patterns and insights from complex datasets. Through their automated and semi-automated nature, these algorithms have significantly reduced the time and effort required for classification tasks, allowing organizations to make faster and more accurate decisions.

One of the key advantages of automatic classification algorithms is their ability to handle massive datasets efficiently. By leveraging advanced computational techniques, such as machine learning and artificial intelligence, these algorithms can process and analyze vast amounts of data in a fraction of the time it would take a human expert. This scalability makes automatic classification algorithms invaluable in domains where timely decision-making is crucial, such as finance, healthcare, and cybersecurity.

Semi-automatic classification algorithms, on the other hand, strike a balance between human expertise and automated processing. These algorithms combine the power of machine learning with human input to achieve accurate and reliable results. By involving human experts in the classification process, semi-automatic algorithms can overcome certain limitations of fully automated approaches, such as dealing with ambiguous or uncertain data. Human intervention can provide context, domain knowledge, and critical thinking, enhancing the quality of the classification outcomes.

Both automatic and semi-automatic classification algorithms have their own strengths and weaknesses. Automatic algorithms excel in situations where there is an abundance of labeled data and the classification problem is well-defined. They can learn from historical data patterns and make predictions or classifications based on those patterns. However, automatic algorithms may struggle when faced with novel or outlier cases that fall outside the scope of their training data.

Semi-automatic classification algorithms offer a more adaptable and flexible approach. They allow human experts to guide the algorithm by providing feedback and refining the classification results. This collaboration between humans and machines creates a synergy that leads to more accurate and robust classifications. Nevertheless, semi-automatic approaches require more human involvement and can be time-consuming, especially when dealing with large datasets.

In summary, the use of automatic and semi-automatic classification algorithms has transformed the way organizations extract valuable insights from data. These algorithms have the potential to enhance decision-making processes, increase efficiency, and improve accuracy. Automatic algorithms offer scalability and speed, while semi-automatic algorithms leverage human expertise to handle complex and uncertain scenarios. The choice between these approaches depends on the specific requirements of the classification task, the availability of labeled data, and the level of human involvement desired.

As technology continues to advance, we can expect further improvements and refinements in automatic and semi-automatic classification algorithms. With the integration of emerging technologies, such as deep learning and natural language processing, these algorithms will become even more powerful and capable of handling increasingly complex datasets. Ultimately, the future of classification lies in harnessing the strengths of both human and machine intelligence to achieve optimal results in data analysis and decision-making processes.

ACKNOWLEDGEMENT

This work was supported by a grant of the Ministry of Research, Innovation and Digitization, CNCS/CCCDI - UEFISCDI, project number PN-III-P2-2.1-SOL-2021-0063, within PNCDI III.

REFERENCES

[1] A. K. Singh, "Classification Techniques: A Review", *International Journal of Advanced Research in Computer Science and Software Engineering*, vol. 5, no. 2, pp. 523-527, 2015.

[2] S. B. Kotsiantis, "Supervised Machine Learning: A Review of Classification Techniques", *Progress in Artificial Intelligence*, vol. 1, no. 3, pp. 147-172, 2007.

[3] C. D. Manning, P. Raghavan, and H. Schütze, "*Introduction to Information Retrieval*", Cambridge University Press, 2008.

[4] M. A. Hossain and M. A. B. Siddiquee, "A Survey on Data Mining Techniques and Applications", *International Journal of Computer Science and Information Technology*, vol. 5, no. 2, pp. 1-10, 2013.

[5] J. Han, M. Kamber, and J. Pei, "*Data Mining: Concepts and Techniques*", 3rd edition, Morgan Kaufmann, 2011.

[6] R. A. Furuhaug, "*Open Source Intelligence Methodology*", University College Dublin, 15 May 2019.

[7] N. A. Hassan, R. Hijazi, "Open Source Intelligence Methods and Tools: A Practical Guide to Online Intelligence", 2018.

[8] R. Ghioni, M. Taddeo and L. Floridi, "*Open source intelligence and AI: a systematic review of the GELSI literature*", Springer Nature, 28 Jan. 2023.

[9] G. T. Ungureanu, "*OPEN SOURCE INTELLIGENCE (OSINT). THE WAY AHEAD*", "Mihai Viteazul" National Intelligence Academy, Bucharest, Romania, 2021

[10] B. Michael, "Urce Intelligence Techniques Resources For Searching And Analyzing Online Information", CCI Publishing, 2022.

[11] H. J. Williams, I. Blum, "Defining Second Generation Open Source Intelligence (OSINT) for the Defense Enterprise", RAND Corporation, 2023.

[12] R. A. Pinto, M. J. Hernández, C. C. Pinzón, D. O. Díaz and J. C. C. García, "Open source intelligence (OSINT) as support of cybersecurity operations. Use of OSINT in a colombian context and sentiment Analysis", Universitad Distrital "Francisco Jose de Caldas", 03 Jun. 2018.

[13] https://www.imperva.com/learn/application-security/open-source-intelligence-osint/ accessed 25th June 2023.

[14] https://www.sans.org/blog/what-is-open-source-intelligence/ accessed 22th Jun. 2023.