applied system innovation

MDPI

Article

Secure Aviation Control through a Streamlined ADS-B Perception System

Qasem Abu Al-Haija [1,*] and Ahmed Al-Tamimi [2]

[1] Department of Cybersecurity, Faculty of Computer & Information Technology, Jordan University of Science and Technology, P.O. Box 3030, Irbid 22110, Jordan
[2] Department of Cybersecurity, King Hussein School of Computing Sciences, Prince Sumaya University for Technology, P.O. Box 1438, Amman 11941, Jordan
* Correspondence: qsabuhaija@just.edu.jo; Tel.: +962-2-7201000

Abstract: Automatic dependent surveillance-broadcast (ADS-B) is the future of aviation surveillance and traffic control, allowing different aircraft types to exchange information periodically. Despite this protocol's advantages, it is vulnerable to flooding, denial of service, and injection attacks. In this paper, we decided to join the initiative of securing this protocol and propose an efficient detection method to help detect any exploitation attempts by injecting these messages with the wrong information. This paper focused mainly on three attacks: path modification, ghost aircraft injection, and velocity drift attacks. This paper aims to provide a revolutionary methodology that, even in the face of new attacks (zero-day attacks), can successfully detect injected messages. The main advantage was utilizing a recent dataset to create more reliable and adaptive training and testing materials, which were then preprocessed before using different machine learning algorithms to feasibly create the most accurate and time-efficient model. The best outcomes of the binary classification were obtained with 99.14% accuracy, an F1-score of 99.14%, and a Matthews correlation coefficient (MCC) of 0.982. At the same time, the best outcomes of the multiclass classification were obtained with 99.41% accuracy, an F1-score of 99.37%, and a Matthews correlation coefficient (MCC) of 0.988. Eventually, our best outcomes outdo existing models, but we believe the model would benefit from more testing of other types of attacks and a bigger dataset.

Keywords: aviation security; aircraft surveillance; automatic dependent surveillance-broadcast (ADS-B); ADS-B security threats; ADS-B message injection; ML detection model

Citation: Abu Al-Haija, Q.; Al-Tamimi, A. Secure Aviation Control through a Streamlined ADS-B Perception System. *Appl. Syst. Innov.* **2024**, *7*, 27. https://doi.org/10.3390/asi7020027

Academic Editor: Felix J. Garcia Clemente

Received: 30 January 2024
Revised: 9 March 2024
Accepted: 22 March 2024
Published: 26 March 2024

1. Introduction

Since the usage of drones is becoming increasingly relevant these days and is still increasing for both commercial and military reasons, and because some of them are used without the need of a human, this forced the need to have a framework that would try and eliminate the chances of any air collision between the different types of aircraft. Thus, automatic dependent surveillance-broadcast (ADS-B) was introduced [1]. Automatic dependent surveillance-broadcast (ADS-B) is an air traffic control system based on information such as the position, identity, time, heading, and velocity received by ground sensors and transmitted via an aircraft periodically with the intent of surveillance. ADS-B is considered a crucial asset in air traffic control, and it is estimated that 42.9% of commercial aircraft equip hardware to provide automatic dependent surveillance-broadcast (ADS-B) [2].

The intention behind designing ADS-B is to improve air traffic control by making it easier, eliminating limitations of previous air traffic control models; it serves as a replacement for secondary surveillance radar (SSR), bearing in mind cyberattacks against aircraft or the aviation industry [3]. Still, despite all of the advantages that ADS-B brings to the aircraft control field, it needs some security limitations. It is also vulnerable to multiple types of attacks, such as eavesdropping, jamming, and message injection, while

also lacking any security measures to ensure the safety of the data transmitted. Failure to formally distinguish between actual and spoofed ADS-B messages is one of the most frequent issues that lead to misunderstandings while communicating with an aircraft or a sensor. Additionally, attacks on ADS-B systems are of many types. Figure 1 demonstrates a variety of attacks that could affect ADS-B systems [3,4].

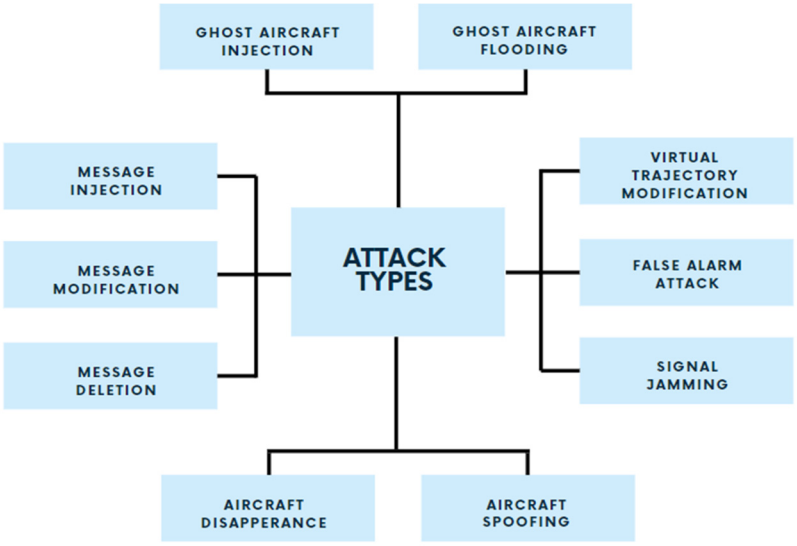


Figure 1. Attacks Against ADS-B.

Nowadays, as stated earlier, around half of commercial aircraft are equipped with ADS-B hardware to improve aircraft traffic control as a whole. With the evident orientation towards using ADS-B and the rapid increase in the sophistication and diversity of cyber-attacks and their techniques, this paper will focus on providing security solutions for detecting false information gathered by sensors to improve aircraft traffic control and security.

Unfortunately, no previous studies have introduced a dataset or a detection model in a realistic environment. This has led us to take the initiative to find a dataset that contains both genuine messages and injected messages that were transmitted and received by a sensor. We also used different classification approaches to provide a higher accuracy rate and enhance the overall security of the detection systems.

The primary contribution of this paper is the proposal of a machine learning-based detection model for classifying injected messages using a specialized dataset. The following is a list of the specific contributions.

- We outlined earlier research related to ADS-B, its threats and vulnerabilities, and some of the proposed solutions so we could determine the gap and select the best method to apply our study to maximize the benefits for the aviation industry.
- We rendered the qualified dataset for ease of use. We also analyzed and performed an in-depth study of the different types of injected messages directed against aircraft and potential defenses against such attacks to aid and optimize our proposed solution.
- We developed an efficient classification model using multiple techniques, and we achieved solid results in detecting injection within the messages used in path modification, ghost aircraft injection, and velocity drift attacks, bearing in mind that it needed to be performed in a rapid manner and with the least amount of training required. Our best results are listed below:
    - o **Binary classification**: accuracy of 99.14%, F1-score of 99.14%, and MCC of 0.982 using a random forest classifier.
    - o **Multiclass classification**: accuracy of 99.41%, F1-score of 99.37%, and MCC of 0.988 using a random forest classifier.

2. Background Review

Automatic dependent surveillance broadcast (ADS-B) is increasingly utilized within the aircraft industry. Since both aircraft need to be piloted or remotely controlled to avoid any misinformation that could cause a collision, there is a higher demand for a security solution to solve and mitigate most of the threats and vulnerabilities ADS-B suffers.

Automatic dependent surveillance-broadcast (ADS-B) is an air traffic control system based on information such as position, identity, time, heading, and velocity received by ground sensors and transmitted periodically via an aircraft with the intent of surveillance. ADS-B messages are acquired from the OpenSky network. The reason was to make aviation data, such as air traffic information, available to the public. Aircraft transponders broadcast ADS-B messages, and all nearby receivers on the same communication channel or within the transmission range could capture these messages and the information contained within them.

2.1. ADS-B Security Threats and Vulnerabilities

ADS-B suffers from various attacks and vulnerabilities. Below are the most known attacks that threaten an ADS-B system and their description [4].

- **Message injection**: if the attacker has the right equipment, an actual and legitimate counterfeit ADS-B message could be generated and injected into an ADS-B System.
- **Message deletion:** physically deleting ADS-B messages using constructive or destructive interference differs from jamming because it drops the whole message rather than blocking it.
- **Message modification**: altering the contents of an ADS-B message using different techniques such as message injection and bit flipping.
- **Signal jamming**: interrupting the signals of ADS-B communications channels using radio frequency devices that send multiple requests leads to jamming and disrupting the communication between the aircraft and ground stations.
- **False alarm attack**: modifying a message to indicate a false alarm. Such an attack could indicate that an aircraft has been hijacked while operating normally.
- **Virtual trajectory modification**: changing the received trajectory of an aircraft by using message manipulation techniques.
- **Ghost aircraft injecting**: this attack creates an imaginary aircraft by faking ADS-B messages within a communication channel.
- **Ghost aircraft flooding**: like ghost aircraft injecting, this attack injects multiple imaginary aircraft simultaneously to cause disruption and confusion and could cause a denial of service in different surveillance systems.
- **Aircraft disappearance**: the objective here is to cause failure within the collision avoidance systems and cause confusion with the ground stations where suddenly all of the messages received from an aircraft are deleted and no longer transmitted; this could force an aircraft to perform an emergency landing to avoid any risks.
- **Aircraft spoofing**: the objective of this attack is to spoof an aircraft's ICAO number, a special identifier for the aircraft transmitter; the main risk here is that any aircraft, even if unauthorized, could pass into the premises of a country as a normal aircraft without triggering any alarms.

2.2. Machine Learning Classifiers

The machine learning classifiers used in this study to create a machine learning model are described below, along with their definitions.

- **Random forest**: T. K. Ho originally invented it in 1995, and then L. Breiman and A. Cutler improved it in 2006. Also known as random decision forests, this ensemble-supervised learning technique builds several decision trees during the training phase and is typically used for classification and regression [5].

- **AdaBoost**: this 1995 model, created by Y. Freund and R. Schapire, can be used with various machine learning methods to improve performance. An ensemble-supervised learning approach uses iteration to strengthen weak classifiers by learning from their flaws [6].
- **Naive Bayes (NB)**: the Bayes theorem is the foundation of a supervised machine-learning method used for classification. Under the naive assumption that the pairs of characteristics are independent, this algorithm is regarded as fast and a good classifier. Still, it could **perform better when estimating or predicting** [7].
- **Neural** networks (NN): a machine learning subset that uses supervised learning. It consists of many algorithms that attempt to imitate the workings of the human brain to uncover links in datasets [8].
- **Support machine vector (SVM)**: V. Vapnik et al. discovered it in 1993. It is regarded as one of the most reliable prediction algorithms and is another supervised machine learning technique. It is a non-probabilistic classifier because its foundation divides the samples into two groups [9].
- **K-nearest neighbors (kNN)**: E. Fix and J. Hodges created the initial version in 1951. Another supervised machine learning model is kNN, or closest neighbor, which assigns various weights to neighbor contributions based on the assumption that objects are similar when they are close together or apart [10].

3. Related Work

This section will showcase the most relevant and recent academic research on ADS-B, its development, some of the problems and security threats ADS-B faces, and recently proposed solutions.

3.1. ADS-B and ADS-B Vulnerabilities

This subsection will showcase ADS-B and the evolution of the security threats it faces. First of all, A. Costin and A. Francillon [11] used MATLAB (2021b) and software-defined radio (SDR) to transmit and receive signals on a frequency that is usually pre-specified, and they used a plane gadget radar as their aircraft to illustrate the first public ADS-B injection. This study emphasized how ADS-B, the newly adopted technology, faces multiple threats and vulnerabilities that must be addressed before deployment.

Then, several studies focused on how low-cost equipment and technology could put aircraft in danger and cause devastating damage. So, in their research, M. Leonardi et al. [12] devised a realistic jamming model and analyzed how it could affect aircraft surveillance. They presented how a top-tier jammer could affect and disrupt ADS-B signals and communications. They also showcased how attacks on ground stations are more menacing than attacks on aircraft because they are less costly to be carried out.

Eskilsson et al. [13] also demonstrated a cost-efficient ADS-B attack costing around $300. They utilized Python as a programming language to perform ADS-B data encoding, HackRF as a radio frequency transmitter, and the message was received utilizing dump1090 with an RTL-SDR transceiver. Their main concern was to present the availability of such a low-cost attack setup and how it might motivate many attackers to perform malicious activity in the aviation industry.

Moreover, the authors in [14] presented how a low-cost jammer can jam ADS-B communications. They jammed by creating an interfering signal in an ADS-B communication channel, which resulted in the original signal being fully or partially deformed.

Another interference attack was performed by N. Pearce et al. [15]. This attack was executed by sending out fake signals. Their observations show that the interference caused an error rate of 32.39%, which is fairly high. They concluded that ADS-B is insecure and could be exploited by cheap and trivial technology.

Schäfer et al. [16] also presented how the attacks targeting ADS-B are low-cost and could yield a very high success rate. The authors transmitted fake signals and performed multiple attacks, such as ghost aircraft flooding, ghost aircraft injection, and ground

sensor/station flooding. The outcome was that no critical air traffic control measure should be solely based on ADS-B data without being fully secure and prone to attacks.

Other studies focused on how pilots and autopilots react to such misleading information an aircraft receives. Therefore, Manesh et al. [17] investigated how an injection attack could impact ADS-B messages; their main focus was testing how the autopilot responds to ghost aircraft injection. This type of attack shows an immediate and unexpected appearance of an aircraft (ghost aircraft) close to the responding aircraft. The misleading information had triggered the autopilot to turn to gain a safe route rapidly. According to the authors, this attack on ADS-B can affect pilots' decisions and other ground sensors, disturb air traffic, and hugely increase the rate of an aircraft collision.

Y. Haddad et al. [18] went on to test how humans would react to misleading and spoofed data; they chose 50 participants and tested their reactions while completing different tasks using a drone simulator. During their tasks, the attacker would send spoofed data of fake aircraft nearby to cause alteration in the drone's path and see how the pilot would react to such information. They gave interesting results from the perspective of having participants with multiple nationalities. However, their methodology still requires testing how an experienced pilot reacts to such false and misleading information.

Another take was to perform other types of attacks rather than just jamming signals to see how reliable ADS-B is. In their research, M. Strohmeier et al. [19] analyzed an OpenSky sensor network in Europe with a distinct focus on the 1090 MHz channel. They showed that ADS-B messages transmitted through this channel are at high risk of radio frequency attacks; these attacks could affect an aircraft's collision avoidance. Another outcome was that they reported a huge loss of messages due to the vast traffic on that specific communication channel. They recommended addressing these security issues before fully relying on ADS-B within the aviation industry.

Moreover, Odin and Gruneau [20] showcased a relatively recent type of attack called teleporting ghost aircraft; they achieved it using Sentry and HackRF, where reports of an aircraft position were signaled at different locations and were moving around in an unstable way. Thus, that aircraft movement seemed to break the fundamental laws of physics. They pointed out that the ADS-B receiver trusted the protocol without verification or validation. They advised that if ADS-B is used more within the aviation industry with its lack of security, an attacker could take control of any aircraft because of the amount of control they would gain.

All in all, these studies have shown that even though ADS-B is an evolution in the industry and brings advantages, it still needs to be more reliable and secure; more work is needed to secure and utilize it fully. We summarize the most relevant studies about ADS-B and its problems in Table 1.

Table 1. Summary of the most relevant studies about ADS-B and its problems.

Ref.	Approach	Contributions	Limitations
[11]	— ADS-B injection using MATLAB and SDR to inject ADS-B publicly.	— The first ADS-B injection attack. — An insight into the threats and vulnerabilities that exist within ADS-B.	— It provided no suggestions for dealing with those security threats and vulnerabilities.
[12]	— Performed a signal-jamming attack on an ADS-B network.	— Attacked an ADS-B network by jamming signals and interrupting communications. — Network-based mitigation technique increasing or modifying the sensors' distribution to redundancy can help mitigate the jamming. — Sensor-based mitigation technique; multichannel signal processing, using a multichannel receiver with more sector antennas.	— The solutions are very costly since they require installing more hardware on a large geographic space.

Table 1. *Cont.*

Ref.	Approach	Contributions	Limitations
[13]	— Demonstrated a low-cost attack on ADS-B.	— Showed the availability of low-cost attack setup. — Highlighted that a low-cost attack would motivate many attackers to affect the aviation industry adversely.	— The study suggested that no mitigation methods were used to avoid or deal with this attack.
[14]	— Low-cost jammer to interfere with signals communicated within an ADS-B communication channel.	— Able to deform the signals, either fully or partially. — Rendered ADS-B communications insecure.	— No security measures were suggested.
[15]	— Interference attack by sending forged signals.	— Achieved a fairly high error rate of 32.39%. — Concluded that ADS-B is insecure.	— They did not provide ways to intercept such signals from reaching aircraft.
[16]	— Transmitted multiple fake signals to flood the communication channels.	— ADS-B is weak in terms of security and is prone to attacks. — Critical air traffic control should not solely rely on ADS-B data.	— No suggestions to improve ADS-B security.
[17]	— Autopilot responses to ghost aircraft injection.	— Performed ghost aircraft injection attack. — Tested how the autopilot responds to a random aircraft suddenly appearing.	— This study suggested no security measures and only focused on the reactions of the autopilot.
[18]	— A drone simulator will be used to assess the reaction of 50 participants to injected messages.	— Showed how pilots would react to spoofed data an attacker sends. The reactions yielded multiple perspectives, and they based the reactions on the nationalities of the participants.	— More in-depth testing is required to show why pilots made certain decisions after receiving false information.
[19]	— Radiofrequency attacks on ADS-B messages.	— A huge loss was reported after performing radio frequency attacks. — Effects of this attack on the collision avoidance system within an aircraft. — ADS-B is currently unreliable for full deployment.	— This study suggested no measures to make the collision avoidance system prone to this attack.
[20]	— Presented a new type of attack; teleporting ghost aircraft.	— Transmit signals of an inexistent aircraft position heading in different positions against the law of physics. — The ADS-B protocol lacks any verification or validation of the information. — This vulnerability can enable attackers to take control of aircraft.	— It did not provide any suggestions on implying verification or validation methods into the ADS-B protocol or any security practices to enhance the security.

3.2. Securing ADS-B

This subsection will discuss how recent researchers tried to solve the current problems of ADS-B security and what limitations they faced while doing so. It will be divided into multiple sub-sections based on the detection method utilized in each study.

3.2.1. Cryptography and Fingerprinting

This subsection will discuss the solutions based on using cryptography and fingerprinting of the signals to provide a secure framework for ADS-B. There were different takes and efforts to try to secure ADS-B using cryptography. Initially, M. Strohmeier et al. [2] suggested that using public key cryptography and fingerprinting is the security measure

that ADS-B needs to be dependable and trustworthy. Unfortunately, this would require protocol changes, making it costly, especially over time.

M. Leonardi et al. [21] proposed a detection model that detects malicious messages by fingerprinting wireless radio signals. Still, even though their proposed method is unfamiliar, only 50 percent of the malicious signals were detected. Additionally, Kacem et al. [22] improved on a previous framework by merging cryptography and timestamp validation, and even though it had huge value, it did not affect the ADS-B messages with any form of alteration that could cause any corruption.

M. Strohmeier et al. [23] tried another method that combined the information of consecutive signals sent by the two antennas on an aircraft to detect attacks that used a single transmitter. On the other hand, the authors in [24] replaced the cyclic redundancy check found at the end of an ADS-B message with a HASH produced by HMAC. They made a compound of different messages in the same digest. They verified them together to have enough space to add the hash into the ADS-B messages to verify all of the messages when they were received.

In another study, M. Strohmeier et al. [25] fingerprinted the aircraft equipped with ADS-B to authenticate a stream of messages. They performed fingerprinting based on the time difference of arrival (TDoA) information of the messages they received from aircraft as they were received on various sensors. Then, a main server evaluates any similarity between the gathered samples and local data to catch any distinction.

Lastly, H. Shen et al. [26] emphasized that messages sent by aircraft are not encrypted in any form; since the ADS-B messages are in basic form, it is relatively easy to forge or tamper with ADS-B messages. Their research established an anti-counterfeiting system that consists of four stations based on the time difference of arrival to locate the actual position of the ADS-B signal and compare it with any alteration found in the claimed position in the counterfeit ADS-B message. We summarize the current cryptography and fingerprinting solutions to secure ADS-B in Table 2.

Table 2. Summary of the current cryptography and fingerprinting solutions to secure ADS-B.

Ref.	Approach	Contributions	Limitations
[2]	— Public key cryptography and fingerprinting.	— Proof that cryptography and fingerprinting could improve the security of ADS-B communications.	— Costly changes to the ADS-B protocol must be performed.
[21]	— Fingerprinting wireless radio signals.	— Detect injected messages using fingerprinting on radio signals.	— They were only able to detect 50% of the injected messages.
[22]	— Merged cryptography with timestamp validation.	— Kept the ADS-B message unchanged even though the size is huge.	— Huge and needs further testing to ensure it does not corrupt any messages or affect the detection of injected messages.
[23]	— Fingerprinting data coming from antennas.	— Detect attacks that use a single transmitter by fingerprinting information in the relative sequence they were sent by.	— It only mitigates attacks carried out using a single transmitter.
[24]	— Adding a HASH value to verify ADS-B messages.	— Added HMAC to a sequence of ADS-B messages to verify and validate them.	— Requires further testing to ensure HMAC is suitable for ADS-B messages.
[26]	— Time difference of arrival (TDoA) fingerprinting.	— Fingerprinting based on the TDoA of the messages. — Detecting and distinction in messages by evaluating any similarities with the gathered samples.	— It is time-consuming as the main server needs to evaluate the similarity each time, and it is unreliable since any lost packet will jeopardize the whole checking process.

3.2.2. Based on Time and Location

This subsection will discuss the solutions that utilize the time and location information and the intervals between signals to provide a realistic check of the authenticity of the sent and received signals to secure ADS-B. M. Leonardi [27] discussed different types of anomalies that occurred within ADS-B messages and used a method to detect intrusion using a sample of 16,000 messages from 52 different aircraft; his method utilizes the time when the information was received, which is retrieved from the sensor clocks, and then he compared it with the actual position of the aircraft. This method allowed him to detect any possible anomalies in the ADS-B messages, such as injection, without any locating algorithms. His model enabled the detection of anomalies in ADS-B messages without solving the hyperbolic inversion problem, and he has shown that his method can improve the synchronization between the different ADS-B sensors within networks.

Sciancalepore and Di Pietro [28] developed Securing Open Skies (SOS), a security framework to secure ADS-B communications and high robustness where a packet loss occurs or a message overhead is missing within an ADS-B message. SOS integrates the well-known timed-efficient stream loss tolerant authentication (TESLA) protocol within the ADS-B message format.

Similarly, Chen and Zhou [29] focused on the attacks and interference of communications between aircraft in China, showing that the original intention of ADS-B technology was to make air traffic control easier as the number of aircraft constantly increases. It can also be used to communicate with ground stations. They proposed three different ADS-B verification methods based on the ADS-B interference in China and possible interference. The first is coverage verification, where the target is instantly discarded when it exceeds the coverage range. The second method is time difference of arrival (TDoA) verification, which requires multiple ground sensors to receive signals from the aircraft simultaneously. The third and final method was cross-verification, which used radar signals to verify whether an ADS-B message was authentic by verifying the target from the coverage area and TDoA.

Lastly, authors in [30] developed a similar solution based on location verification using multi-iteration techniques based on TDoA. Specifically, they leverage multi-alteration techniques based on ADS-B signals to verify the reliability of information broadcasted by aircraft. Still, it is fair to say that these are unreliable solutions because if one packet is lost, the whole communication will not be validated and thus would cause any ADS-B signal received to be invalid or considered corrupt or malicious. We summarize the most relevant solutions utilizing time and location to secure ADS-B in Table 3.

Table 3. Summary of the most relevant solutions utilizing time and location to secure ADS-B.

Ref.	Approach	Contributions	Limitations
[27]	— Proposed the Securing Open Skies (SOS) framework.	— Integrating timed-efficient stream loss tolerant authentication (TESLA) protocol with ADS-B messages. — Able to discard messages if information is missing or if a loss occurrs.	— Requires changes to the ADS-B protocol to integrate TESLA within it. — Only works with packet loss and missing headers, not an actual injection.
[28]	— Coverage verification, time difference of arrival and cross-verification for ADS-B messages.	— Coverage verification technique: a request that exceeds the original target range is immediately discarded. — TDoA verification technique: based on the time difference of arrival for the messages, it is decided if the message is authentic or injected. — Cross-verification technique utilizing both of the techniques to verify ADS-B messages.	— It has only been tested on aircraft communications interference in China. — Costly in terms of time and hardware since multiple sensors are required to work simultaneously to detect injection.

Table 3. *Cont.*

Ref.	Approach	Contributions	Limitations
[29]	— ADS-B anti-counterfeiting system.	— Pointed out that ADS-B messages are easily forged since they are not encrypted. — Anti-counterfeiting system based on the TDoA received from four different stations.	— Needing four stations to confirm the authenticity of a message is time-consuming and inefficient for decision making, and this might only sometimes be available.

3.2.3. Machine Learning Techniques

This subsection will discuss the solutions that utilize different machine-learning techniques to build efficient detection models and the limitations of these proposed models. A. Cretin et al. [4], in their study, discussed the new attacks that the aircraft could be exposed to, such as false data injection, where the attacker could modify the contents of an ADS-B message and swindle other aircraft and sensors that rely on the ADS-B information. They proposed a set of algorithms that works towards not accepting any messages that contain any injection. However, this method still needs work, and modifications to ADS-B need to be applied to the devices on different aircraft.

The authors of [31] carried out a study where they showcased multiple threats and vulnerabilities in what is considered the pillar of next-gen aircraft surveillance, which is currently being used among regions. The authors evaluated and showed the impacts of cyberattacks that utilized radio frequency to affect multiple networks. They implemented 12 attacks and only demonstrated five of them in their research. In total, the authors used 36 different ADS-B in message configurations (13 hardware devices and 22 software) and attacked them using BladeRF and HackRF as attacking devices; they achieved 90% accuracy in detecting spoofed messages and showed that their method could help distinguish the different types of ADS-B messages, which would help with mitigating some types of attacks.

B. Kujur et al. [32] developed a novel method to detect global navigation satellite systems (GNSS) spoofing for aircraft equipped with ADS-B. Since the ADS-B equipment is mandated in civil aircraft in the US and surrounding areas by the Federal Aviation Administration, there was a need to address the GNSS spoofing. They proposed a model that would detect the spoofed signal by comparing the ADS-B inertial navigation system positions to the ones obtained by the spoofed GNSS. We summarize the most relevant solutions utilizing machine learning techniques to secure ADS-B in Table 4.

Table 4. Summary of the most relevant solutions utilizing machine learning techniques to secure ADS-B.

Ref.	Approach	Contributions	Limitations
[4]	— Machine learning to block injected messages.	— Aircraft are exposed to false data injection attacks. — Multiple machine-learning models were tested to block such data.	— More work is needed to improve the results, and changes to the ADS-B protocol must be implemented within the different devices.
[31]	— They performed attacks and analyses to improve machine learning-based detection.	— Performed 12 attacks but only presented five in their paper. — Achieved 90% accuracy in detecting spoofed messages.	— Unclear which ML models were used, and they acknowledged that their model still requires further improvements.
[32]	— Detecting global navigation satellite systems spoofing.	— Detect spoofed messages by comparing the spoofed GNSS messages with the locations from the messages sent by the ADS-B sensors.	— It only works against GNSS spoofing and can detect it based on velocity and location.

3.3. Research Gap

As the studies suggested, ADS-B is currently facing problems related to its security, and much work is still needed to improve on that aspect before fully deploying it in the aviation industry. These vulnerabilities would affect the security and safety of the different aircraft types.

We thoroughly discussed currently proposed solutions and categorized them based on the technique used and the limitations they face, focusing on showing how researchers tried implementing those techniques to minimize or eliminate the risks caused by these vulnerabilities. Some studies suggested securing ADS-B physically, while others proposed a framework such as the TESLA framework and considered it a satisfactory method to secure ADS-B communications without enough proof of robustness and reliability.

While most proposed solutions focused on solving the issues using hardware or cryptography, time, and location information, these techniques proved costly and time-consuming to produce and implement in the different types of aircraft or sensors within an ADS-B network.

However, as we also mentioned, some studies utilized machine learning techniques, which still show limitations, most importantly detection rate, and require a change in the ADS-B protocol to render their solution effective.

The biggest gap observed is that there was a need to aid in developing a detection model that is more dependable, powerful, long-lasting, and time-efficient. Thus, we decided to address the issues we observed within previous studies and tried to develop a solution that is efficient in terms of time and cost, is robust, and that does not require any change to be carried out on the ADS-B protocol. We also decided that we would mainly focus on the detection of fake and forged messages that are used to carry out the following attacks: path modification, ghost aircraft injection, and velocity drift attack to boost the security of this protocol, which would eventually increase the reliability and safety of it and allow to deploy and utilize it fully within the aviation industry.

4. Methodology

A machine learning-based detection model is adopted to achieve an efficient detection model of malicious ADS-B messages. The machine learning approach would require a dataset to train on, and then a testing dataset that the model has never seen before would be used to assess the model's performance. As was previously mentioned, the dataset by H. O. Slimane et al. was published in 2022 but has yet to be used by academics. Therefore, several machine learning techniques will be used to preprocess, balance, and test the unique dataset to verify that the detection model performs at its best [33]. This section covers the methodology of the proposed model and will be divided into three subsections. Section 4.1 will provide a detailed view of the dataset, followed by Section 4.2, which highlights data preprocessing steps. Section 4.3 outlines the various machine learning classifiers used and tested in this study. Figure 2 presents an overview of the proposed model for this study. Our main focus using this model is to detect and classify path modification, ghost aircraft injection, and velocity drift attacks.

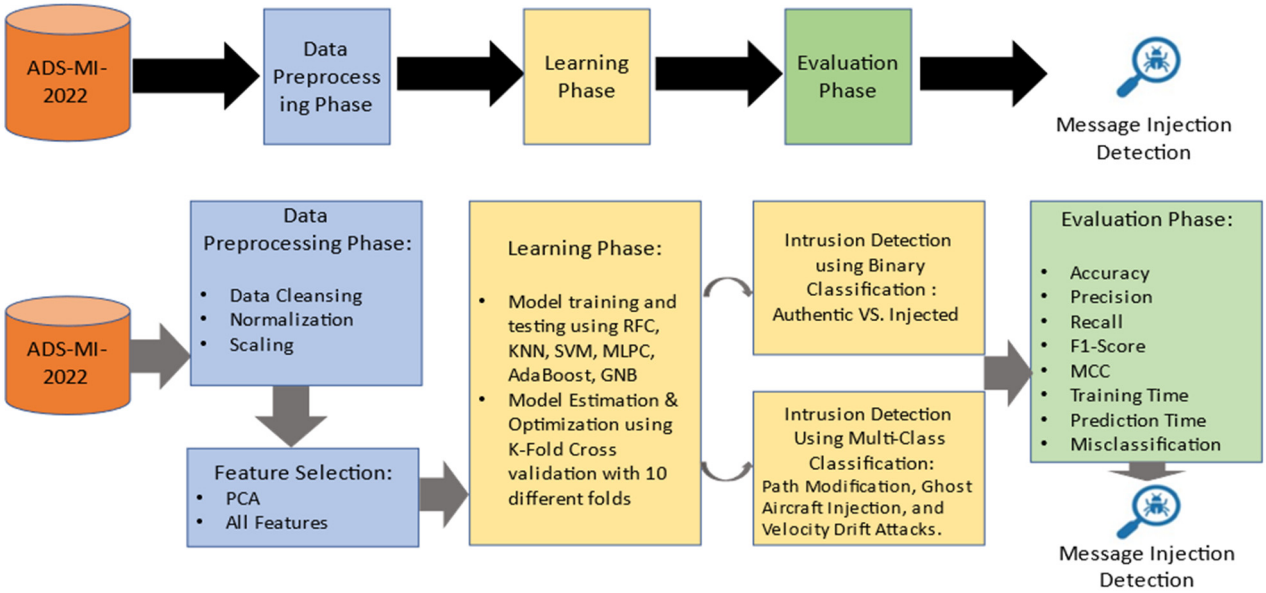


Figure 2. An overview of the proposed model.

4.1. The Dataset

This sub-section will discuss the dataset regarding features, records, and content; the dataset is publicly available on Mendeley Data and includes authentic ADS-B messages samples acquired from the OpenSky Network and injected messages simulated using PyCharm (version 2021.3.1). There are no personally identifiable information violations in this dataset. The adopted dataset by H. O. Slimane et al. has two files totaling 3 MB. This dataset contains authentic ADS-B message samples acquired from the OpenSky Network and injected messages for path modification, ghost aircraft injection, and velocity drift attacks.

The dataset authors used PyCharm, a development environment for programming integrated within Python, to simulate malicious messages with the required attack intentions. The dataset is balanced and contains 22,316 messages, of which 11,158 are authentic, and 11,158 were injected within the dataset files. The CSV files were specifically chosen because they typically provide data distribution information.

After removing authentic messages from the dataset, we noticed that the injected message numbers were unequal, implying that the dataset was imbalanced. This required us to use different over- and under-sampling techniques to balance it and improve the model's quality to help classify imbalanced data and eliminate bias. The following over- and under-sampling techniques were utilized in this study:

- **Random oversampling**: performing random oversampling involves selecting random data from the class that is considered the minority and then replacing or adding that data to the training dataset.
- **Random undersampling**: the opposite of random oversampling, performing random undersampling involves selecting random data from the class that is considered the majority and replacing or adding that data to the training dataset.
- **Synthetic minority oversampling technique (SMOTE)**: an oversampling technique used with imbalanced datasets to help classification using machine learning, performed by generating new data from the existing minority data and then using it to supplement the dataset [34].

4.2. Data Preprocessing

This subsection will discuss data preprocessing, which outlines the processes we carried out on raw data to prepare it for different operations to be conducted on that data. Data preprocessing is a crucial first stage in our study. The methods have recently been

modified to aid in training AI and machine learning models by changing the data into a format that can be processed in machine learning and other data science techniques in a way that is quicker and more efficient than using raw data, bearing in mind that these processes are typically applied at the beginning of the machine learning development pipeline to ensure reliable findings. The dataset is available online in CSV files, which are program-generated data files. The data preprocessing consisted of three main processes, outlined below.

- Data normalization: this mainly consisted of standardizing features for us to be able to utilize it with the machine learning models.
- Data cleansing: we performed data cleansing on the data to make sure there were no errors or duplications while processing the data to avoid bias and conflict in results.
- Data scaling: this process revolved around data encoding, where we changed the results of some selected features from textual values (such as strings and boolean data) to numerical values (integer/float) to be able to use them with the machine learning models.
- Data Shuffling: We used different techniques to randomize the training and testing data sets for more reliable results.

The procedures used during data preprocessing are outlined below.

- We coded labels to convert categorical columns into numerical values on the icao24 and callsign columns. We also removed the remaining categorical features to decrease the margin of error.
- We converted all data to integer/float values using the one-hot encoding technique to ensure that the machine learning model can understand the data, which creates a one-numeric array to encode categorical information [35].
- We split the dataset into two main splits: one represents all features, and the other represents only the label feature (message type).
- We utilized pandas. drop, pandas.resample, and train_test_split to manipulate the dataset before performing any training or testing on our dataset. We also used multiple balancing techniques to aid with the classification process.
- We produced multiple training and testing subsets in different sizes that we randomly shuffled and generated at each iteration to initiate the learning process.

As discussed, data preprocessing is a crucial stage in machine learning since the effectiveness of the machine learning model is directly impacted by the data quality. We processed, updated, and cleaned the dataset during data preprocessing so it fits the machine learning model.

4.3. Feature Selection

The dataset consists of 18 features; they characterize what a typical ADS-B message contains and give more details and understanding of its structure. Table 5 introduces the features and briefly defines what the values represent. We omitted two features in this section (icao24 and callsign); the reason was that these attributes are strings that correlate to the aircraft type, and such information has no impact on detecting or classifying an injected message. Moreover, it could aid with decreasing the margin of error caused by encoding those strings into numerical values before testing.

Table 5. Dataset features.

Feature	Datatype/Value	Definition
Time	Int/No nulls	The time of the last position was reported.
ICAO24	String/No nulls	International Civil Aviation Organization 24, a unique hex address of transponder.
LAT	Float/Nulls exist	Latitude.
LON	Float/Nulls exist	Longitude.
Velocity	Float/Nulls exist	Velocity in m/s.
Heading	Float/Nulls exist	The angle at which an aircraft is moving.
VertRate	Float/Nulls exist	The ascending/descending rate of the aircraft.
Call Sign	String/Nulls exist	The callsign of the aircraft.
On-ground	Bool/No nulls	Indicate if the aircraft is on the ground or not.
SPI	Bool/No nulls	Special purpose identifier confirms the identity of an aircraft.
Squawk	Int/Nulls exist	Code for identification and emergency purposes.
Baroaltitude	Int/Nulls exist	Bar altitude in meters.
Geoaltitude	Int/Nulls exist	Geo altitude in meters.
Lastposupdate	Int/No nulls	Time of the last position update.
Last contact	Int/No nulls	Time of the last signal sent/received.
RSS	Float/Nulls exist	Signal strength.
Doppler	Float/Nulls exist	Any changes in the frequency of transmission.
Label	Float/Nulls exist	Indicates an authentic or malicious ADS-B message.

4.4. Detection and Classification

A model that forecasts the class of supplied data points is called a machine learning classifier model. In supervised learning, test datasets that the machine learning model has never seen before are used to test and evaluate the model after it has been trained with potential attacks using training data. Multiple classifiers were evaluated while creating the ensemble machine learning model to attain the best outcomes. We utilized Jupyter Notebook to test the algorithms mentioned previously in the overview section (random forest, AdaBoost, naive Bayes (NB), neural networks, SVM, and kNN), and the comparison factors are represented as confusion matrices for the top three performing models. The algorithms were tested respectively, without parallelization for both detection and classification.

- **Detection**: the models' first test is performed on the dataset containing only original and malicious ADS-B messages while ignoring the different types of malicious messages. The dataset was balanced, so there was no need for any techniques to balance it, and then we tested the model's ability to detect a fake message without necessarily classifying it into its relevant attack type. The dataset was split into multiple types. The model was tested using different training dataset sizes to achieve the highest results by lowering the amount needed in the training process since our model is efficient, which is important because we are dealing with decisions that could depend on these messages. These decisions could change in a matter of seconds.
- **Classification**: here, we emit authentic messages from the dataset, and we are left with the injected messages; this is done to enable testing of the classification of attack types, which consist of three different types, as discussed above. Unfortunately, the dataset is unbalanced, so we had to test different balancing techniques to maximize our model's efficiency. After balancing the dataset and before further testing of the models' classification, we also split our experiment into different stages to obtain better results with the lowest amount of training needed, hence proposing an efficient model. Finally, it is obvious and trivial that the model will perform better with larger training data. Still, since we aim to achieve efficiency in our model and have a fast response to zero-day attacks on ADS-B messages, we decided to move forward with lower training dataset sizes. We have yet to notice much improvement after 30%, but this still needs further research with a larger data set.

5. Analysis and Discussion

The primary objective of our study is to propose a reliable and efficient model for detecting injection within ADS-B messages. We tested the model to work even with a low-volume dataset and minimal training to be fast and responsive without affecting the detection quality. The adopted dataset is relatively new, was only published in 2022, and has only been used once by academics. The dataset will be tested using a variety of machine learning algorithms to guarantee the detection model performs as well as possible.

This section describes the methods implemented to build our proposed detection model. The section will be divided into four main sub-sections. Section 5.1 presents the environment where the experiment took place. Section 5.2 showcases the model testing process. Section 5.3 contains details of the model evaluation metrics. Section 5.4 presents a comparison with state-of-the-art models. Firstly, Table 6 represents the experimental environment where the model was tested.

Table 6. Experimental environment.

Feature	Value
Operating System	Windows 11 Pro
Processor	AMD Ryzen™ 5 5600X
Frequency	4.6 Ghz
# Cores/# Threads	6 Cores/12 Threads
GPU	RTX 2070 Super Advanced Edition/8 GB GDDR6
Memory/Memory Frequency	16 GB GDDR4/3200Mgz
ML Framework	Scikit-Learn
Language	Python
Tool	Jupyter Notebook

5.1. Model Evaluation

This section presents the performance evaluation results for the proposed detection model based on random forest in various indicators. A confusion matrix is used to confirm the performance of the suggested machine learning model. The confusion matrix uses the following criteria to assess the model's prediction:

- True positives (TP): as anticipated by the model; an injected message is true (injected).
- True negatives (TN): although true, the model anticipated that it would be negative (genuine).
- False positives (FP): although negative, the model anticipated it to be positive (genuine).
- False negatives (FN): although positive, the model predicted it to be negative (injected).

The confusion matrix can be used to build several equations that can be used to understand and assess the performance of the proposed model, including the following ones:

- Accuracy indicates the percentage of real positives and real negatives that the model correctly identified.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \quad (1)$$

- Precision indicates the percentage of real positives for each model's successful predictions. Precision measures how probable a method is to produce accurate results.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad (2)$$

- **Recall** calculates the total number of correct predictions produced over the entire dataset, including correct predictions the model missed. Therefore, a high recall value for an IDS model is desired.

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{3}$$

- F1-score calculates the percentage of correctly predicted events. It is a weighted harmonic mean of recall and precision.

$$\text{F1} - \text{Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{4}$$

- Matthews correlation coefficient (MCC) is one of the best measures to summarize a confusion matrix and the quality of a binary classification. Simply, it is a correlation coefficient between observed and predicted binary classification, and it will only have good results if all four categories inside a confusion matrix have good results.

$$\text{MCC} = \frac{(\text{TP} \times \text{TN}) - (\text{FP} \times \text{FN})}{\sqrt{((\text{TP}+\text{FP})(\text{TP}+\text{FN})(\text{TN}+\text{FP})(\text{TN}+\text{FN}))}} \tag{5}$$

We will use accuracy, F1-score, and Matthews correlation coefficient to compare the findings of this study with the previous related literature since the accuracy statistic is frequently used to assess the effectiveness of machine learning algorithms. Accuracy could be misleading when working with imbalanced datasets because it would predict based on the majority class, but this was addressed using different sampling techniques to remove bias. The F1-score and MCC metrics were used due to their ability to showcase the quality of the results.

5.2. Model Estimation and Optimization

We split the dataset into training and testing datasets using ten different folds and k-fold cross-validation. The k-fold cross-validation technique helps split the dataset into multiple subsets of training and testing, which helps verify a model's performance by choosing different sets at each iteration to serve as training and testing datasets, which helps with utilizing the whole dataset, shuffling the data and making sure to remove any bias. K is the parameter that indicates the number of folds. Figure 3 displays the method of k-cross validation [36].

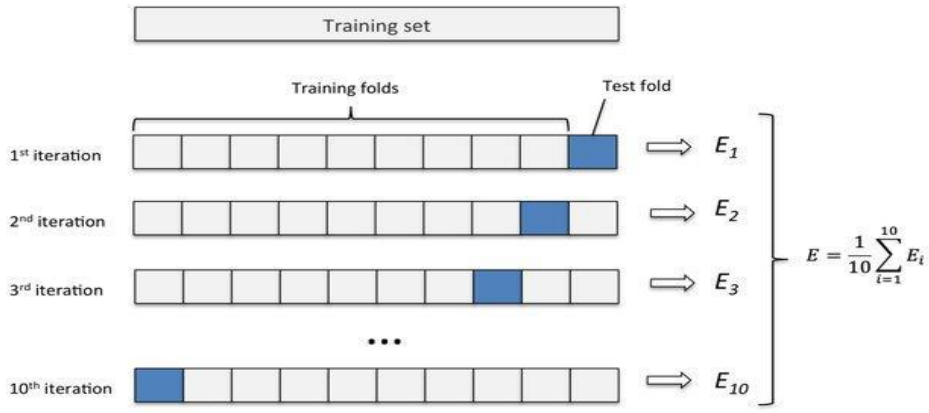


Figure 3. K-cross validation technique [37].

We also performed dimensionality reduction, which is the transformation of data from a high-dimensional space into a low-dimensional space so that the low-dimensional representation has meaningful properties of the originating dataset, ideally close to its

intrinsic dimension; because working with data in a high-dimensional space is usually undesirable and computationally hard, lowering the dimensionality of data could boost the performance of a model.

We used principal component analysis (PCA), which is considered the main linear technique for reducing dimensionality. PCA performs a linear mapping of the data to a lower-dimensional space in such a way that the variance of the data in the low-dimensional representation is maximized.

In principle, this will boost the effectiveness of our model and lower the needed features. PCA was applied, where 16 principal components were identified. Unfortunately, in our case, it caused a loss of information and problems with data interpretation since it uses linear combinations of original variables; this led to a drop in accuracy from 99.41% to 93.01%. Likewise, F1-score also dropped from 99.37% to 92.15%. Lastly, MCC also dropped from 0.988 to 0.888. The model performed better by utilizing all features of the ADS-MI-2022 dataset with no linear combinations or dimensionality reductions. So, we continued testing our model by utilizing all of the features of the ADS-MI-2022 dataset.

5.3. Model Analysis

5.3.1. Binary Classification

A model that forecasts the class of supplied datasets is called a machine learning classifier model. In supervised learning, test datasets that the machine learning model has never seen before are used to test and evaluate the model after it has been trained with potential attacks using training data. Multiple classifiers were tested while the suggested machine learning model was constructed to attain the best outcomes. The best outcome was found between 20–30% training set size. We have decided to go ahead and continue with 30% to have an optimal training set size and to be more robust to unknown attacks, as the model is intended to be as efficient as possible. It has shown some positive effects on the different binary classification results. The top three used ML classifiers evaluated for this work are listed in Table 7. Random forest was utilized to form an ensemble machine learning classifier from the classifiers described since it produced better overall results.

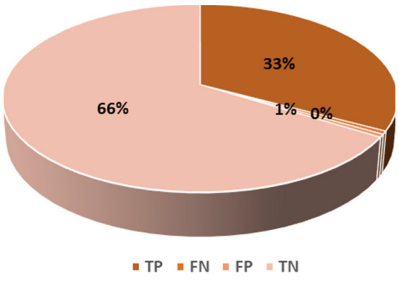
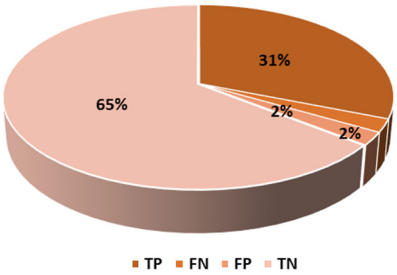
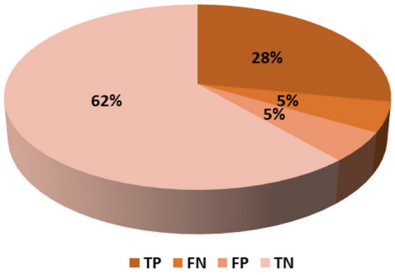
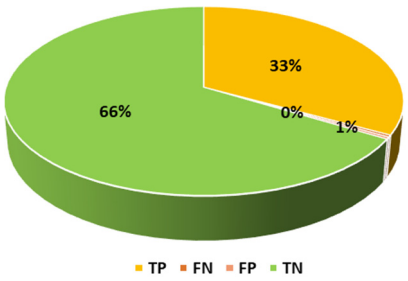
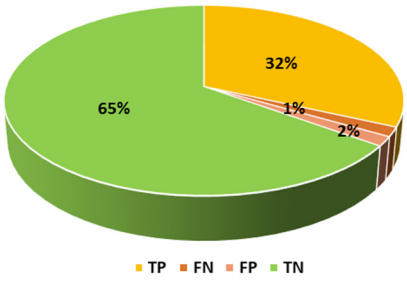
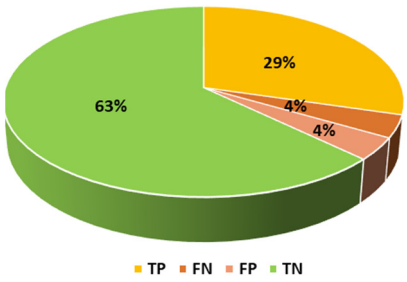
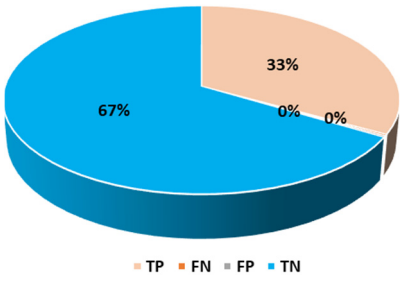
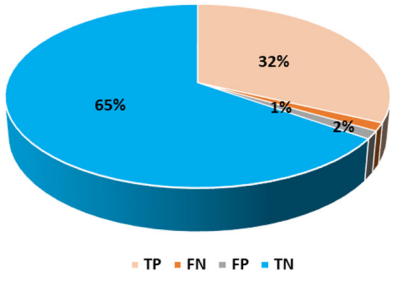
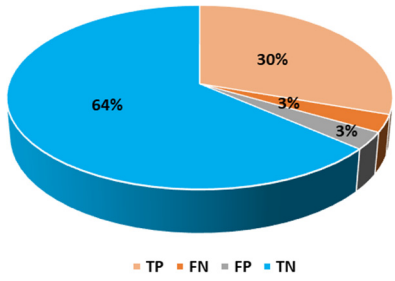
Table 7. Binary classification report.

Algorithm	Accuracy	F1-Score	MCC	Training Time	Prediction Time	Limitation/s
RFC	99.14%	99.14%	0.982	0.57 s	~4 ms	Requires more computational power when the dataset is huge.
MLP	93.45%	93.44%	0.870	4.95 s	~1 ms	It requires a larger dataset and is time-consuming to train.
kNN	96.50%	96.50%	0.930	0.008 s	~1 ms	It is computationally expensive because it stores all the training data and makes slow predictions if the dataset is large.

5.3.2. Multiclass Classification

Authentic messages were removed to train the model on the different types of attacks and improve its accuracy and detection rate. This has resulted in the dataset needing to be more balanced. Thus, we have used three different data sampling models. Since we are proposing an efficient detection model, only results using a training dataset of 10–30% of the whole dataset size will be showcased in this section. Table 8 showcases the top three algorithms and their performance in each train-test split. To conclude this phase, which included training the ML models, it was clear that using 30% of the dataset while it was being oversampled using random oversampling gave us the best results. Therefore, we continued the testing based on those results and used the random forest classifier.

Table 8. Performance of the top three ML models.

RFC	MLPC	kNN
A training set of 10% of the dataset		
		
Accuracy 98.18%	Accuracy 93.87%	Accuracy 84.25%
Precision 97.94%	Precision 93.10%	Precision 82.84%
Recall 97.65%	Recall 93.48%	Recall 85.21%
F1-Score 97.79%	F1-Score 93.29%	F1-Score 83.42%
MCC 0.970	MCC 0.902	MCC 0.762
A training set of 20% of the dataset		
		
Accuracy 98.96%	Accuracy 95.16%	Accuracy 88.61%
Precision 98.86%	Precision 94.43%	Precision 87.26%
Recall 98.70%	Recall 94.97%	Recall 89.75%
F1-Score 98.78%	F1-Score 94.69%	F1-Score 88.09%
MCC 0.983	MCC 0.907	MCC 0.823
A training set of 30% of the dataset		
		
Accuracy 99.41%	Accuracy 96.78%	Accuracy 94.80%
Precision 99.48%	Precision 96.51%	Precision 93.87%
Recall 99.33%	Recall 96.18%	Recall 95.60%
F1-Score 99.37%	F1-Score 96.53%	F1-Score 94.61%
MCC 0.988	MCC 0.925	MCC 0.843

5.4. Discussion

The study's primary goal is to identify binary classification to detect injected ADS-B messages and try to counter path modification, ghost aircraft injection, and velocity drift attacks. By identifying the sort of assault that was launched, we can better prepare our

defenses. Below is a full examination of the findings using the proposed model compared to the related literature.

- The authors in [4] used radio frequency transmitters to attack ADS-B communication channels. They used 36 different ADS-B in message configurations (13 hardware devices and 22 software). They attacked them using BladeRF and HackRF as attacking devices. They achieved 90% accuracy in detecting spoofed messages and showed that their method could help distinguish the different ADS-B messages.
- M. Leonardi et al. [21] proposed a detection model that detects malicious messages by fingerprinting wireless radio signals. Still, even though their proposed method is unfamiliar, only 50 percent of the malicious signals were detected.
- The authors of [38] proposed employing multiple variants of the SVM algorithm to detect and classify attacks on ADS-B messages and then compare the results. They produced the dataset used to test our proposed model.
- N. Li et al. [39] proposed a model utilizing supervised deep learning to detect attacks on ADS-B messages using a fake data generator.

Now, after showcasing the results of the proposed system with a set of different attributes, we produced the following table, Table 6, which compares the proposed model with other existing models in a similar study field. Table 9 compares other state-of-the-art models. The comparison was based on the following performance factors: analysis algorithm, accuracy, F1-score, and Matthews correlation coefficient (MCC).

Table 9. Comparison with some of the related work.

Reference	Analysis Algorithm/Method	Accuracy	F1-Score	MCC
[4]	Multiple machine learning models	90%	92%	-
[21]	Fingerprinting radio signals	50%	-	-
[38]	C-SVM	95.32%	-	-
[39]	Supervised deep learning	-	98.46	-
Our model	Random forest	99.41%	99.37%	0.988

Our results show that it is possible to build precise detection models for this kind of data that can operate across various stages of the life cycle of these messages by relying on the analysis of transmitted signals. The suggested detection model outperformed the majority of those in the literature in terms of results. This section reviewed the experiment results, evaluated the suggested approach, and compared the output with previous pioneering work in this field. The experimental results are outstanding and impactful in the field. However, further improvements can be made to improve the quality of these results.

6. Conclusions and Remarks

Automatic dependent surveillance-broadcast (ADS-B) is considered the future of aviation surveillance and aircraft traffic control. This protocol still shows that it lacks security. Researchers are still trying to improve the ADS-B security aspect to avoid risks such as causing collision avoidance system failure, reporting the wrong status of an aircraft, or even stealing the aircraft. Use of this technology is growing, but the data transmitted are unencrypted and have no actual method to authenticate, tempting hackers to exploit its flaws. We examined most of the prior research in the field for this study.

Firstly, the literature addressed some key issues and vulnerabilities related to ADS-B technology. However, attacks still occur using low-cost equipment supported by technological advancement, creating new obstacles to overcoming these multiple risks.

Secondly, this study aims to suggest a robust and efficient detection model for injected ADS-B messages to aid with the security of air traffic management and offer some answers to some of the challenges associated with identifying those messages and attacks related to them. The unique dataset has just over 11000 injected samples and 11000 benign samples,

totaling 3 MB in size. A set of related features was chosen to identify the existing samples after the study's dataset had to be cleaned up before conducting any research because it obstructed the classification process. This experiment required us to test against six algorithms for better prediction results. We then compared the top three results from the binary classification stage.

Thirdly, the results of our tests came as follows: random forest, multilayer perceptron classifier, and k-nearest neighbor, which, respectively, had an accuracy of 99.14%, 93.45%, and 96.50%. Additionally, random forest outperformed the multilayer perceptron classifier and k-nearest neighbor with an F1-score of 99.14% as opposed to 93.45% and 96.50%, respectively; and with an MCC of 0.982 as opposed to 0.870 and 0.930, respectively.

Then, these models were further tested for multiclass classification, and since the dataset was imbalanced at this stage, three sampling techniques were used. The results came as follows: random forest, multilayer perceptron classifier, and support k-nearest neighbor, respectively, had an accuracy of 99.41%, 96.78%, and 94.80%. Additionally, random forest outperformed the multilayer perceptron classifier and k-nearest neighbor with an F1-score of 99.37% as opposed to 96.53% and 94.61%, respectively; and with an MCC of 0.988 as opposed to 0.925 and 0.843, respectively.

While our research focuses on building and evaluating a standalone IDS to help improve the security of aviation control systems against injected cyberattacks (specifically: path modification, ghost aircraft, and velocity drift), the proposed IDS system needs to be integrated within the current aviation control infrastructures. The proposed sequence of events for implementing the detection system is to fine-tune the approach, integrate it with aviation systems, obtain regulatory approval, and then deploy and train the crew. However, several integration challenges might be faced; assuring data quality, accomplishing real-time processing, and reducing false alarms are among the challenges. Aviation security is improved by integration with current protocols; however, modifications are required for standardization, affordability, and scalability for broad implementation.

Author Contributions: Conceptualization, Q.A.A.-H. and A.A.-T.; methodology, Q.A.A.-H. and A.A.-T.; software, A.A.-T.; validation, Q.A.A.-H.; formal analysis, Q.A.A.-H. and A.A.-T.; investigation Q.A.A.-H. and A.A.-T.; resources, Q.A.A.-H. and A.A.-T.; data curation, Q.A.A.-H.; writing—original draft preparation, Q.A.A.-H. and A.A.-T.; writing—review and editing, Q.A.A.-H. and A.A.-T.; visualization, A.A.-T.; supervision, Q.A.A.-H.; project administration, Q.A.A.-H.; funding acquisition, Q.A.A.-H. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The dataset associated with this research is publicly available through the Mendeley-Elsevier data repository and can be accessed via https://doi.org/10.17632/6fhw732ccz.1 (accessed on 12 November 2023).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. European Union Aviation Safety. Opinion No 01/2020: High-Level Regulatory Framework for the U-Space. Available online: https://www.easa.europa.eu/en/document-library/opinions/opinion-012020 (accessed on 17 December 2023).
2. Strohmeier, M.; Lenders, V.; Martinovic, I. On the Security of the Automatic Dependent Surveillance-Broadcast Protocol. *IEEE Commun. Surv. Tutor.* **2015**, *17*, 1066–1087. [CrossRef]
3. McCallie, D.; Butts, J.; Mills, R. Security analysis of the ADS-B implementation in the next generation air transportation system. *Int. J. Crit. Infrastruct. Prot.* **2011**, *4*, 78–87. [CrossRef]
4. Cretin, A.; Vernotte, A.; Chevrot, A.; Peureux, F.; Legeard, B. Test Data Generation for False Data Injection Attack Testing in Air Traffic Surveillance. In Proceedings of the IEEE International Conference on Software Testing, Verification and Validation Workshops (ICSTW), Porto, Portugal, 24–28 October 2020. [CrossRef]
5. Ho, T.K. Random decision forests. In Proceedings of the 3rd International Conference on Document Analysis and Recognition, Montreal, QC, Canada, 14–16 August 1995.
6. Schapire, R.E. I am explaining Adaboost. In *Empirical Inference*; Springer: Berlin/Heidelberg, Germany, 2015.
7. Vikram, K.; Vijaykumar, B.; Trilochan. Bayes and Naive Bayes Classifier. *arXiv* **2014**, arXiv:1404.0933. [CrossRef]

8. McCulloch, W.S.; Pitts, W. A logical calculus of the ideas immanent in nervous activity. *Bull. Math. Biophys.* **1943**, *5*, 115–133. [CrossRef]
9. Cortes, C.; Vapnik, V. Support-vector networks. *Mach. Learn.* **1995**, *20*, 273–297. [CrossRef]
10. Zhang, Z. Introduction to machine learning: K-nearest neighbors. *Ann. Transl. Med.* **2016**, *4*, 218. [CrossRef] [PubMed]
11. Costin, A.; Francillon, A. Ghost in the air (traffic): On the insecurity of ADS-B protocol and practical attacks on ADS-B devices. *Black Hat USA* **2012**, *1*, 1–12.
12. Leonardi, M.; Strohmeier, M.; Lenders, V. On Jamming Attacks in Crowdsourced Air Traffic Surveillance. *IEEE Aerosp. Electron. Syst. Mag.* **2021**, *36*, 44–54. [CrossRef]
13. Eskilsson, S.; Gustafsson, H.; Khan, S.; Gurtov, A. Demonstrating ADS-B and CPDLC attacks with software-defined radio. In Proceedings of the 2020 Integrated Communications Navigation and Surveillance Conference (ICNS), Herndon, VA, USA, 8–10 September 2020.
14. Leonardi, M.; Piracci, E.; Galati, G. ADS-B vulnerability to low-cost jammers: Risk assessment and possible solutions. In Proceedings of the 2014 Tyrrhenian International Workshop on Digital Communications-Enhanced Surveillance of Aircraft and Vehicles (TIWDC/ESAV), Rome, Italy, 15–16 September 2014.
15. Pearce, N.; Duncan, K.J.; Jonas, B. Signal discrimination and exploitation of ADS-B transmission. In Proceedings of the SoutheastCon 2021, Atlanta, GA, USA, 10–13 March 2021.
16. Schäfer, M.; Lenders, V.; Martinovic, I. Experimental analysis of attacks on next-generation air traffic communication. In Proceedings of the International Conference on Applied Cryptography and Network Security, Banff, AB, Canada, 25–28 June 2013; Springer: Berlin/Heidelberg, Germany, 2013.
17. Manesh, M.R.; Mullins, M.; Foerster, K.; Kaabouch, N. A preliminary effort toward investigating the impacts of ADS-B message injection attack. In Proceedings of the 2018 IEEE Aerospace Conference, Big Sky, MT, USA, 3–10 March 2018. [CrossRef]
18. Haddad, Y.; Orye, E.; Maennel, O. Ghost Injection Attack on Automatic Dependent Surveillance–Broadcast Equipped Drones Impact on Human Behaviour. In Proceedings of the 2021 IEEE Conference on Cognitive and Computational Aspects of Situation Management (CogSIMA), Tallinn, Estonia, 14–22 May 2021. [CrossRef]
19. Strohmeier, M.; Schäfer, M.; Lenders, V.; Martinovic, I. Realities and challenges of next gen air traffic management: The case of ADS-B. *IEEE Commun. Mag.* **2014**, *52*, 111–118. [CrossRef]
20. Odin, A.; Gruneau, M. The ADS-B Protocol and Its' Weaknesses. Ph.D. Thesis, KTH Royal Institute of Technology, Stockholm, Sweden, 2020.
21. Leonardi, M.; Di Gregorio, L.; Di Fausto, D. Air traffic security: Aircraft classification using ADS-B message's phase-pattern. *Aerospace* **2017**, *4*, 51. [CrossRef]
22. Kacem, T.; Wijesekera, D.; Costa, P. ADS-Bsec: A holistic framework to secure ADS-B. *IEEE Trans. Intell. Veh.* **2018**, *3*, 511–521. [CrossRef]
23. Strohmeier, M.; Lenders, V.; Martinovic, I. Intrusion detection for airborne communication using PHY-layer information. In Proceedings of the 12th International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment, Milan, Italy, 9–10 July 2015.
24. Kacem, T.; Wijesekera, D.; Costa, P. Integrity and authenticity of ADS-B broadcasts. In Proceedings of the 2015 IEEE Aerospace Conference, Big Sky, MT, USA, 7–14 March 2015.
25. Strohmeier, M.; Lenders, V.; Martinovic, I. Lightweight location verification in air traffic surveillance networks. In Proceedings of the 1st ACM Workshop on Cyber-Physical System Security, Singapore, 14–17 April 2015.
26. Shen, H.; Liu, K.; Yao, Y.; Wang, J. An ADS-B Anti-counterfeiting System Based on TDOA. In Proceedings of the IEEE International Conference on Signal, Information, and Data Processing (ICSIDP), Chongqing, China, 11–13 December 2019. [CrossRef]
27. Leonardi, M. ADS-B Anomalies and Intrusions Detection by Sensor Clocks Tracking. *IEEE Trans. Aerosp. Electron. Syst.* **2019**, *55*, 2370–2381. [CrossRef]
28. Sciancalepore, S.; Di Pietro, R. SOS: Standard-Compliant and Packet Loss Tolerant Security Framework for ADS-B Communications. *IEEE Trans. Dependable Secur. Comput.* **2021**, *18*, 1681–1698. [CrossRef]
29. Chen, Y.; Zhou, L. Vulnerabilities in ADS-B and Verification Method. In Proceedings of the 2020 IEEE 2nd International Conference on Civil Aviation Safety and Information Technology (ICCASIT), Weihai, China, 14–16 October 2020. [CrossRef]
30. Monteiro, M.; Barreto, A.; Division, R. Detecting malicious ADS-B broadcasts using wide area multilateration. In Proceedings of the 2015 IEEE/AIAA 34th Digital Avionics Systems Conference (DASC), Prague, Czech Republic, 13–17 September 2015.
31. Khandker, S.; Turtiainen, H.; Costin, A.; Hamalainen, T. Cybersecurity Attacks on Software Logic and Error Handling Within ADS-B Implementations: Systematic Testing of Resilience and Countermeasures. *IEEE Trans. Aerosp. Electron. Syst.* **2022**, *58*, 2702–2719. [CrossRef]
32. Kujur, B.; Khanafseh, S.; Pervan, B. Detecting GNSS spoofing of ADS-B-equipped aircraft using INS. In Proceedings of the IEEE/ION Position, Location and Navigation Symposium (PLANS), Portland, OR, USA, 20–23 April 2020. [CrossRef]
33. Ould Slimane, H.; Benouadah, S.; Kaabouch, N. *ADS-B Message Injection Attacks Dataset*; Mendeley Data, V1; University of North Dakota: Grand Forks, ND, USA, 2022. [CrossRef]
34. Chawla, N.V.; Bowyer, K.W.; Hall, L.O.; Kegelmeyer, W.P. SMOTE: Synthetic minority over-sampling technique. *J. Artif. Intell. Res.* **2002**, *16*, 321–357. [CrossRef]

35. Abu Al-Haija, Q. Top-Down Machine Learning-Based Architecture for Cyberattacks Identification and Classification in IoT Communication Networks. *Front. Big Data* **2022**, *4*, 782902. [CrossRef] [PubMed]
36. Abu Al-Haija, Q.; Krichen, M.; Abu Elhaija, W. Machine-Learning-Based Darknet Traffic Detection System for IoT Applications. *Electronics* **2022**, *11*, 556. [CrossRef]
37. Ashfaque, J.M.; Iqbal, A. Introduction to Support Vector Machines and Kernel Methods. 2019. Available online: https://www.academia.edu/38782451/ (accessed on 12 April 2019).
38. Slimane, H.O.; Benouadah, S.; Al Shamaileh, K.; Devabhaktuni, V.; Kaabouch, N. ADS-B Message Injection Attack on UAVs: Assessment of SVM-based Detection Techniques. In Proceedings of the IEEE International Conference on Electro-Information Technology (EIT), Mankato, MN, USA, 19–21 May 2022. [CrossRef]
39. Li, N.; Lin, L.; Li, F. ADS-B Anomaly Data Detection Using SVDD-based LSTM Encoder-Decoder Algorithm. In Proceedings of the IEEE 3rd International Conference on Civil Aviation Safety and Information Technology (ICCASIT), Changsha, China, 20–22 October 2021. [CrossRef]