

## Metaheuristic Algorithms for Feature Selection in Intrusion Detection Systems: A Systematic Review

Yashar Pourardbilkhah <sup>1</sup>, Mirsaeid Hosseini Shirvani <sup>2\*</sup>, Homayun Motameni <sup>3</sup>

1. PhD student, Department of Computer Engineering, Sari Branch, Islamic Azad University, Sari, Iran.  
[yashar.uni2010@gmail.com](mailto:yashar.uni2010@gmail.com)
2. Assistant Professor, Department of Computer Engineering, Sari Branch, Islamic Azad University, Sari, Iran.  
[mirsaeid\\_hosseini@iausari.ac.ir](mailto:mirsaeid_hosseini@iausari.ac.ir)
3. Professor, Department of Computer Engineering, Sari Branch, Islamic Azad University, Sari, Iran.  
[motameni@iausari.ac.ir](mailto:motameni@iausari.ac.ir)

### Abstract

**Introduction:** Considering the increasing importance of network security in complex and dynamic environments, intrusion detection systems (IDS) play a very important role in identifying and dealing with security threats. However, the presence of a large amount of data in networks affects the efficiency of intrusion detection systems. Feature selection as a critical step in data preprocessing can help improve the accuracy, speed, and efficiency of these systems. This article deals with the systematic review of feature selection methods based on meta-heuristic algorithms in intrusion detection systems in the cloud computing environment.

**Method:** The methodology of this article includes a comprehensive review of the research conducted in the field of feature selection for IDS. In this review, meta-heuristic algorithms such as genetic algorithm, particle swarm optimization (PSO), bee colony optimization, bat algorithm, and other nature-inspired optimization methods are thoroughly reviewed. These algorithms are chosen due to their ability to search a large space of features and identify the best combinations to improve the performance of IDSs.

**Evaluation:** In this paper, detailed comparisons have been made between different algorithms in terms of performance criteria such as detection accuracy, false alarm rate, processing time and the number of selected features. Also, different data sets that have been used in this field have been discussed to evaluate the efficiency of each of the methods in different conditions.

**Challenge:** In addition, key challenges in this field have been identified and analyzed. These challenges include things such as high computational complexity, the problem of processing overhead in cloud environments, the balance between accuracy and detection speed, and the problem of feature interference. Also, research gaps in this field that require further research have been identified.

**Keywords:** Feature selection, meta-heuristic algorithm, intrusion detection system, cloud computing, optimization, network security.

## الگوریتم‌های فراابتکاری برای انتخاب ویژگی در سیستم‌های تشخیص نفوذ: یک بررسی سیستماتیک

دوره چهارم، پاییز ۱۴۰۲  
شماره سوم، صص: ۶۳-۹۲

تاریخ دریافت: ۱۴۰۲/۰۴/۳۱  
تاریخ پذیرش: ۱۴۰۲/۰۶/۱۶

یاشار پوراردبیل خواه<sup>۱</sup>، میرسعید حسینی شیروانی<sup>۲\*</sup>، همایون مؤتمنی<sup>۳</sup>

۱. دانشجوی دکتری، گروه مهندسی کامپیوتر، دانشکده فنی مهندسی، دانشگاه آزاد اسلامی، ساری، ایران  
[yashar.uni2010@gmail.com](mailto:yashar.uni2010@gmail.com)
۲. استادیار، گروه مهندسی کامپیوتر، واحد ساری، دانشگاه آزاد اسلامی، ساری، ایران  
[mirsaeid\\_hosseini@iausari.ac.ir](mailto:mirsaeid_hosseini@iausari.ac.ir)
۳. استاد، گروه مهندسی کامپیوتر، واحد ساری، دانشگاه آزاد اسلامی، ساری، ایران  
[motameni@iausari.ac.ir](mailto:motameni@iausari.ac.ir)

### چکیده:

با توجه به اهمیت روزافزون امنیت شبکه در محیط‌های پیچیده و پویا، سیستم‌های تشخیص نفوذ (IDS) نقش بسیار مهمی در شناسایی و مقابله با تهدیدات امنیتی ایفای می‌کنند. با این حال، وجود حجم زیادی از داده‌ها در شبکه‌ها، کارایی سیستم‌های تشخیص نفوذ را تحت تأثیر قرار می‌دهد. انتخاب ویژگی به عنوان یک مرحله حیاتی در پیش‌پردازش داده‌ها می‌تواند به بهبود دقت، سرعت و کارایی این سیستم‌ها کمک کند. این مقاله به بررسی سیستماتیک روش‌های انتخاب ویژگی مبتنی بر الگوریتم‌های فراابتکاری در سیستم‌های تشخیص نفوذ در محیط رایانش ابری می‌پردازد. روش کار شامل مرور جامعی بر تحقیقات انجام‌شده در حوزه انتخاب ویژگی برای IDS ها است. در این بررسی، الگوریتم‌های فراابتکاری مانند الگوریتم ژنتیک، بهینه‌سازی ازدحام ذرات (PSO)، بهینه‌سازی کلونی زنبور، الگوریتم خفاش، و سایر روش‌های بهینه‌سازی الهام‌گرفته از طبیعت، به‌طور کامل بررسی شده‌اند. این الگوریتم‌ها به دلیل توانایی‌شان در جستجوی فضای بزرگ ویژگی‌ها و شناسایی بهترین ترکیب‌ها برای بهبود عملکرد IDS ها، انتخاب شده‌اند. در این مقاله، مقایسه‌های دقیقی بین الگوریتم‌های مختلف از نظر معیارهای عملکردی همچون دقت تشخیص، نرخ هشدار غلط، زمان پردازش و تعداد ویژگی‌های انتخاب‌شده انجام شده است. همچنین، مجموعه داده‌های مختلفی در این زمینه، مورد بحث و بررسی قرار گرفته‌اند تا کارایی هر یک از روش‌ها در شرایط مختلف ارزیابی شود. علاوه بر این، چالش‌های کلیدی در این حوزه مورد شناسایی و تحلیل قرار گرفته‌اند. این چالش‌ها شامل مواردی همچون پیچیدگی محاسباتی بالا، مسئله سربرار پردازشی در محیط‌های ابری، تعادل بین دقت و سرعت تشخیص، و مشکل تداخل ویژگی‌ها هستند. همچنین، شکاف‌های تحقیقاتی موجود در این زمینه که نیازمند تحقیقات بیشتر است، شناسایی شده‌اند. نتایج این بررسی نشان می‌دهد که هر یک از الگوریتم‌های فراابتکاری دارای نقاط قوت و ضعف خاص خود هستند و انتخاب بهترین روش بستگی به نیازمندی‌ها و شرایط خاص سیستم دارد. این مقاله با ارائه توصیه‌هایی برای تحقیقات آینده، به‌عنوان یک راهنمای جامع برای پژوهشگران و توسعه‌دهندگان سیستم‌های تشخیص نفوذ در محیط‌های ابری مطرح می‌شود.

**واژه‌های کلیدی:** انتخاب ویژگی، الگوریتم فراابتکاری، سیستم تشخیص نفوذ، رایانش ابری، بهینه‌سازی، امنیت شبکه

## ۱. مقدمه

امروزه، سازمان‌ها به منظور کاهش هزینه‌های سراسری کسب و کار خود و دسترسی سریع به منابع ذخیره‌سازی مناسب برای داده‌ها، از سیاست برون‌سپاری منابع بهره‌می‌برند. در این راستا، ابرها به‌عنوان گزینه‌ای اصلی برای برون‌سپاری مورد توجه سازمان‌ها قرار می‌گیرند. ابرها با ارائه ویژگی‌هایی چون ذخیره‌سازی اطلاعات و دسترسی از راه دور، برای استفاده از اینترنت طراحی شده‌اند. رایانش ابری از سه لایه انتزاعی تشکیل شده است که شامل لایه زیرساخت، لایه بستر، و لایه کاربردی می‌باشد. دو لایه اول مربوط به ماشین‌های مجازی (VM) و سیستم‌عامل است، در حالی که آخرین لایه شامل برنامه‌های کاربردی ارائه شده توسط ابر است [۱]. خدمات رایانش ابری به کاربران در سه مدل مختلف ارائه می‌شود: زیرساخت به‌عنوان یک سرویس (IaaS)، بستر به‌عنوان یک سرویس (PaaS)، و نرم‌افزار به‌عنوان یک سرویس (SaaS). در مدل زیرساخت به‌عنوان یک سرویس، مدیران می‌توانند سیستم‌عامل، منابع سخت‌افزاری، و ذخیره‌سازی را کنترل یا حتی برنامه‌های کاربردی و اجزای شبکه را مستقر کنند. مدل PaaS هدف خود را در توسعه‌دهندگان قرار می‌دهد تا برنامه‌های ایجاد شده توسط کاربران را در فضای ابری مستقر کنند، در حالی که مدل SaaS کاربران را قادر می‌سازد تا برنامه‌های ارائه‌دهنده را اجرا کنند. معماری ابر از دو جزء متوالی تشکیل شده است: جزء اول که front-end نام دارد و توسط کاربر قابل مشاهده است (مثلاً شبکه یا کامپیوتر کاربر و برنامه‌ای که برای دسترسی به ابر از طریق رابط کاربری استفاده می‌کند)، و جزء دوم که backend نام دارد و خود ابر است و از سرورهای مختلفی تشکیل شده است [۲].

این اطلاعات نشان می‌دهد که استفاده از ابر به‌عنوان یک راهکار مناسب برای سازمان‌هاست، اما در عین حال، با چالش‌های امنیتی نیز روبرو می‌باشد که نیازمند توجه و پیشرفت در زمینه سیستم‌های امنیتی هوشمند می‌باشد. علی‌رغم مزایای بی‌شماری که ابر ارائه می‌دهد، موانع و تهدیداتی نیز در این حوزه وجود دارد که می‌تواند موجب نگرانی کسب و کارها و سازمان‌های مختلف شود. یکی از این موانع مسائل امنیتی است که به‌عنوان یکی از مهمترین مسائل در استفاده از ابر مطرح می‌شود. بنابراین، ابرها نیازمند یک سیستم امنیتی هوشمند می‌باشند که به‌طور خودکار قادر به تشخیص نفوذ باشد و از آسیب‌پذیری‌های مختلف محافظت کند. این امر نیازمند توسعه و اجرای استانداردهای امنیتی فراگیر و مؤثر در سطح ابر و همچنین آموزش و آگاهی کاربران از روش‌های بهبود امنیت و حفاظت از داده‌ها در محیط ابر می‌باشد. همچنین، مدیران و اپراتورهای ابر باید به‌طور دائمی نگرانی‌ها و تهدیدات امنیتی را بررسی و برای پیشگیری از حملات و حفظ امنیت داده‌ها تدابیر مناسبی را اتخاذ کنند.

تشخیص نفوذ (Intrusion Detection) به‌عنوان یکی از مهم‌ترین ابزارها در حفاظت از امنیت شبکه‌ها و سیستم‌های کامپیوتری مورد توجه قرار گرفته است. هدف اصلی این سیستم‌ها، تشخیص و پیش‌بینی حملات و نفوذهای ناخواسته به شبکه‌ها و سیستم‌های اطلاعاتی

می‌باشد. تشخیص نفوذ به دو صورت نقش‌های مختلف انجام می‌پذیرد: تشخیص نفوذ مبتنی بر امضا (Signature-based IDS) و تشخیص نفوذ مبتنی بر رفتار (Behavior-based IDS) [۳]. در روش تشخیص نفوذ مبتنی بر امضا، الگوریتم‌ها و قوانینی براساس الگوهای مشخصی از حملات و نفوذهای قبلی تعیین می‌شوند و در صورت تطابق الگو، هشدار صادر می‌شود. اما در روش تشخیص نفوذ مبتنی بر رفتار، سیستم بر اساس عملکرد و رفتار غیرمعمولی کاربران و شبکه، حملات را شناسایی می‌کند. سیستم‌های تشخیص نفوذ تاکنون توانسته‌اند به طور موفقیت‌آمیزی به تشخیص حملاتی چون حملات (Denial of Service) (DoS)، حملات با استفاده از نفوذگرهای مخرب (Malware)، حملات شنود (Sniffing) و حملات نفوذ به سیستم‌های کنترل صنعتی (ICS) پرداخته [۳] و مدیران شبکه را در جلوگیری از این حملات یاری‌نموده‌اند. اما با پیشرفت فناوری و رویکردهای حمله‌کنندگان، نیاز به ارتقاء و بهبود مداوم سیستم‌های تشخیص نفوذ، احساس می‌شود.

سیستم تشخیص نفوذ در ابر شامل تعداد متغیرها و ویژگی‌های زیادی است که پردازش آن‌ها زمانبر است و از آنجا که سرعت تشخیص نفوذ در ابر بسیار بااهمیت است، نیاز به کاهش متغیرهای نامربوط و زائد است. متغیرها و ویژگی‌های نامربوط و غیرضروری، موجب کاهش سرعت و دقت سیستم تشخیص نفوذ می‌شوند بنابراین چندین تکنیک برای رسیدگی به مشکل کاهش متغیرها، توسعه داده شده است. کاهش متغیرها و ویژگی‌ها را انتخاب ویژگی گویند. انتخاب ویژگی به درک داده‌ها، کاهش نیاز محاسباتی، کاهش اثر بد ابعاد و بهبود عملکرد تشخیص کمک می‌کند [۳]. در این مقاله برخی از روش‌های موجود در تحقیقات گذشته بررسی خواهد شد که از تکنیک‌های مختلفی برای یافتن زیرمجموعه‌ای از ویژگی‌ها استفاده می‌کنند که موجب بهبود عملکرد کلی تشخیص نفوذ می‌شود.

روش انتخاب ویژگی، در واقع کاهش تعداد ویژگی‌های ورودی است. این روش، زیرمجموعه‌ای از متغیرهای مرتبط را انتخاب می‌کند تا یک مدل تشخیص صحیح با کمترین داده و بهترین دقت داشته باشیم. ویژگی‌های کمتر، پیچیدگی مدل را کاهش می‌دهند که موجب ارائه یک مدل ساده‌تر و قابل فهم‌تر می‌شود. روش‌های انتخاب ویژگی موجب شناسایی و حذف ویژگی‌های غیرضروری، غیرمرتبط و اضافی می‌شوند، این ویژگی‌ها تأثیری در دقت مدل پیش‌بینی ندارند و حتی ممکن است باعث کاهش دقت مدل شوند. بنابراین انتخاب ویژگی دارای سه هدف مهم است: بهبود کارایی سیستم تشخیص، افزایش سرعت تشخیص و ارائه درک بهتری از فرایند تولید داده. با استفاده از تکنیک‌های انتخاب ویژگی، می‌توان بینشی در مورد فرآیند داده‌ها به دست آورد و نیاز محاسباتی و دقت پیش‌بینی را بهبود بخشید [۳]. یکی از روش‌های انتخاب ویژگی، استفاده از الگوریتم‌های فراابتکاری است.

الگوریتم فراابتکاری (Metaheuristic Algorithm) یک روش محاسباتی است که برای حل مسائل بهینه‌سازی و تصمیم‌گیری استفاده-

الگوریتم‌های فراابتکاری برای تشخیص نفوذ پرداخته‌اند [۸-۱۵]. ادبیات [۸، ۱۳] انواع تکنیک‌های انتخاب ویژگی برای سیستم‌های تشخیص نفوذ را بررسی کرده‌اند. برخی از ادبیات سیستم‌های تشخیص نفوذ مبتنی بر الگوریتم‌های فراابتکاری را مورد بررسی قرار دادند [۹، ۱۲، ۱۰] همچنین ادبیات [۱۴ و ۱۵ و ۱۶] به بررسی الگوریتم‌های فراابتکاری در انتخاب ویژگی پرداختند.

Balasaraswathi و همکاران [۱۱] هر دو روش الگوریتم‌های فراابتکاری بهینه‌سازی زیستی و غیر زیستی را با استفاده از تکنیک‌های انتخاب ویژگی در تشخیص نفوذ شبکه‌های کامپیوتری به کار گرفتند. Javier Maldonado و همکارانش [۸] مروری بر پیشرفت‌های اخیر در تکنیک‌های انتخاب ویژگی روش‌های بسته‌بند (Wrapper) برای تشخیص نفوذ ارائه می‌کند. هدف آن طبقه‌بندی و ارزیابی رویکردهای مختلف برای تشخیص و طبقه‌بندی حمله است. این مقاله چالش‌های تشخیص نفوذ را به دلیل پیچیدگی و تنوع حملات و همچنین اهمیت انتخاب ویژگی‌های کلیدی برای تشخیص مؤثر مورد بحث قرار می‌دهد. این بررسی یک طبقه‌بندی classification را پیشنهاد می‌کند، سناریوهای تجربی را تحلیل می‌کند، و چالش‌ها و مزایای تکنیک‌های موجود را برجسته می‌کند. همچنین جنبه‌های کلیدی خود را با سایر بررسی‌ها در ادبیات مقایسه می‌کند و یک نمای سازمان‌یافته از برنامه‌های انتخاب ویژگی روش‌های بسته‌بند (Wrapper) در تشخیص حمله ارائه می‌دهد.

Shubhkirti Sharma و همکاران [۹] بررسی منظمی از الگوریتم‌های بهینه‌سازی چندهدفه برای تشخیص نفوذ در شبکه‌های اینترنت اشیاء (IoT) انجام داده‌اند. آن‌ها یک مقاله مروری سیستماتیک ارائه دادند که در آن با جمع‌آوری و تحلیل مقالات و تحقیقات پیشین در این حوزه، الگوریتم‌های بهینه‌سازی چندهدفه برای تشخیص نفوذ در شبکه‌های IoT را بررسی نمودند. یک بررسی جامع از پژوهش‌های پیشین در این زمینه ارائه شده است که شامل انواع الگوریتم‌های بهینه‌سازی چندهدفه، روش‌های ارزیابی، و نتایج حاصل از این تحقیقات می‌شود. اهداف متعدد برای تشخیص نفوذ در شبکه‌های IoT، مانند دقت طبقه‌بندی، نرخ تشخیص، دقت، نرخ هشدار نادرست، ویژگی و تعداد ویژگی‌ها، با فرمول‌بندی‌های ریاضی آن‌ها مورد بحث قرار می‌گیرند. ، نقاط قوت و ضعف این الگوریتم‌ها و کاربردهای آن‌ها در شبکه‌های IoT بررسی می‌شود و پیشنهادهای برای جهت‌گیری تحقیقات آینده در این زمینه ارائه می‌شود.

Dukka Karun Kumar Reddy و همکارانش [۱۰] از روش‌های سیستماتیک برای جمع‌آوری و تحلیل مقالات علمی و تحقیقات پیشین در زمینه استفاده از الگوریتم‌های هوش گروهی برای تشخیص نفوذ استفاده کرده‌اند. ابتدا یک بررسی جامع از پژوهش‌ها و مقالات پیشین در این زمینه ارائه شده است که شامل انواع الگوریتم‌های هوش گروهی مورد استفاده، روش‌های ارزیابی، و نتایج حاصل از این تحقیقات می‌شود. چالش‌ها شامل انتخاب ویژگی، انتخاب مدل ML و فقدان مجموعه

می‌شود. این الگوریتم‌ها معمولاً از رویکردهای مبتنی بر طبیعت الهام گرفته شده و قدرتمندترین روش‌های موجود برای حل مسائل پیچیده بهینه‌سازی محسوب می‌شوند [۴]. این الگوریتم‌ها برای حل مسائل به صورت تکاملی و هماهنگ استفاده می‌شوند. هدف اصلی این الگوریتم‌ها، یافتن راه‌حل‌های بهینه و تعاملی برای مسائل پیچیده است. الگوریتم‌های فراابتکاری از تکرار، تعامل بین عامل‌ها، و انطباق به محیط استفاده می‌کنند تا به یافتن راه‌حل‌های بهتر و بهینه‌تر برای مسئله مورد نظر بپردازند. این الگوریتم‌ها شامل مجموعه‌ای از قوانین و روش‌های مختلفی می‌باشند که با توجه به نوع مسئله و شرایط محیطی، مناسب‌ترین و بهینه‌ترین راه‌حل را ارائه می‌دهند [۵].

جستجوی تصادفی، یادگیری و بهبود، قابلیت تطبیق از ویژگی‌های کلیدی الگوریتم‌های فراابتکاری هستند. از مزایای الگوریتم‌های فراابتکاری این است که می‌توانند مسائلی را حل کنند که برای الگوریتم‌های بهینه‌سازی سنتی دشوار یا غیرقابل حل هستند. همچنین این الگوریتم‌ها می‌توانند به طور کارآمد در فضای جستجوی بزرگ و نامحدود به دنبال راه‌حل بگردند و برای کارکردن نیازی به محاسبات پیچیده تابع هدف ندارند، که در مورد بسیاری از توابع غیرخطی، محاسبات، دشوار یا غیرممکن است. استفاده از الگوریتم‌های فراابتکاری در مسائل بهینه‌سازی، بهبود عملکرد سیستم‌ها و فرآیندها را به دنبال دارد. این الگوریتم‌ها علاوه بر کاربردهای بهینه‌سازی، در حوزه‌های مختلفی از جمله مسائل تصمیم‌گیری چندمعیاره (MCDM)، بهینه‌سازی مسائل شبکه‌های مخابراتی، و مسائل برنامه‌ریزی ریاضی نیز کاربرد دارند [۶].

الگوریتم‌های فراابتکاری می‌توانند به عنوان ابزارهای قدرتمندی برای انتخاب ویژگی در طیف گسترده‌ای از مسائل به کار گرفته شوند. الگوریتم‌های فراابتکاری می‌توانند مسائلی را حل کنند که برای روش‌های سنتی انتخاب ویژگی، مانند روش‌های فیلتر (filter methods) دشوار یا غیرقابل حل هستند. این الگوریتم‌ها می‌توانند با روش‌های سنتی انتخاب ویژگی ترکیب شوند تا نتایج بهتری حاصل شود [۷]. الگوریتم‌های فراابتکاری در طیف گسترده‌ای از مسائل انتخاب ویژگی به کار گرفته می‌شوند، از جمله: مسائل مربوط به پزشکی (تشخیص بیماری، پیش‌بینی بیماری، و انتخاب ژن‌های مرتبط با بیماری)، مسائل مربوط به مالی: (شخصی‌تقلب، پیش‌بینی بازار سهام، و مدیریت سبد سهام) و مسائل مربوط به مهندسی (طراحی سازه‌ها، فرآیندهای تولید، و سیستم‌های حمل و نقل).

### ۱.۱. خلاصه‌ای از مقالات مروری گذشته

به منظور آماده‌سازی این مطالعه، نتایج جستجوی اولیه شامل ۴۳۹ مقاله درباره تشخیص نفوذ از سال‌های ۲۰۱۸ تا ۲۰۲۴ می‌باشد. تعداد ۳۱۴ مقاله به دلیل عدم تناسب با محیط ابری حذف شدند. ۸۰ مطالعه دیگر به دلیل عدم ارتباط قوی با روش‌های انتخاب ویژگی از لیست مقالات مورد مطالعه حذف شدند. در نهایت ۴۵ مقاله مناسب برای این پژوهش انتخاب شد. تعداد کمی از ادبیات فعلی به انتخاب ویژگی با استفاده از

داده‌های برجسب‌گذاری شده‌است. علاوه بر این، پرداختن به مجموعه داده‌های نامتعادل و پیمایش بین رفتارهای عادی و غیرعادی چالش‌هایی را در ID ایجاد می‌کند.

Veeran Ranganathan Balasaraswathi و همکارانش [۱۱] تکنیک‌های انتخاب ویژگی برای سیستم‌های تشخیص نفوذ (IDS) را برای بهبود عملکرد با حذف ویژگی‌های نامربوط با استفاده از الگوریتم‌های بهینه‌سازی غیرالهام‌گرفته و زیستی مورد بحث قرار می‌دهد. انواع مختلف IDS مانند NIDS و HIDS به همراه مراحل عملکرد IDS توضیح داده شده‌است. همچنین یک نمای کلی از FS، از جمله روش‌های بسته‌بند (Wrapper)، فیلتر و ترکیبی ارائه می‌کند و نحوه استفاده از الگوریتم‌های الهام‌گرفته از زیستی مانند الگوریتم‌های ژنتیک و برنامه‌ریزی ژنتیک را برای انتخاب ویژگی مورد بحث قرار می‌دهد. مراحل دخیل در FS با استفاده از الگوریتم‌های الهام‌گرفته از زیستی مشخص شده‌اند و مزایا و کاربردهای الگوریتم‌های تکاملی را در حل مؤثر مسائل پیچیده برجسته می‌کند.

Rafika Saadouni و همکارانش [۱۲] مروری بر ادبیات سیستماتیک در مورد تکنیک‌های الهام‌گرفته از زیستی و یادگیری ماشین برای سیستم‌های تشخیص نفوذ در شبکه‌های IoT انجام دادند. آن‌ها دریافتند که ادغام الگوریتم‌های الهام‌گرفته از زیست با رویکردهای ML و DL می‌تواند عملکرد IDS را افزایش دهد و امنیت شبکه اینترنت اشیا را تقویت کند. این مطالعه ۲۵ مقاله انتخاب شده را مقایسه کرد و چالش‌ها و جهت‌گیری‌های آینده در این زمینه را شناسایی کرد. توصیه‌هایی برای تحریک تحقیقات و نوآوری بیشتر در امنیت اینترنت اشیا ارائه شد.

C. Kalimuthan و همکاران [۱۳] اهمیت امنیت شبکه و نیاز به تکنیک‌های جدید برای تشخیص نفوذ با استفاده از یادگیری ماشین را مورد بحث قرار می‌دهند. انواع مختلف حملات شبکه، سیستم‌های تشخیص نفوذ، روش‌های انتخاب ویژگی و طبقه‌بندی‌کننده‌های یادگیری ماشین را تشریح می‌کند. موضوعات کلیدی شامل الگوریتم Naive Bayes، Support Vector Machine، Decision Tree و K-Nearest Neighbor است.

PRACHI AGRAWAL و همکارانش [۱۴] این مقاله مروری بر ادبیات انتخاب ویژگی با استفاده از الگوریتم‌های فراابتکاری از سال ۲۰۰۹ تا ۲۰۱۹ ارائه می‌کند. آن‌ها با استفاده از روش‌های سیستماتیک، ادبیات موجود در زمینه الگوریتم‌های فراابتکاری برای انتخاب ویژگی‌ها را بررسی و مورد ارزیابی قرار داده‌اند. مقاله شامل بررسی الگوریتم‌های مختلفی از جمله الگوریتم مورچه، الگوریتم ژنتیک، الگوریتم گرگ‌های خاکستری، و سایر الگوریتم‌های فراابتکاری است.

هدف اصلی این بررسی، بررسی عملکرد و کارایی این الگوریتم‌ها در فرایند انتخاب ویژگی‌ها، برای مسائل مختلف مانند تشخیص نفوذ و دیگر کاربردهای یادگیری ماشین است.

Mohamed Ziad Ali و همکاران [۱۵] چالش‌های انتخاب ویژگی در یادگیری ماشین و کاربردهای تشخیص الگو را مورد بحث قرار می‌دهند

و بر روش‌های بهینه‌سازی فراابتکاری الهام‌گرفته از طبیعت تمرکز می‌کنند. تجزیه و تحلیل دقیقی از رویکردها و استراتژی‌های مختلف در انتخاب ویژگی و همچنین تکنیک‌های مختلف بهینه‌سازی فراابتکاری پیشرفته ارائه و مزایا و معایب آن‌ها را برجسته می‌کنند. همچنین به مسائل و چالش‌های فعلی در انتخاب ویژگی و بهینه‌سازی فراابتکاری می‌پردازد و توصیه‌هایی برای تحقیقات آینده در این زمینه ارائه می‌دهد. TIN H. PHAM و همکارانش [۱۶] الگوریتم‌های انتخاب ویژگی الهام‌گرفته از زیستی و کاربردهای آن‌ها در زمینه‌های مختلف را مورد بحث قرار می‌دهند. آن‌ها ۳۸ مطالعه انتخاب شده را خلاصه می‌کنند و ۲۱ الگوریتم الهام‌گرفته از زیست را بررسی می‌کنند و استفاده از الگوریتم‌های مبتنی بر ازدحام و روش‌های طبقه‌بندی نظارت شده را برجسته می‌کنند. آن‌ها پیشنهاد می‌کنند که تحقیقات آینده باید بر روی استفاده از انتخاب ویژگی‌های الهام‌گرفته از زیستی در زمینه‌های مختلف و کاوش تکنیک‌های بهبود، تمرکز کند. تکنیک‌های کاهش ابعاد، از جمله روش‌های خطی و غیرخطی، نیز مورد بحث قرار می‌گیرند. در مجموع، تمام کارهای مروری گذشته، منابع خوبی هستند، اما جامع و کامل نیستند. مروری [۸] بر اساس مجموعه داده‌های استفاده شده، دسته‌بندی شده‌است و معیارهای ارزیابی و توابع هدف مورد بحث قرار گرفته‌است. مطالعه [۹] به الگوریتم‌های بهینه‌سازی چندهدفه برای سیستم‌های تشخیص نفوذ توجه دارد و کمتر به انتخاب ویژگی پرداخته‌است. توجه مطالعه [۱۰] بیشتر به الگوریتم‌های هوش جمعی برای سیستم‌های تشخیص نفوذ است و در بخش کوتاهی به انتخاب ویژگی توجه کرده‌است. مطالعه [۱۱] بر روی یک مجموعه داده تمرکز کرده‌است. مطالعه [۱۲] از تکنیک‌های انتخاب ویژگی مبتنی بر الگوریتم طبیعی و یادگیری ماشین برای استفاده در مسئله تشخیص نفوذ بهره‌برده‌اند. مطالعه [۱۳] به بررسی انتخاب ویژگی بر اساس یادگیری ماشین پرداخته‌است و الگوریتم‌های فراابتکاری را در نظر نگرفته‌است و فقط الگوریتم ژنتیک را بررسی کرده‌است. مطالعات [۱۴ و ۱۵ و ۱۶] دسته‌بندی خوبی درباره الگوریتم‌های فراابتکاری ارائه کرده‌است اما در مورد تشخیص نفوذ نیست.

## ۲.۱. دامنه بحث

تمرکز اصلی مرور ادبیات ما بر شناسایی تکنیک انتخاب ویژگی مبتنی بر فراابتکاری در سیستم‌های تشخیص نفوذ از سال ۲۰۱۹ تا ۲۰۲۴ است. هدف ما بررسی توابع هدف، مجموعه داده‌ها و معیارهای ارزیابی در ادبیات گذشته است. تا جایی که ما می‌دانیم هیچ مروری، توابع هدف و معیارهای ارزیابی در زمینه تشخیص نفوذ را به‌طور جامع بررسی نکرده‌است. این کار بر روی انتخاب ویژگی مبتنی بر فراابتکاری تمرکز دارد و تأثیر آن را در حوزه تشخیص نفوذ بررسی می‌کند. ما در ادامه، مجموعه داده‌های مختلفی که در مطالعات آزمایش شده‌اند را معرفی می‌کنیم و همچنین تعداد ویژگی‌های انتخاب شده هر مجموعه داده را مشخص می‌کنیم. سپس معیارهای ارزیابی مطالعه شده و فراوانی آن‌ها در مطالعات

گذشته را مروری کنیم. توابع هدف مطالعات و پارامترهای مختلف محبوب را شرح می‌دهیم.

### ۳.۱. نوآوری مقاله

این مطالعه مروری حاضر، الگوریتم‌های فراابتکاری که برای انتخاب ویژگی در سیستم‌های تشخیص نفوذ استفاده شده‌اند را شناسایی می‌کند. این پژوهش، برای تحقیقات آینده مفید خواهد بود زیرا تصویر کاملی از مطالعات قبلی را همراه نقاط قوت و ضعف ارائه می‌دهد. بنابراین، نوآوری مقاله به صورت زیر آورده شده است:

- بررسی جامع از الگوریتم‌های فراابتکاری را ارائه می‌کنیم که برای انتخاب ویژگی در سیستم‌های تشخیص نفوذ استفاده شده‌اند.
- انواع مجموعه داده‌های به‌کاررفته را بررسی می‌کنیم تا متداول‌ترین مجموعه داده‌ها مشخص شود.
- توابع هدف استفاده شده در کارهای گذشته را تجزیه و تحلیل و بررسی می‌کنیم تا بینش روشنی از کاربردهای آن ارائه کنیم.
- معیارهای ارزیابی مطالعات قبلی را ارائه می‌کنیم.
- چالش‌ها و پیشنهاد‌های آینده در حوزه انتخاب ویژگی در تشخیص نفوذ مورد بحث قرار می‌گیرد.

مقاله به شرح زیر سازماندهی شده است: بخش ۲ به بررسی سیستم‌های رایانش ابری، سیستم تشخیص نفوذ و مسئله انتخاب ویژگی پرداخته است. بخش ۳، روش بررسی ادبیات شرح داده می‌شود و بخش ۴، مسئله انتخاب ویژگی در سیستم‌های تشخیص نفوذ را طبقه‌بندی می‌کند. بخش ۵، تأثیر الگوریتم‌های فراابتکاری در مسئله انتخاب ویژگی برای تشخیص نفوذ را بررسی می‌کند و مجموعه داده‌ها، معیارهای ارزیابی و توابع هدف مطالعات را بررسی می‌کند. بخش ۶ چالش‌ها و شکاف‌های تحقیقاتی این حوزه بیان می‌شود. در نهایت، بخش ۷، نتیجه‌گیری و پیشنهاد‌های آینده برای مسئله انتخاب ویژگی در سیستم‌های تشخیص نفوذ آمده است.

## ۲. بررسی ادبیات

### 1.2. رایانش ابری

با پیشرفت فناوری اطلاعات نیاز به انجام کارهای محاسباتی در همه جا و همه زمان به وجود آمده است. همچنین نیاز به این هست که افراد بتوانند کارهای محاسباتی سنگین خود را بدون داشتن سخت‌افزارها و نرم‌افزارهای گران‌قیمت انجام دهند. رایانش ابری آخرین پاسخ فناوری به این نیازها بوده است. عموماً مصرف‌کننده‌های رایانش ابری مالک زیرساخت فیزیکی ابر نیستند، بلکه برای اجتناب از هزینه سرمایه‌ای آن را از عرضه‌کنندگان شخص ثالث اجاره می‌کنند. آن‌ها منابع را در قالب سرویس مصرف‌کرده و تنها هزینه منابعی که به‌کار می‌برند را می‌پردازند. این در حالی است که سایر گونه‌های عرضه‌کنندگان بر مبنای اشتراک سرویس‌های خود را عرضه می‌کنند. به اشتراک گذاردن قدرت رایانشی

مصرف‌شدنی و ناملموس میان چند مستاجر می‌تواند باعث بهبود نرخ بهره‌وری شود؛ زیرا با این شیوه دیگر سرویس‌دهنده‌ها بدون دلیل بیکار نمی‌مانند و همین امر سبب می‌شود در عین حال که سرعت تولید و توسعه برنامه‌های کاربردی افزایش یابد، هزینه‌ها به میزان قابل‌توجهی کاهش یابند. اهدافی که تکنولوژی رایانش ابری در صنعت فناوری اطلاعات دنبال می‌کند به شرح زیر می‌باشد [۱۷]: مقیاس‌پذیری، در دسترس بودن، قابلیت اطمینان، امنیت، قابلیت انعطاف‌پذیری، قابلیت سرویس‌دهی و کارایی است. امنیت یکی از اهداف مهم رایانش ابری است که نگرانی بزرگی در تمام جنبه‌های ساخت رایانش ابری است. بیشتر این جنبه‌ها منحصر برای تنظیمات ابر نیست بلکه داده‌ها بدون در نظر گرفتن محل ذخیره‌سازی‌شان در معرض خطر قرار دارند. بنابراین امنیت رایانش ابری شامل تمام موضوعات امنیت محاسباتی از جمله طراحی معماری، به حداقل رساندن سطوح حمله و حفاظت در برابر ابزارهای مخرب و اعمال کنترل دسترسی می‌باشد. جنبه‌های خاص امنیت رایانش ابری شامل موارد ذیل است:

1. ابر به‌طور معمول منابع را به اشتراک می‌گذارد و شرکا ممکن است جزو هکرها باشند.
  2. داده‌های مبتنی بر ابر عمدتاً توسط پروتکل‌های ناامن و رابط‌های برنامه‌های کاربردی (API) در سراسر شبکه عمومی در دسترس هستند.
  3. داده‌ها در ابر در معرض خطر از بین رفتن یا تغییر توسط ارائه‌دهنده ابر قرار دارد.
  4. داده در ابر می‌تواند توسط ارائه‌دهنده ابر، پیمانکاران فرعی و کارکنان آن در دسترس قرار گیرند.
- تضمین امنیت دسترسی به داده در شبکه عمومی اگرچه برای رایانش ابری ضروری است ولی درحقیقت یک چالش است و این چالش در واقع این است که داده مشتریان در ابر فاش شود و در دسترس عموم قرار گیرد.

### ۲.۲. سیستم تشخیص نفوذ

عملکردهای اصلی یک سیستم تشخیص نفوذ نظارت بر رویدادهای رخ داده در یک سیستم یا شبکه رایانه‌ای، تجزیه و تحلیل رویدادهای سیستم، شناسایی فعالیت‌های مشکوک و اعلام هشدار در صورت شناسایی نفوذ است. یک سیستم تشخیص نفوذ معمولی را می‌توان به سه جزء عملکردی [۱۸] تقسیم کرد: منبع اطلاعات، موتور تجزیه و تحلیل و تصمیم‌گیرنده. این سه جزء را می‌توان بر روی یک کامپیوتر یا در چند کامپیوتر مختلف اعمال کرد. بنابراین کل سیستم تشخیص نفوذ می‌تواند یک سیستم میزبان در یک کامپیوتر یا یک سیستم توزیع شده در یک شبکه محلی یا حتی در سراسر اینترنت باشد.

اولین جزء از سیستم تشخیص نفوذ، منبع اطلاعات است که برای نظارت بر رویدادهای رخ داده در یک سیستم یا شبکه کامپیوتری استفاده می‌شود. در طول مدت نظارت، منبع اطلاعات جریانی از سوابق رویداد را برای تجزیه و تحلیل فراهم می‌کند. این مؤلفه به‌عنوان یک مولد رویداد کاری‌کند. منابع داده‌های مختلف را نظارت می‌کند و داده‌های

رویداد را تولید می‌کند که آبا به خوبی قالب‌بندی شده‌اند و برای یک موتور تجزیه و تحلیل برای انجام تجزیه و تحلیل بیشتر مناسب هستند. منابع داده را براساس اینکه داده‌های آن‌ها از کجا آمده‌است، می‌توان به سه دسته تقسیم کرد: دسته اول منبع داده مربوط به سیستم عامل‌ها مانند تماس‌های سیستم و گزارش‌های سیستم است. مانند مجموعه داده UNM [۱۹]. دسته دوم مانیتورهای ترافیک شبکه هستند که بسته‌های خام شبکه را تولید می‌کنند. یکی از نمونه‌های این نوع داده‌ها، مجموعه داده‌های KDD است [۱۹]. دسته سوم جمع‌آوری داده‌های برنامه‌های مختلف در حال اجرا در سیستم‌های شبکه یا محیط ابری است. جزء دوم، موتور تجزیه و تحلیل بخش کلیدی سیستم تشخیص نفوذ است. یک سیستم تشخیص نفوذ برای یافتن علائم نفوذ به موتور آنالیز متکی است. تمام تکنیک‌های هوش مصنوعی را می‌توان برای این جزء اعمال کرد. موتور تجزیه و تحلیل، اطلاعاتی را که از منبع اطلاعات به‌دستی می‌آید، فیلتر می‌کند، و هرگونه داده نامربوط در اطلاعات را حذف می‌کند و در نتیجه فعالیت‌های مشکوک را شناسایی می‌کند. موتور تجزیه و تحلیل، معمولاً از یک پایگاه داده سیاست تشخیص، برای تجزیه و تحلیل استفاده می‌کند. بسته به روش‌ها و تکنیک‌های مختلف تشخیص نفوذ، امضاهای حمله، پروفایل‌های رفتار عادی و پارامترهای لازم (مانند آستانه‌ها) در پایگاه داده سیاست تشخیص وجود دارد.

آخرین مؤلفه، تصمیم‌گیرنده است که قوانینی را برای نتایج موتور تحلیل اعمال می‌کند و تصمیم می‌گیرد که براساس نتایج موتور تجزیه و تحلیل چه واکنش‌هایی باید انجام شود. دلیل اصلی استفاده از تصمیم‌گیرنده افزایش قابلیت استفاده از سیستم تشخیص نفوذ است.

### ۳.۲ انتخاب ویژگی

یک ویژگی یک خاصیت قابل اندازه‌گیری منحصر به فرد از فرآیند مشاهده شده است [۲۰]. ویژگی‌ها پایه بسیاری از عملیات داده کاوی و یادگیری ماشین هستند. در واقع با استفاده از مجموعه‌ای از ویژگی‌ها، هر الگوریتم یادگیری ماشین می‌تواند طبقه‌بندی را انجام دهد. تعداد ویژگی‌ها در مسایل مختلف متفاوت است و در بعضی از مسایل تعداد آن‌ها از ده‌ها به صدها متغیر یا ویژگی گسترش یافته‌است بنابراین نیاز به کاهش متغیرهای نامربوط و زائد است. چندین تکنیک برای رسیدگی به این مشکل توسعه داده شده‌است.

انتخاب ویژگی (حذف متغیر) به درک داده‌ها، کاهش نیاز محاسباتی، کاهش اثر بد ابعاد و بهبود عملکرد پیش‌بینی کمک می‌کند [۲۰]. تمرکز انتخاب ویژگی زیرمجموعه‌ای از متغیرها از ورودی است که می‌تواند به‌طور مؤثر داده‌های ورودی را توصیف کند و در عین حال اثرات ناشی از نویز یا متغیرهای نامربوط را کاهش دهد و همچنان نتایج پیش‌بینی خوبی ارائه دهد [۲۱]. انتخاب ویژگی تعداد کل ویژگی‌ها را به حداقل می‌رساند و فقط ویژگی‌های کارآمد را بر اساس ورودی ارائه شده با کاهش داده‌های نویز انتخاب می‌کند، که به شناسایی سریع نتیجه کمک می‌کند. در مجموعه داده‌هایی که تعداد ویژگی بسیار زیاد است تعدادی از ویژگی‌ها با دیگر ویژگی همبستگی بالایی دارند و وقتی دو

ویژگی کاملاً همبسته هستند، وجود تنها یک ویژگی برای توصیف داده‌ها کافی است. بنابراین ویژگی‌های وابسته مانند یک نویز برای ماشین‌های یادگیری عمل می‌کنند. پس می‌توان با حذف آن‌ها، میزان داده‌ها را کاهش داد که منجر به بهبود عملکرد ماشین‌های یادگیری شود. در برخی برنامه‌ها، متغیرهایی که هیچ ارتباطی با کلاس‌ها ندارند، به عنوان نویز خالص عمل می‌کنند، ممکن است اختلالی در پیش‌بینی‌کننده ایجاد کنند و عملکرد طبقه‌بندی را کاهش دهند. این امر می‌تواند زمانی اتفاق بیفتد که اطلاعات کافی در مورد فرآیند مورد مطالعه وجود نداشته باشد [۲۰]. با استفاده از تکنیک‌های انتخاب ویژگی، می‌توانیم بینشی در مورد فرآیند به‌دست آوریم و می‌توانیم نیاز محاسباتی و دقت پیش‌بینی را بهبود بخشیم [۲۰].

روش‌های مختلفی در انتخاب ویژگی وجود دارد. روش‌های اصلی در انتخاب ویژگی عبارتند از: انتخاب ویژگی مبتنی بر فیلتر، انتخاب ویژگی مبتنی بر روش‌های بسته‌بند (Wrapper)، روش‌های تعبیه شده (Embedded methods).

فرآیند روش فیلتر معمولاً از تکنیک رتبه‌بندی استفاده می‌کند. برای امتیازدهی به متغیرها از معیار رتبه‌بندی مناسب و برای حذف متغیرهای زیر آستانه از آستانه استفاده می‌شود. روش‌های رتبه‌بندی، روش‌های فیلتر هستند، زیرا قبل از طبقه‌بندی برای فیلتر کردن متغیرهای کمتر مرتبط، به کار می‌روند [۲۰]. در فیلترها (Filters) میزان اهمیت و تفکیک‌پذیری ویژگی‌ها جداگانه بررسی می‌شوند و طبق رویکرد الگوریتم مورد نظر به تک‌تک ویژگی‌ها امتیازی داده می‌شود که این امتیاز میزان تفکیک‌پذیری و اهمیت ویژگی را مشخص می‌کند. در روش‌های فیلتر، مدل (طبقه‌بند) در جریان انتخاب ویژگی نیست که این امر هم مزیت و هم عیب این روش‌ها محسوب می‌شود. مزیت این رویکرد این است که زمان اجرای الگوریتم بسیار پایین است و در مدت زمان بسیار کوتاه بدون اینکه مدل در جریان کار باشد، به ویژگی‌ها امتیاز داده می‌شود و از بین ویژگی‌ها یک تعداد ویژگی بسته به نیاز مسأله، انتخاب می‌شوند. عیب این رویکرد این است که چون مدل در جریان کار نیست، ممکن است ویژگی‌های مناسب برای مدل انتخاب نشود. ایراد دوم روش‌های فیلتر این است که ارتباط بین ویژگی‌ها در نظر گرفته نمی‌شود، و همین ممکن است باعث انتخاب ویژگی‌هایی شود که به تنهایی خوب عمل کنند، ولی کنار هم عالی عمل نکنند. هدف اصلی ما این است که ویژگی‌های انتخاب‌شوند که کنار هم عملکرد طبقه‌بندی یا رگرسیون را بالا ببرند، ولی در این رویکرد ممکن است ویژگی‌های انتخاب‌شده ترکیب خوبی تشکیل ندهند. با این حال روش‌های فیلتر، روش‌های بسیار مهمی، مخصوصاً در تعیین اهمیت ویژگی‌ها، هستند، زیرا که با هزینه زمانی بسیار پایین می‌توانند مشخص کنند که یک ویژگی مناسب است یا نه.

روش‌های بسته‌بند (Wrapper)، مدل را در جریان انتخاب ویژگی قرار می‌دهند و ارتباط بین ویژگی‌ها را در نظر می‌گیرند. این روش‌ها سعی بر آن دارند که بهترین ترکیب از بین ویژگی‌های موجود انتخاب کنند. مدل روش‌های بسته‌بند (Wrapper) از الگوریتم‌های یادگیری ماشین

<p>- این روش خاصیت چند خطی را حذف نمی کند به طور متناوب باید با مکانیسم دیگری حذف شود.</p>	
<p><b>مدل بسته‌بند</b></p> <p>- این روش دقت مدل را برای تخصیص رتبه یا امتیاز به هر ویژگی نشان می‌دهد.</p> <p>- این روش از وابستگی بین ویژگی‌ها برای تولید زیر مجموعه ویژگی پیروی می‌کند.</p> <p>- برای تولید زیر مجموعه، ویژگی طبقه‌بندی کننده چندین بار اجرامی شود.</p> <p>- این روش برای تولید زیر مجموعه ویژگی‌ها با در نظر گرفتن مدل یادگیری استفاده می‌شود.</p> <p>- این روش مفیدترین ویژگی‌ها را برای تولید زیر مجموعه ویژگی انتخاب می‌کند.</p> <p>- از استنباط مدل یادگیری قبلی این روش تصمیم می‌گیرد که آیا ویژگی را از زیر مجموعه ویژگی اضافه یا حذف کند.</p> <p>- این روش از یک طبقه‌بندی خاص برای ارزیابی کیفیت ویژگی‌های انتخاب شده استفاده می‌کند.</p>	
<p><b>مدل تعبیه شده</b></p> <p>- فرآیند FS در مرحله آموزش ایجاد می‌شود و این روش زیر مجموعه ویژگی را برای الگوریتم در حال آموزش ارزیابی می‌کند.</p> <p>- این روش از ویژگی مفید بودن ویژگی‌ها برای تولید زیر مجموعه ویژگی‌ها استفاده می‌کند.</p> <p>- این روش از مکانیسم یادگیری برای فضای جستجو استفاده می‌کند.</p> <p>- در این روش از تجمیع مزایای روش فیلتر و لفاف استفاده می‌شود.</p> <p>- این رویکرد وابستگی بین ویژگی‌ها را بسیار موثر محاسبه می‌کند. با توجه به در نظر گرفتن طبقه‌بندی کننده، این روش ویژگی‌های مربوطه را انتخاب می‌کند.</p>	

بهترین روش (به صورت مطلق) برای انتخاب ویژگی وجود ندارد و از این رو تلاش‌های محققان بر انتخاب روشی متمرکز شده که برای یک مسأله مشخص بهتر عمل می‌کند. روش‌های گوناگونی برای مواجهه با مجموعه داده‌های کلان مقیاس وجود دارد که اهمیت انتخاب ویژگی در آن‌ها واقعیتی غیرقابل انکار است، زیرا منجر به کمینه کردن زمان آموزش و حافظه تخصیصی با حفظ صحت نتایج می‌شود.

#### ۴.۲. الگوریتم‌های فراابتکاری

الگوریتم‌های فراابتکاری، مجموعه‌ای از روش‌های بهینه‌سازی تصادفی هستند که برای حل مسائل پیچیده که با روش‌های سنتی به‌سختی حل می‌شوند، به‌کار می‌روند. این الگوریتم‌ها از الهام‌گیری از فرایندهای طبیعی مانند تکامل، رفتار اجتماعی حشرات و فیزیک ذرات برای یافتن راه‌حل‌های بهینه یا نزدیک بهینه استفاده می‌کنند [۲۲]. الگوریتم‌های

برای تولید زیرمجموعه ویژگی به‌عنوان بخشی از تابع ارزیابی ویژگی استفاده می‌کند. یعنی از زیرمجموعه ویژگی‌ها استفاده می‌کند و مدل را آموزش می‌دهد [۲۱]. زیرمجموعه‌های ویژگی براساس الگوریتم‌های استقرایی انتخاب می‌شوند. این زیرمجموعه ویژگی انتخاب شده، دقت مدل آموزشی را تخمین می‌زند. بسته به دقت اندازه‌گیری در مرحله قبل، روش، تصمیم می‌گیرد که آیا یک ویژگی را از زیر مجموعه انتخاب شده، اضافه یا حذف کند. به همین دلیل، روش بسته‌بند (Wrapper) از نظر محاسباتی پیچیده‌تر است. روش بسته‌بند (Wrapper) مدل را در جریان انتخاب ویژگی قرار می‌دهند لذا زمان‌بر است و امکان دارد زمانی که تعداد ویژگی‌ها زیاد شود، هزینه زمانی بالایی داشته‌باشد. اما چون از لحاظ رویکرد انتخاب ویژگی بسیار توانمند است، در عمل بسیار استفاده می‌شود.

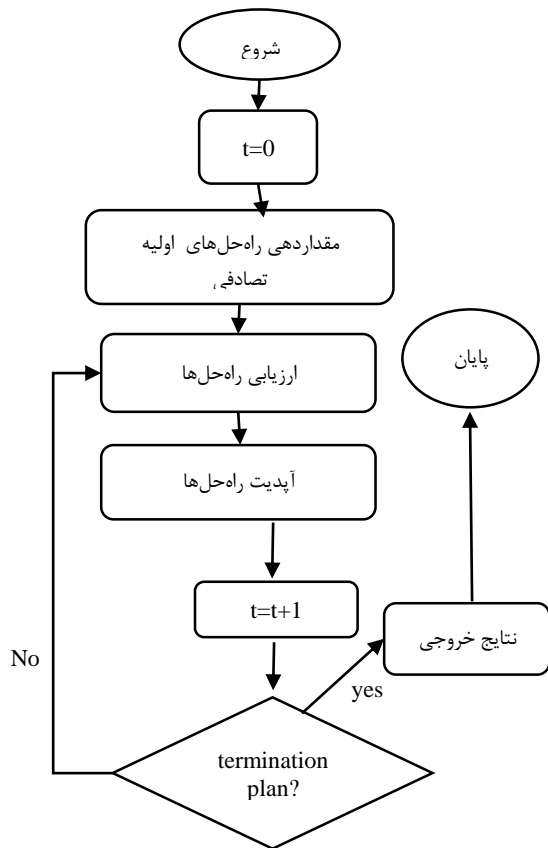
روش‌های توکار (Embedded) انتخاب ویژگی را در فرآیند آموزش انجام می‌دهند و معمولاً برای ماشین‌های یادگیری خاصی مورد استفاده قرار می‌گیرند. این روش به گونه‌ای عمل می‌کند که بهترین ویژگی‌ها در طول فرآیند یادگیری انتخاب می‌شوند. ترکیب انتخاب ویژگی در طول فرآیند یادگیری مزایایی در بهبود هزینه محاسباتی، دقت طبقه‌بندی دارد و همچنین از آموزش مدل در هر بار اضافه شدن ویژگی جدید جلوگیری می‌کند [۹]. روش Embedded بر پیچیدگی محاسباتی غلبه می‌کند. در این روش انتخاب ویژگی مناسب و یادگیری مدل به طور همزمان انجام می‌شود و ویژگی‌ها در مرحله آموزش مدل انتخاب می‌شوند. به همین دلیل هزینه محاسباتی این روش در مقایسه با روش بسته‌بند (Wrapper) به‌طور قطعی کمتر است. در این روش، هر بار که یک ویژگی جدید پیدامی‌شود، نیاز به آموزش مدل نیست.

**جدول ۳: نقش فیلتر، بسته‌بندی و مدل تعبیه شده در فرآیند انتخاب ویژگی**

روش انتخاب ویژگی	نقش در فرآیند انتخاب ویژگی
<p><b>مدل فیلتر</b></p> <p>- این روش معیارهای آماری مختلفی را برای تخصیص رتبه یا امتیاز به هر ویژگی انجام می‌دهد.</p> <p>- این روش زیر مجموعه‌های ویژگی را قبل از مدل یادگیری انتخاب می‌کند فرآیند انتخاب یک زیر مجموعه ویژگی تنها یک بار در مکانیسم کل مدل یادگیری ساختمان انجام می‌شود.</p> <p>- این روش برای ایجاد زیر مجموعه ویژگی از ویژگی مربوط به ویژگی پیروی می‌کند.</p> <p>- فرآیند تولید زیر مجموعه ویژگی مستقل از هر الگوریتم ML است.</p> <p>- این روش از ویژگی ذاتی ویژگی‌ها پیروی می‌کند. همچنین از وابستگی بین ویژگی‌ها برای تولید زیر مجموعه ویژگی‌ها غفلت می‌کند.</p>	



۲. مجموعه راه‌حل‌ها را به صورت تصادفی مقداردهی اولیه کنید.
۳. هریک از راه‌حل‌ها را ارزیابی کنید.
۴. راه‌حل‌ها را آپدیت کنید.
۵. یک واحد  $t$  اضافه کنید.
۶. مراحل ۳ تا ۵ را تا زمانی که یک طرح خاتمه (termination plan) برآورده شده است، ادامه دهید.



شکل ۱: فلوچارت الگوریتم‌های فراابتکاری

هر الگوریتم فراابتکاری طی چندین مرحله به جواب بهینه می‌رسد. در مرحله اول، شمارنده  $t$  مقدار صفر می‌گیرد تا تعداد تکرار الگوریتم را بشمارد. در مرحله ۲، مجموعه راه‌حل‌های مسئله به طور تصادفی و دیگر پارامترهای الگوریتم فراابتکاری مقداردهی اولیه می‌شوند. روش تولید راه‌حل‌های اولیه، نقش مهمی در عملکرد الگوریتم‌های فراابتکاری دارد. بنابراین روش‌های متفاوتی برای مقداردهی راه‌حل‌های اولیه وجود دارد. در مرحله ۳، با استفاده از تابع هدف، ارزش هر راه‌حل به دست می‌آید که با توجه به مقدار ارزش، بهترین راه‌حل مشخص می‌شود. ارزش تابع هدف، معیاری برای ارزیابی عملکرد خوب الگوریتم است [۲۵، ۲۷]. در مرحله ۴، با توجه به راه‌حل‌های قبلی بسته به نوع الگوریتم فراابتکاری، راه‌حل‌های جدید ارائه می‌شود. سپس در انتهای حلقه، شمارنده آن یک واحد اضافه می‌شود. در مرحله ۶ چنانچه شرط پایانی حلقه (که معمولاً آستانه دقت یا تعداد مراحل تکرار الگوریتم فراابتکاری است) برآورده شود نتایج به خروجی ارسال شده و گرنه دوباره باید الگوریتم تکرار شود. برای انتخاب ویژگی با الگوریتم‌های فراابتکاری باید برای هر عامل جستجو، به تعداد ویژگی‌های مجموعه داده، ویژگی در نظر گرفت.

فراابتکاری از جستجوی تصادفی برای کاوش در فضای جستجو و یافتن راه‌حل‌های بالقوه استفاده می‌کند. این الگوریتم‌ها مکرر راه‌حل‌های موجود را بهبود می‌بخشند تا به راه حل بهینه ممکن برسند و البته الگوریتم‌های فراابتکاری تضمین نمی‌کنند که به بهترین راه‌حل ممکن دست پیدا کنند، اما می‌توانند راه‌حل‌های با کیفیت بالا را در زمان معقول پیدا کنند.

الگوریتم‌های فراابتکاری به سه دسته کلی تقسیم می‌شوند: الگوریتم‌های تکاملی: این الگوریتم‌ها از فرایند تکامل الهام می‌گیرند و از جهش، انتخاب و تقاطع برای ایجاد راه‌حل‌های جدید استفاده می‌کنند مانند الگوریتم ژنتیک (Genetic Algorithm) و استراتژی تکامل الگوریتم‌ها از یک جمعیت از راه‌حل‌ها استفاده می‌کنند و مکرر آن‌ها را بر اساس تابع هدفشان بهبود می‌بخشند مانند بهینه‌سازی ازدحام ذرات (Particle Swarm Optimization)، بهینه‌سازی کلنی مورچه‌ها (Colony Optimization)، بهینه‌سازی کلنی زنبورها (Bee Colony Optimization). الگوریتم‌های مبتنی بر فیزیک: این الگوریتم‌ها از مفاهیم فیزیکی مانند شبیه‌سازی تبرید و جستجوی تصادفی شبیه‌سازی شده برای یافتن راه‌حل‌های بهینه استفاده می‌کنند [۲۵]. یک موضوع مشترک که در تمام فراابتکاری‌ها دیده می‌شود، تعادل بین اکتشاف (Exploration) و بهره‌برداری (Exploitation) است.

اکتشاف، به این معنی است که عامل‌ها باید در فضای مسئله جستجو کنند تا راه‌حل‌های ممکن مسئله را بیابند [۲۶]. این جنبه به الگوریتم فراابتکاری یک رفتار جستجوی جهانی می‌دهد. بهره‌برداری به این اشاره دارد که عامل‌ها چگونه می‌توانند از اطلاعات موجود از راه‌حل‌های تکرارهای قبلی، برای یافتن راه‌حل بهینه استفاده کنند [۲۶]. چنین تشدید به فراابتکاری ویژگی جستجوی محلی می‌دهد.

الگوریتم‌های فراابتکاری، در بسیاری از مسائل کاربرد دارند، از جمله مسائل بهینه‌سازی مانند زمان بندی، مسیریابی و غیره، مسائل پیش‌بینی یا طبقه‌بندی داده‌ها و مسائل مهندسی مانند طراحی سیستم‌های قدرت و شبکه‌های حمل و نقل. از دیگر کاربردهای الگوریتم‌های فراابتکاری، انتخاب ویژگی است [۲۷]. به عنوان مثال، می‌توان از آن‌ها برای جستجوی زیرمجموعه‌ای از ویژگی‌ها که منجر به بهترین دقت مدل می‌شوند، استفاده کرد. یکی از مزایای استفاده از الگوریتم‌های فراابتکاری برای انتخاب ویژگی این است که می‌توان از آن‌ها برای حل مسائل با تعداد زیادی ویژگی استفاده کرد. علاوه بر این، الگوریتم‌های فراابتکاری می‌توانند همزمان چندین ویژگی را انتخاب کنند، که می‌تواند به یافتن تعاملات بین ویژگی‌ها کمک کند.

شکل ۱ فلوچارت الگوریتم‌های فراابتکاری را نشان می‌دهد. اکثر الگوریتم‌های فراابتکاری، عملیات مشابهی را دنبال می‌کنند و بنابراین می‌توانند در یک چارچوب عمومی مشترک تعریف شوند که در زیر آمده است:

۱. شمارنده تکرار  $t$  را برابر با ۰ مقداردهی کنید.

## ۵.۲. انتخاب ویژگی مبتنی بر الگوریتم‌های فراابتکاری

انتخاب ویژگی فرآیندی است برای انتخاب زیرمجموعه بهینه ویژگی‌ها از یک مجموعه ویژگی بزرگ برای بهبود دقت طبقه‌بندی، عملکرد و هزینه استخراج ویژگی‌ها [۲۹]. در این مقاله در مورد روش‌های انتخاب ویژگی (فیلتر، روش بسته‌بند (Wrapper)، روش تعبیه‌شده (Embedded)) صحبت خواهد شد. در این بخش، الگوریتم‌های استفاده شده در مقالات انتخابی و میزان تأثیر هر یک را بررسی و سعی می‌کنیم به سه سوال پیشتر مطرح شده پاسخ دهیم.

پاسخ سوال اول: الگوریتم‌های تکاملی (Evolutionary Algorithms) دسته‌ای از الگوریتم‌های بهینه‌سازی مبتنی بر اصول تکامل زیستی و ژنتیک هستند [۳۰]. این الگوریتم‌ها از مفاهیمی مانند انتخاب طبیعی، جهش، ترکیب و بازتولید برای پیدا کردن راه‌حل‌های بهینه استفاده می‌کنند. مهمترین الگوریتم‌های تکاملی عبارتند از: الگوریتم ژنتیک و الگوریتم استراتژی‌های تکاملی. الگوریتم ژنتیک (Genetic Algorithm - GA): یک روش بهینه‌سازی مبتنی بر فرآیندهای تکاملی زیستی است که از مفاهیم انتخاب طبیعی، جهش، ترکیب و بازتولید برای جستجوی راه‌حل‌های بهینه استفاده می‌کند. این الگوریتم با شروع از یک جمعیت اولیه از کروموزوم‌ها (راه‌حل‌های ممکن)، در هر نسل با استفاده از عملگرهای ژنتیکی مانند انتخاب (که به راه‌حل‌های بهتر شانس بیشتری برای بازتولید می‌دهد)، ترکیب (ادغام دو راه‌حل برای تولید راه‌حل‌های جدید) و جهش (ایجاد تغییرات تصادفی برای معرفی تنوع) به تدریج به سمت بهبود راه‌حل‌ها حرکت می‌کند. یکی از مزایای اصلی الگوریتم ژنتیک، توانایی آن در فرار از بهینه‌های محلی و جستجوی فضای گسترده‌ای از راه‌حل‌ها است. این الگوریتم به دلیل انعطاف‌پذیری بالا و قابلیت موازی‌سازی، در مسائل مختلفی مانند بهینه‌سازی توابع پیچیده، برنامه‌ریزی و طراحی مهندسی استفاده می‌شود. الگوریتم استراتژی‌های تکاملی (Evolution Strategies - ES) یکی از روش‌های قدرتمند بهینه‌سازی مبتنی بر اصول تکاملی زیستی است که به‌ویژه برای مسائل بهینه‌سازی پیوسته مناسب است. این الگوریتم با شروع از یک جمعیت اولیه از راه‌حل‌ها، از فرآیندهای انتخاب، جهش و بازتولید برای بهبود تدریجی راه‌حل‌ها استفاده می‌کند. در هر نسل، راه‌حل‌های بهتر برای تولید نسل بعد انتخاب می‌شوند و از توزیع‌های آماری برای ایجاد جهش‌های کوچک و بزرگ در پارامترهای راه‌حل‌ها استفاده می‌شود. برخلاف برخی دیگر از الگوریتم‌های تکاملی، ES معمولاً از عملگرهای ترکیب (جفت‌گیری) استفاده نمی‌کند و بیشتر بر روی تغییرات و جهش‌های تصادفی تمرکز دارد. یکی از مزایای اصلی استراتژی‌های تکاملی، کارایی بالا در بهینه‌سازی مسائل با پارامترهای واقعی و پیوسته است. این الگوریتم به دلیل انعطاف‌پذیری در انتخاب توزیع‌های جهش و توانایی تطبیق با شرایط مختلف، می‌تواند به راه‌حل‌های با کیفیت بالا دست یابد. همچنین، ES به دلیل ساختار ساده و قابلیت موازی‌سازی، در کاربردهای صنعتی و مهندسی مانند طراحی مکانیکی، بهینه‌سازی فرآیندها و تنظیم پارامترهای سیستم‌های پیچیده استفاده می‌شود.

مقادیر هر ویژگی نیز باینری می‌باشد یعنی هر ویژگی مقدار ۰ یا ۱ می‌گیرند. مثلاً اگر بخواهیم برای انتخاب ویژگی مجموعه داده‌ای با ۲۴ ویژگی از الگوریتم ژنتیک استفاده کنیم باید برای هر کروموزوم ۲۴ ژن در نظر بگیریم که مقادیر هر یک از آن‌ها ۰ یا ۱ است. صفر نشان‌دهنده عدم انتخاب ویژگی متناظر با آن و یک نشان‌دهنده این است که ویژگی متناظر با آن انتخاب شده است.

## ۳. روش بررسی ادبیات

این بخش نحوه انتخاب و بررسی تحقیقات مربوط به تکنیک انتخاب ویژگی مبتنی بر الگوریتم فراابتکاری در سیستم‌های تشخیص نفوذ را معرفی می‌کند. بر اساس سه سوال زیر، تحقیقات مرتبط انتخاب شده‌اند. ۱. چه الگوریتم‌های فراابتکاری در انتخاب ویژگی برای سیستم‌های تشخیص نفوذ استفاده می‌شود؟

۲. انتخاب ویژگی مبتنی بر الگوریتم‌های فراابتکاری در سیستم‌های تشخیص نفوذ چقدر مؤثرند؟

۳. چالش‌ها و شکاف‌های تحقیقاتی در الگوریتم‌های فراابتکاری موجود برای انتخاب ویژگی در سیستم‌های تشخیص نفوذ چیست؟

سوال اول به ما کمک می‌کند تا الگوریتم‌های فراابتکاری جدیدی که در بحث انتخاب ویژگی برای سیستم‌های تشخیص نفوذ استفاده می‌شوند، بشناسیم و تحقیقات مربوط به آن را جمع‌آوری و تجزیه و تحلیل کنیم. سوال دوم، به بررسی معیارهای ارزیابی و عملکرد آن الگوریتم‌ها می‌پردازد. و سوال آخر چالش‌های موجود در بحث را بررسی می‌کند تا پیشنهادهای تحقیقات آینده ارائه کند.

کلمات کلیدی جستجو شده برای یافتن مقالات معتبر برای این مطالعه عبارتند از: "سیستم تشخیص نفوذ"، "سیستم تشخیص نفوذ در ابر"، و "انتخاب ویژگی در سیستم تشخیص نفوذ". این مقالات در مجلات معتبر مانند ScienceDirect, Springer, IEEE و سایر منابع علمی معتبر، در بازه زمانی سال‌های ۲۰۱۸ تا ۲۰۲۴ منتشر شده‌اند.

## ۴. مسئله انتخاب ویژگی در سیستم تشخیص نفوذ:

ترافیک داده در محیط ابری خیلی متنوع و دارای حجم زیادی از ویژگی‌هاست. سیستم‌های تشخیص نفوذ، با این ویژگی‌ها سروکار دارند. طبقه‌بندی ترافیک ورودی مسئله سختی است زیرا: (۱) وجود بسیاری از ویژگی‌ها منجر به فرآیندهای آموزشی طولانی می‌شود (به ویژه زمانی که ویژگی‌ها بسیار همبسته هستند)، درحالی‌که دقت پیش‌بینی متناسب آن بهبود نمی‌یابد. (۲) برخی از ویژگی‌ها ممکن است در طول فرآیند طبقه‌بندی، سوگیری ایجاد کنند، به‌ویژه آن‌هایی که ارتباط کمی با ترافیک داده‌ای مورد طبقه‌بندی، داشته باشند [۱۰]. برای این منظور، با کاهش فضای ویژگی و حفظ مهم‌ترین ویژگی‌ها، انتخاب ویژگی به یک مرحله پیش پردازش مهم در مدیریت شبکه و به‌ویژه برای اهداف تشخیص نفوذ شبکه تبدیل می‌شود [۱۰] و [۲۸].

الگوریتم‌های هوش جمعی (Swarm Intelligence Algorithms): الگوریتم‌های هوش جمعی از رفتار اجتماعی گروه‌های موجودات زنده مانند مورچه‌ها، پرندگان، و ماهی‌ها الهام گرفته شده‌اند. این الگوریتم‌ها به دنبال یافتن راه‌حل‌های بهینه از طریق تعاملات ساده بین اعضای جمعیت هستند [۳۱]. مهمترین الگوریتم‌های هوش جمعی عبارتند از: الگوریتم بهینه‌سازی ازدحام ذرات، الگوریتم بهینه‌سازی زنبور، الگوریتم کبوتر.

الگوریتم بهینه‌سازی ازدحام ذرات (Particle Swarm Optimization - PSO) الهام گرفته از رفتار جمعی ذرات در طبیعت است. در این الگوریتم، یک جمعیت از ذرات در یک فضای چندبعدی قرار دارند که هر ذره یک حالت (یا راه‌حل) را نمایش می‌دهد. هر ذره به یک مکان در فضا نسبت داده می‌شود و در هر مرحله، بهترین حالت خود و بهترین حالت جمعیت را ذخیره می‌کند. ذرات در جستجوی بهینه‌سازی با تغییر مکان‌شان به سمت بهترین حالت‌ها حرکت می‌کنند. در هر مرحله، هر ذره با در نظر گرفتن مکان بهترین حالت خود و بهترین حالت جمعیت، سرعت و مکان خود را به‌روزرسانی می‌کند. این به‌روزرسانی از طریق محاسبه میزان تأثیرگذاری دو عامل اصلی، یعنی جهت بهترین حالت شخصی ذره و جهت بهترین حالت جمعیت انجام می‌شود. این تأثیرگذاری می‌تواند با استفاده از ضرایب وزنی مناسبی کنترل شود. یکی از مزایای اصلی PSO، سادگی پیاده‌سازی و انطباق آسان آن با مسائل مختلف است.

الگوریتم بهینه‌سازی زنبور (Bee Colony Optimization - BCO) الهام گرفته از رفتار جمعی زنبورها در جستجوی منابع غذایی است. در این الگوریتم، یک جمعیت از زنبورها وجود دارد که هر کدام از آن‌ها یک مسیر احتمالی برای یافتن بهینه‌سازی را نمایش می‌دهد. زنبورها در این الگوریتم به دو دسته تقسیم می‌شوند: زنبورهای کارگر و زنبورهای معمولی. زنبورهای کارگر به دنبال مکان‌های مختلف منابع غذایی (راه‌حل‌های پتانسیلی) در فضای جستجو هستند، در حالی که زنبورهای معمولی اطلاعات از زنبورهای کارگر را دریافت کرده و تصمیم‌گیری می‌کنند که کدام مکان‌ها را برای بررسی بیشتر انتخاب کنند. در هر مرحله، زنبورهای کارگر به مکان‌های مختلف در فضای جستجو حرکت می‌کنند و اطلاعات دریافتی را برای بهبود مکان‌های پتانسیلی به زنبورهای معمولی ارسال می‌کنند. زنبورهای معمولی سپس این اطلاعات را بررسی و مکان‌های مناسب برای جستجوی بیشتر تعیین می‌کنند. این فرآیند تکرار می‌شود تا بهینه‌سازی بهبود یابد. مزیت اصلی الگوریتم بهینه‌سازی زنبور، توانایی ترکیب جستجوی محلی و جهانی در یافتن بهینه‌سازی است. همچنین، قابلیت تطبیق با محیط‌های مختلف و ساختار ساده این الگوریتم آن را به یک روش مؤثر در حل مسائل بهینه‌سازی مختلف، از جمله مسائل مسیریابی و بهینه‌سازی سیستم‌های پیچیده می‌کند. الگوریتم‌های مبتنی بر مسیر جستجو به دنبال بهینه‌سازی با پیگیری و اصلاح مسیر حرکت یک راه‌حل در فضای

جستجو هستند. این الگوریتم‌ها معمولاً از یک نقطه شروع و با دنبال کردن مسیرهایی به سمت بهبود کیفیت راه‌حل حرکت می‌کنند [۳۲]. مهمترین الگوریتم مبتنی بر مسیر جستجو الگوریتم جستجوی تابو (Tabu Search) است. الگوریتم جستجوی تابو (Tabu Search) یک الگوریتم بهینه‌سازی است که برای حل مسائل بهینه‌سازی ترکیباتی استفاده می‌شود. این الگوریتم ابتدا از یک حالت اولیه شروع می‌کند و سپس با استفاده از جستجوی محلی، به بهبود آن حالت می‌پردازد. اما برخلاف الگوریتم‌های جستجوی محلی که ممکن است در یک بهینه محلی گیر کنند، الگوریتم جستجوی تابو از طریق استفاده از یک لیست تابو (Tabu List)، از بازگشت به حالت‌هایی که قبلاً بررسی شده‌اند و به‌عنوان محلی بهینه مشخص شده‌اند، جلوگیری می‌کند. علاوه بر استفاده از لیست تابو، الگوریتم جستجوی تابو از استراتژی‌های متنوعی مانند تغییرات جوار (Neighborhood Changes)، جستجوی محلی (Local Search) و معیارهای ارزیابی مختلف برای بهبود بهینه‌سازی استفاده می‌کند [۳۲].

این الگوریتم قابلیت پیشرفت پیش‌بینی را داراست، به این معنی که با گذر از هر مرحله، به یک بهینه بهتر نسبت به حالت قبلی دست می‌یابد. مزیت اصلی الگوریتم جستجوی تابو، توانایی پیش‌بینی و جلوگیری از گیر افتادن در بهینه‌های محلی است، همچنین این الگوریتم به دلیل ساختار انعطاف‌پذیری که دارد، می‌تواند در حل مسائل مختلفی از جمله مسائل ناپیوسته، مسائل جریان کار، و بهینه‌سازی ترکیباتی استفاده شود. شکل ۲ طبقه‌بندی الگوریتم‌های فراابتکاری و فراوانی آن‌ها را در مطالعات بررسی شده، نشان می‌دهد. الگوریتم‌های فراابتکاری را طبق مقالات بررسی شده به سه دسته تقسیم کردیم. الگوریتم‌های تکاملی که شامل انواع الگوریتم‌های ژنتیک و تکاملی تفصلی است [۴۸، ۵۷، ۶۰، ۶۵، ۵۰، ۵۱، ۵۹، ۶۲، ۶۴، ۶۶]. الگوریتم‌های مبتنی بر جستجو مانند الگوریتم جستجوی تابو [۳۶] و الگوریتم‌های مبتنی بر هوش جمعی که شامل الگوریتم‌های کبوتر [۴۶، ۷۰]، خفاش [۳۴، ۵۴، ۵۸، ۷۱]، گرگ خاکستری [۷۵ و ۷۶]، کلاغ [۴۱ و ۶۳]، ازدحام ذرات [۴۵، ۶۷، ۶۸]، کلونی زنبورها [۴۳، ۵۶، ۶۹]، جستجوی گرانشی [۳۹]، کاتل فیش (cuttlefish) [۳۵، ۵۵]، کرم شبتاب [۳۳، ۵۳]، الگوریتم نهنگ [۷۲، ۷۳ و ۷۴] و دیگر الگوریتم‌ها [۱۱۴، ۳۸، ۴۰، ۴۲، ۴۴، ۴۷، ۴۹، ۵۲، ۶۱] می‌باشد.

شکل ۳ نشان می‌دهد که الگوریتم‌های فراابتکاری ۲۲٪، الگوریتم‌های مبتنی بر جستجو ۲٪ و الگوریتم‌های مبتنی بر هوش جمعی ۷۶٪ در انتخاب ویژگی برای تشخیص نفوذ استفاده شده‌اند. الگوریتم‌های مبتنی بر هوش جمعی از رفتار جمعی سیستم‌های غیرمتمرکز استفاده می‌کنند و مقیاس‌پذیری، سازگاری و استحکام جمعی را در حل مسائل پیچیده فراهم می‌کنند [۱۵۴]. استفاده از الگوریتم‌های هوش جمعی مزایای زیادی دارد از جمله:

۱. دسترسی به دانش جمعی: با استفاده از الگوریتم‌های هوش جمعی، می‌توان از دانش و تجربه یک گروه از افراد بهره‌برداری کرد. این

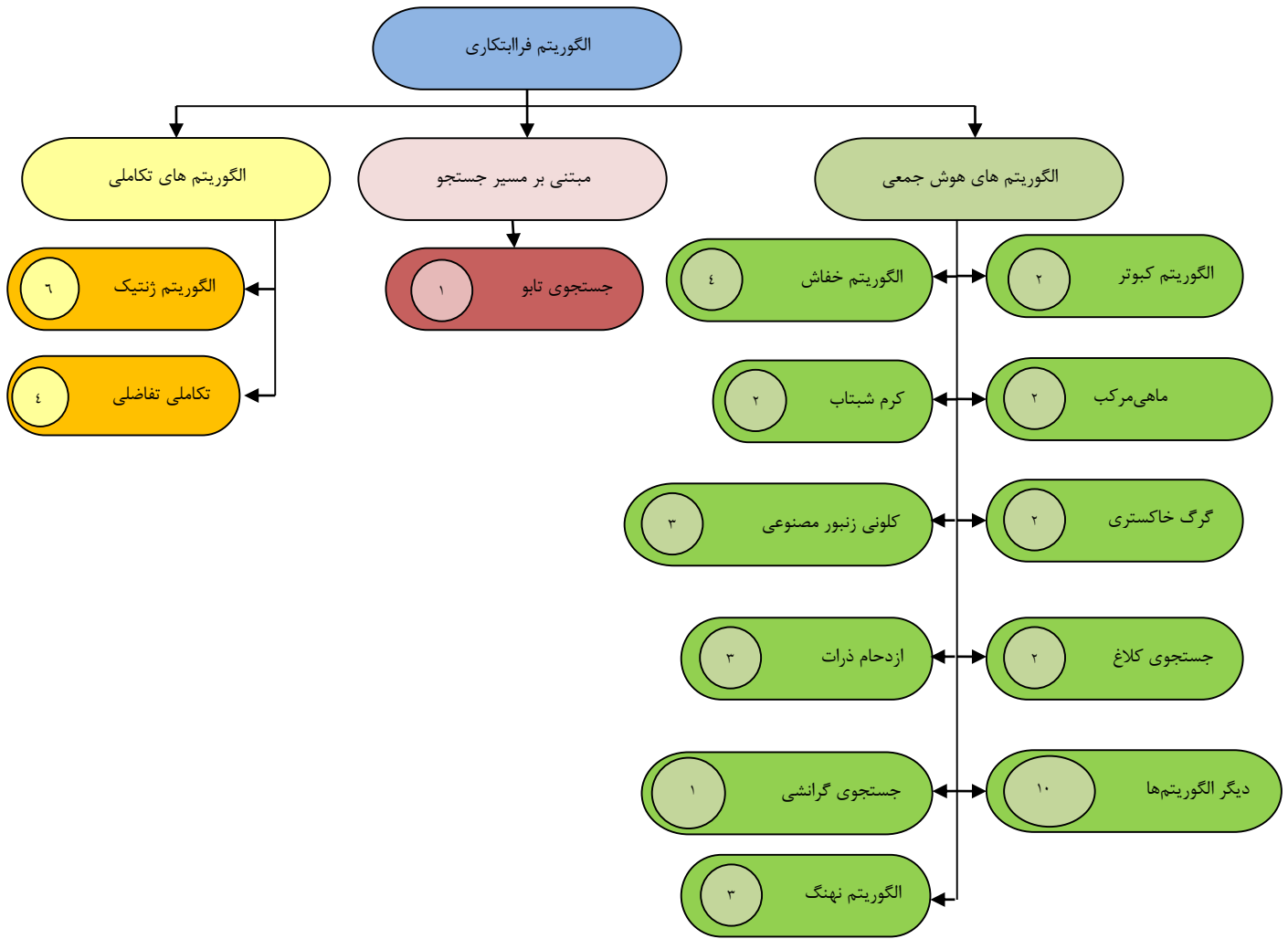
الگوریتم‌ها از تلاش و همکاری گروهی افراد برای حل مسئله استفاده می‌کنند و در نتیجه، دسترسی به دانش و تجربه بیشتری نسبت به روش‌های تک‌شخصی فراهم می‌کنند.

۲. بهبود کارایی و کیفیت: با استفاده از الگوریتم‌های هوش جمعی، می‌توان بهبود کارایی و کیفیت حل مسائل را به دست آورد. این الگوریتم‌ها با ترکیب ایده‌ها و راه‌حل‌های مختلف اعضای گروه، به یافتن راه‌حل بهینه و بهتری برای مسئله کمک می‌کنند.

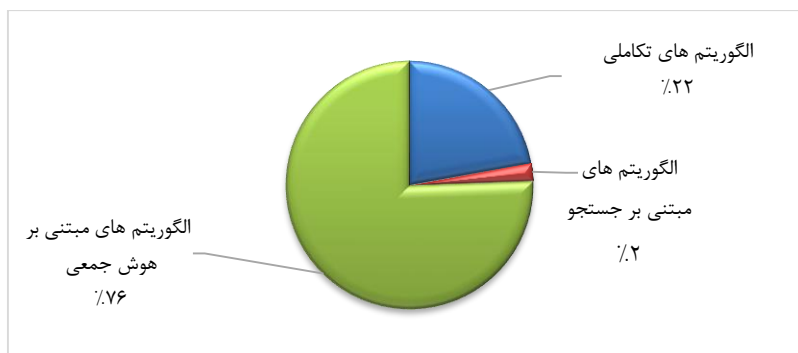
۳. مقاومت در برابر خطا: الگوریتم‌های هوش جمعی، به دلیل استفاده از چندین منبع دانش و تجربه، مقاومت بالاتری در برابر خطا دارند. اگر یکی از اعضای گروه اشتباه کند یا اطلاعات نادرستی ارائه دهد، سایر اعضا می‌توانند با ترکیب دانش خود، این خطا را تشخیص داده و راه‌حل درست را پیدا کنند.

۴. انعطاف‌پذیری: الگوریتم‌های هوش جمعی از انعطاف‌پذیری بالایی برخوردار و قادرند با تغییرات در شرایط و متغیرهای ورودی، بهبود و تطبیق بهتری را ارائه دهند. انعطاف‌پذیری به این معناست که الگوریتم‌ها می‌توانند با تغییرات و اضافه شدن عناصر جدید، به نتیجه بهتری برسند.

شکل (۴) تعداد تحقیق بر اساس الگوریتم‌های فراابتکاری را نشان می‌دهد. الگوریتم ژنتیک با ۱۳٪ بیشترین استفاده را در تحقیقات بررسی شده دارد (با ۶ مقاله). الگوریتم ژنتیک یکی از قدیمی‌ترین و معروف‌ترین الگوریتم‌های بهینه‌سازی فراابتکاری است که از رفتار و اصول انتخاب طبیعی در فرآیند تکاملی پیروی می‌کند. این الگوریتم از مکانیزم تکامل گیاهان و جانوران در طبیعت الهام گرفته است، یعنی در هر نسل، افراد با ویژگی‌های بهتر از نسل قبلی برای جمعیت انتخاب می‌شوند و از طریق ترکیب و جابجایی این ویژگی‌ها، نسل‌های بعدی به بهبود و بهینه‌سازی پیوسته می‌رسند. در این الگوریتم، هر فرد به‌عنوان یک کروموزوم نماینده از راه‌حل‌ها در فضای جستجو است و با استفاده از عملیات مانند انتخاب، ترکیب، و جابجایی، فرزندان جدیدی ایجاد می‌شوند که ویژگی‌های بهتری نسبت به والدین خود دارند. این فرآیند دائم تکرار می‌شود تا به یک حل بهینه یا حداقل بهینه‌تر برای مسئله مورد نظر برسیم. تحقیقات [۵۱، ۶۴] از ژنتیک استاندارد استفاده می‌کنند و در [۶۶، ۵۰] از ژنتیک چندهدفه استفاده شده است.



شکل ۲: طبقه بندی الگوریتم‌های فراابتکاری و فراوانی آن‌ها در مقالات بررسی شده



شکل ۳: درصد استفاده از دسته‌های مختلف الگوریتم‌های فراابتکاری در مطالعات بررسی شده

RUIZHE ZHAO و همکاران [۶۰] برای کاهش ابعاد مسئله تشخیص نفوذ ابتدا واریانس هر ویژگی را محاسبه کردند سپس ویژگی‌هایی که واریانس آن‌ها زیر مقدار آستانه باشد را حذف و در نهایت با استفاده از الگوریتم DE ویژگی‌های مناسب را از بین مجموعه ویژگی باقیمانده انتخاب کردند.

Ranjit Panigrahi و همکاران [۵۷] از تکاملی تفاضلی چندهدفه استفاده کرده‌اند.

Faezah Hamad Almasoudy و همکاران [۶۵] از DE استاندارد برای انتخاب ویژگی استفاده کرده‌اند.

الگوریتم بهینه‌سازی نهنگ، الگوریتم کلونی زنبور عسل و الگوریتم ازدحام ذرات با ۷٪ سومین الگوریتم فراابتکاری است

الگوریتم بهینه‌سازی نهنگ‌ها (Whale Optimization Algorithm - WOA) یک روش بهینه‌سازی فراابتکاری است که از رفتار شکار نهنگ‌های گوشت‌پشت الهام گرفته‌است. این نهنگ‌ها از روش خاص "حباب شبکه‌ای" برای شکار طعمه استفاده می‌کنند. در این روش، نهنگ‌ها به صورت دایره‌ای دور طعمه می‌چرخند و حباب‌هایی را ایجاد می‌کنند تا طعمه را به سمت سطح آب ببرند. این الگوریتم به دلیل قابلیت تطبیق با مسائل پیچیده و فضاهای جستجوی بزرگ، در بسیاری از مسائل بهینه‌سازی استفاده می‌شود.

Vaishali Ravindranath و همکاران [۷۳] یک نسخه بهبود یافته از الگوریتم بهینه‌سازی نهنگ باینری را برای انتخاب ویژگی ارائه و برای به‌روزرسانی موقعیت نهنگ‌ها از همبستگی جدید و دقت تشخیص طبقه بندی استفاده کردند.

Riyadh Rahef Nuiaa Al Ogaili و همکاران [۷۴] از الگوریتم نهنگ تغییر یافته بر اساس میانگین وزنی برای انتخاب ویژگی جهت شناسایی حملات بدافزار استفاده کرده‌اند. ویژگی‌هایی که وزن بالاتری دارند، انتخاب و ویژگی‌هایی که وزن کمتری دارند حذف می‌شوند.

[۷۲] از ترکیب الگوریتم‌های ژنتیک و نهنگ استفاده کرده‌است. برای بهبود فضای جستجوی نهنگ از عملگر متقاطع و برای جلوگیری از گیر افتادن در بهینه محلی از عملگر جهش ژنتیک استفاده شده‌است.

الگوریتم کلونی زنبورها (Bee Colony Algorithm) از رفتار جستجوی غذا توسط زنبورهای عسل الهام گرفته‌است. این الگوریتم بر اساس سه نوع زنبور کار می‌کند: زنبورهای جستجوگر، زنبورهای مشاهده‌گر و زنبورهای کارگر. زنبورهای جستجوگر به صورت تصادفی منابع غذایی جدید را در فضای جستجو پیدامی‌کنند و زنبورهای کارگر به منابع غذایی که قبلاً شناسایی شده‌اند، بازمی‌گردند و اطلاعات مربوط به کیفیت و موقعیت این منابع را به کندو گزارش می‌دهند. زنبورهای مشاهده‌گر بر اساس اطلاعات دریافتی از زنبورهای کارگر، تصمیم می‌گیرند که به کدام منبع غذایی بروند. این فرایند باعث می‌شود زنبورهای بیشتری به سمت منابع غذایی با کیفیت بالا هدایت شوند و به‌مرور زمان بهترین منابع غذایی (راه‌حل‌های بهینه) شناسایی شوند. توانایی الگوریتم زنبورها در کاوش کارآمد فضاهای راه‌حل پیچیده و

Dogukan Aksu و همکارش [۵۹] ژنتیک استاندارد را تغییر دادند طوری که موجب افزایش تنوع جمعیت و سرعت رسیدن به راه‌حل بهینه می‌شود. Nileshe Kunhare و همکارانش [۶۲] از الگوریتم ژنتیک برای انتخاب ویژگی در سیستم تشخیص نفوذ و از الگوریتم گرگ خاکستری برای بهینه‌سازی زیرمجموعه انتخابی استفاده کردند.

الگوریتم بهینه‌سازی خفاش و الگوریتم تکاملی تفاضلی با ۹٪ دومین الگوریتم فراابتکاری است که بیشترین استفاده را در مقالات مورد بررسی داشته‌اند.

الگوریتم خفاش، الهام‌گرفته از رفتار و روش شکار خفاش‌ها در طبیعت است. در این الگوریتم، خفاش‌ها برای شکار طعمه از یک روش پیچیده و هوشمندانه استفاده می‌کنند که شامل حرکت‌های تصادفی و جستجوی هوشمندانه در فضای سه‌بعدی است. خفاش‌ها باید به دنبال طعمه در فضای سه‌بعدی حرکت کنند و با توجه به فاصله خود از طعمه و جذابیت طعمه، حرکات مناسبی را انجام دهند تا به بهترین نقطه در فضا برسند.

Yuyang Zhou و همکارانش [۳۴] ویژگی‌های با اهمیت را بر اساس ترکیب الگوریتم خفاش و همبستگی بین ویژگی‌ها انتخاب می‌کنند. آن‌ها از CFS به عنوان تابع هدف استفاده کردند و معتقدند CFS باعث بهبود دقت طبقه‌بندی می‌شود و برای بهینه‌سازی مراحل آموزش و آزمون مفید است.

Waheed Ali H. M. Ghanem و همکارانش [۵۸] یک چارچوب دو مرحله‌ای با استفاده از یک الگوریتم BAT چندهدفه برای انتخاب ویژگی و یک BAT پیشرفته برای آموزش شبکه عصبی mlp ارائه کرده‌اند.

Bukola Fatimah Balogun و همکارانش [۵۴] برای انتخاب ویژگی در سیستم تشخیص نفوذ ابتدا از الگوریتم خفاش، سپس RNS برای انتخاب ویژگی از باقی‌مانده ویژگی‌ها استفاده کردند و پس از آن PCA آخرین مرحله انتخاب ویژگی است.

Arunesh K و Maheswari S [۷۱] از الگوریتم خفاش باینری برای انتخاب ویژگی استفاده کردند.

الگوریتم تکاملی تفاضلی (DE) ابتدا یک جمعیت از بردارها را تصادفی ایجاد می‌کند و سپس با استفاده از عملیات تفاضلی و ترکیب این بردارها، به بهبود و بهینه‌سازی آن‌ها می‌پردازد. در هر مرحله از الگوریتم، بردارهای جدیدی با استفاده از تفاضل بین بردارهای موجود در جمعیت ایجاد می‌شوند که با اعمال عملیات تفاضلی و انتخاب بهترین بردارها، جمعیت در هر مرحله به سمت بهبود و بهینه‌سازی حرکت می‌کند. این الگوریتم از ویژگی‌هایی مانند سادگی پیاده‌سازی، قابلیت تنظیم پارامترها و قابلیت مقیاس‌پذیری برخوردار است که آن را به یکی از الگوریتم‌های پرکاربرد و مؤثر در حل مسائل بهینه‌سازی می‌سازد.

محمد فریس و همکارانش [۴۸] از الگوریتم تکاملی تفاضلی اصلاح شده و برای کاهش همگرایی زودرس این الگوریتم از عملیات جهش جدید استفاده کرده‌اند.

یافتن بهینه جهانی، آن را به ابزاری قدرتمند در تحقیقات و برنامه‌های کاربردی هوش محاسباتی مدرن تبدیل می‌کند.

Arun Kumar Sangaiah و همکاران [۴۳] یک انتخاب ویژگی ترکیبی ابتکاری برای تشخیص نفوذ در محیط‌های ابری با استفاده از الگوریتم‌های بهینه‌سازی کلونی مورچه‌ها و زنبورها پیشنهاد کردند. آن‌ها ابتدا با استفاده از الگوریتم زنبور برای یافتن نگاشت عددی استفاده کرده، سپس با استفاده از ACO پاسخ‌های تقریبی را به دست می‌آورند.

Milos Stankovic و همکاران [۵۶] استفاده از ترکیب الگوریتم کلونی زنبورهای مصنوعی و کرم شبتاب را برای انتخاب ویژگی پیشنهاد می‌کنند. با توجه به یک عدد تصادفی، راه‌حل‌ها بر اساس الگوریتم‌های زنبور و کرم شبتاب به‌روزرسانی می‌شوند.

S. Velliangiri • P. Karthikeyan [۶۹] ترکیب کلونی زنبورهای مصنوعی تطبیقی با بهینه‌سازی ازدحام ذرات تطبیقی برای تشخیص فعالیت‌های نفوذی را طراحی کرده‌اند که در هر مرحله این دو الگوریتم نتایج خود را با یکدیگر به اشتراک می‌گذارند.

الگوریتم ازدحام ذرات (Particle Swarm Optimization - PSO) از رفتار اجتماعی گروهی از ذرات الهام گرفته‌است، مشابه رفتار پرندگان در هنگام پرواز دسته‌جمعی یا ماهی‌ها در شنا کردن گروهی. در این الگوریتم، هر ذره نماینده یک راه‌حل بالقوه است که در فضای جستجو حرکت می‌کند و موقعیت خود را بر اساس تجربه خود و همچنین بهترین موقعیتی که توسط دیگر ذرات مشاهده شده‌است، به‌روزرسانی می‌کند. PSO از دو مؤلفه اصلی، یعنی سرعت و موقعیت، برای هر ذره استفاده می‌کند و با به‌روزرسانی مداوم این مقادیر، به سمت بهترین راه‌حل همگرا می‌شود.

Shalini Subramani, M. Selvi [۴۵]، از ازدحام ذرات چندهدفه استفاده کرده‌اند. Wisam Elmasry و دوستانش [۶۷] ازدحام ذرات دوگانه باینری را برای انتخاب زیرمجموعه ویژگی‌ها و فراپارامترهای مدل پیشنهاد کردند. Roseline Oluwaseun Ogundokun و همکاران [۶۸] برای انتخاب ویژگی از ازدحام ذرات باینری، قبل از طبقه‌بندی ترافیک‌های نرمال و غیرنرمال استفاده کرده‌اند.

برای هر کدام از الگوریتم‌های کبوتر، کلاغ، گرگ خاکستری، کرم شبتاب و ماهی مرکب، ۲ مقاله بررسی شده‌است بنابراین ۴٪ از مقالات را شامل می‌شود.

الگوریتم کبوتر، یک روش بهینه‌سازی مستقیم الهام‌گرفته از رفتار و روش جستجوی غذا در کبوترها در طبیعت می‌باشد. در این الگوریتم، کبوترها به دنبال یافتن منابع غذایی در محیط خود می‌گردند و با استفاده از روش‌های جستجوی مؤثر و هوشمند، به بهینه‌سازی جستجوی خود می‌پردازند. این الگوریتم بر اساس اصول رفتاری و جستجوی کبوترها طراحی شده‌است، هر کبوتر به‌عنوان یک نقطه در فضای جستجو در نظر گرفته می‌شود و با حرکت به سمت منابع غذایی و استفاده از اطلاعات محیط، به بهترین نقطه در فضا هدایت می‌شود.

Orieh Abu Alghanam و همکاران [۴۶] یک الگوریتم کبوتر جستجوی محلی پیشرفته پیشنهاد داده‌اند.

Hadeel Alazzam و همکاران [۷۰] از یک روش جدید برای باینری کردن الگوریتم کبوتر استفاده کردند. در الگوریتم کلاغ، کلاغ‌ها به دنبال یافتن منابع غذایی در محیط خود می‌گردند و با استفاده از روش‌های جستجویی خاص، به بهینه‌سازی جستجوی خود می‌پردازند.

D. Jayalatchumy و همکارانش [۴۱] الگوریتم جستجوی کلاغ بهبود یافته ارائه داده و با استفاده از تکنیک آشوب، جستجوی محلی الگوریتم کلاغ را بهبود دادند. Ashish Khanna و همکارانش [۶۳] نیز از الگوریتم کلاغ باینری برای انتخاب ویژگی استفاده کرده‌اند.

در الگوریتم گرگ خاکستری، گرگ‌ها به دنبال شکار طعمه در محیط خود می‌گردند و با استفاده از تاکتیک‌ها و استراتژی‌های مختلف، سعی در بهینه‌سازی شکار خود دارند. هر گرگ خاکستری در الگوریتم به‌عنوان یک نقطه در فضای جستجو در نظر گرفته می‌شود و با استفاده از قوانین و اصول رفتاری خود، به بهترین نقطه در فضا هدایت می‌شود. این الگوریتم بر اساس اصول تکاملی و انتخاب طبیعی طراحی شده‌است، گرگ‌های با ویژگی‌های بهتر و مطابق با محیط خود برای شکار، انتخاب می‌شوند و با ترکیب و جابجایی ویژگی‌های مختلف، نسل‌های بهتری ایجاد می‌شود.

Qusay M. Alzubi و همکاران [۷۵] گرگ خاکستری باینری اصلاح شده پیشنهاد دادند. آن‌ها موقعیت گرگ‌ها را بر اساس چهار بهترین راه حل  $\alpha$ ،  $\beta$ ،  $\delta$  و  $\omega$  آپدیت می‌کنند یعنی گرگ امگا نیز در تغییر موقعیت گرگ‌ها مؤثر است. Taief Alaa Alamiedy و همکارانش [۷۶] از الگوریتم بهینه‌سازی گرگ خاکستری چندهدفه برای انتخاب ویژگی استفاده کرده‌اند.

الگوریتم کرم شبتاب یک الگوریتم بهینه‌ساز فراابتکاری است که از رفتار چشم‌کزن کرم‌های شبتاب الهام گرفته و مبتنی بر جمعیت است. Selvakumar B و دوستش [۳۳] از الگوریتم کرم شبتاب مبتنی بر Mutual Information (MI) استفاده کردند. Yakub Kayode Saheed [۵۳] الگوریتم کرم شبتاب را با استفاده از تابع سیگموئید باینری کرده و برای انتخاب ویژگی در سیستم تشخیص نفوذ استفاده کرده‌است.

الگوریتم ماهی مرکب (Cuttlefish Algorithm - CFA) یک الگوریتم بهینه‌سازی فراابتکاری است که از رفتار و ویژگی‌های ماهی مرکب الهام گرفته‌است. الگوریتم از توانایی ماهی مرکب در تغییر رنگ و الگوهای بدن خود برای به‌روزرسانی موقعیت راه‌حل‌ها استفاده می‌کند. این تغییر رنگ به معنای تغییر در موقعیت راه‌حل‌ها در فضای جستجو است تا به نقاط بهینه نزدیک‌تر شوند. ماهی‌های مرکب در طبیعت از استتار برای مخفی شدن از دید شکارچیان و جلب توجه طعمه‌ها استفاده می‌کنند. در الگوریتم، این رفتار به‌عنوان جستجوی محلی برای بهبود راه‌حل‌های فعلی مدل‌سازی می‌شود. ترکیب راه‌حل‌ها (مشابه جفت‌گیری در طبیعت) و جهش‌های تصادفی به منظور افزایش تنوع جمعیت و

## 5. تأثیر انتخاب ویژگی مبتنی بر الگوریتم‌های فرآینک‌سازی

### در IDS

برای پاسخ به پرسش دوم، نیاز است که معیارهای ارزیابی مقالات بررسی شود. از آنجاکه در مقالات گذشته، روی مجموعه داده‌های مختلف کار شده است، ابتدا مجموعه داده‌های استفاده شده در مقالات را بررسی می‌کنیم سپس به معیارهای ارزیابی می‌پردازیم تا بتوانیم الگوریتم‌های فرآینک‌سازی برای انتخاب ویژگی در تشخیص نفوذ را مقایسه کنیم.

### 1.5.1. مجموعه داده‌ها

یکی از مسائل مهم هنگام برخورد با رویکردهای انتخاب ویژگی نظارت-شده در تحلیل ترافیک، یافتن مجموعه‌های آموزشی است که هم جدید باشد و هم برچسب‌گذاری شده باشد [۱۰]. در ادامه به مجموعه داده‌هایی که در تحقیقات گذشته استفاده شده است می‌پردازیم.

UNSW-NB: این مجموعه داده توسط چهار ابزار IXIA PerfectStorm, Tcpdump, Argus و Bro-IDS تولید شده است. این ابزارها برای ایجاد برخی از انواع حملات از جمله DoS, Exploits, Shellcode, Reconnaissance, Generic Worms و استفاده می‌شوند [۲۱]. KDD99: این مجموعه داده براساس برنامه تخمین IDS DARPA'98 ایجاد شده است و ترافیک شبکه به مدت هفت هفته است. شامل ۴۹۰۰۰۰ رکورد است. حملات در گروه‌های زیر دسته‌بندی می‌شوند: حمله کاربر به ریشه (U2R)، حمله انکار سرویس (DoS)، حمله از راه دور به محلی (R2L)، و حمله کاوشگر. مجموعه داده KDD Cup 1999 شامل ۴۱ ویژگی است که به سه کلاس زیر طبقه‌بندی می‌شوند: ویژگی‌های اساسی، ویژگی‌های محتوا، و ویژگی‌های ترافیک [۲۱]-NSL-KDD: این مورد توسط تولایی و همکاران پیشنهاد شده است [۲۲] که منجر به حل شدن برخی از مشکلات مجموعه داده‌های KDD'99 شد. آن‌ها مجموعه داده‌های KDD را در موارد زیر بهبود دادند: بدون افزودن، ورودی‌های تکراری وجود ندارد، تعداد ورودی‌های انتخابی سازمان‌دهی شده است، و تعداد ورودی‌ها معقول است. مقالات زیادی در مورد تشخیص نفوذ وجود دارد که از هر دو مجموعه داده با هم در تکامل عملکرد استفاده می‌کنند و دریافتند که بهترین نتایج در NSL-KDD یافت می‌شود [۲۱]. CDMC2012: یکی دیگر از مجموعه داده‌های محبوب CDMC2012 [۲۳] است که با استفاده از چندین HoneyPot از پنج شبکه مختلف ایجاد شد. در میان نمونه‌های CDMC2012، مواردی با عنوان «ناشناخته» وجود دارد که صرفاً مواردی که به عنوان «حمله» یا «عادی» برچسب‌گذاری شده بودند، کنار گذاشته شدند. بدون از دست دادن کلیت، این مجموعه داده به دو مجموعه تقسیم شد، یک مجموعه آموزشی شامل ۴۸۲۵۷ نمونه و یک مجموعه تست با ۸۰۰۰۰. هر نمونه با ۱۴ ویژگی، از جمله کلاس، نشان داده می‌شود [۱۵]. CICIDS2017: این مجموعه داده حاوی داده‌هایی است که در همان دوره از مجموعه داده‌های CIC DoS گرفته شده است. همان دوره از مجموعه داده‌های CIC DoS گرفته شده است. CICIDS2017 توسط شرف‌الدین و همکاران پیشنهاد شده است [۲۴].

جلبگیری از گرفتار شدن در بهینه‌های محلی استفاده می‌شود. در هر تکرار، راه‌حل‌های جدید با راه‌حل‌های قبلی مقایسه می‌شوند و بهترین‌ها برای تکرارهای بعدی انتخاب می‌شوند.

سارا محمدی و همکاران [۳۵] ترکیبی از روش فیلتر و روش بسته‌بند به‌عنوان انتخاب ویژگی پیشنهاد داده و ابتدا بر اساس ضریب همبستگی خطی، ویژگی‌ها را گروه‌بندی می‌کنند سپس با الگوریتم ماهی مرکب (CFA) ویژگی‌های نامربوط و اضافی را از مجموعه داده اصلی حذف می‌کنند.

Sharma و همکارانش [۵۵] الگوریتم ماهی مرکب اصلاح شده‌ای را پیشنهاد داده‌اند که از دو فرآیند اصلی تشکیل شده است: فرآیند دید و فرآیند بازتاب. این فرآیندها به یافتن راه‌حل بهینه جهانی کمک می‌کنند و راه‌حل جدید نتیجه ترکیب فرآیندهای دید و بازتاب است.

در این بین مقالاتی وجود دارند که از دیگر الگوریتم‌های فرآینک‌سازی استفاده کرده‌اند. Anjum Nazir و همکارش [۳۶] از جستجوی تابو برای انتخاب ویژگی استفاده کردند. Nojood O. Aljehane و همکارانش [۳۹] جستجوی گرانشی باینری را برای انتخاب ویژگی به کار بردند.

Selva Rani و همکاران [۱۱۴] ترکیبی از الگوریتم بهینه‌سازی ارشمیدس (AOA) و بهینه‌سازی شاهین آتشین (FHO) ارائه کردند.

Anil V Turukmane و همکار [۳۸] ویژگی‌های استخراج شده را توسط الگوریتم بهینه‌سازی گوساوک شمالی مبتنی بر مخالفت (ONgO) بهینه‌سازی کردند. Laith Abualigah و همکارانش [۴۰] از الگوریتم آکوا اصلاح شده (mao) برای سیستم‌های تشخیص نفوذ استفاده کردند.

Tingyao Jiang و همکارانش [۴۲] الگوریتم بهینه‌سازی مبتنی بر جغرافیای زیستی را بهبود دادند. آن‌ها برای کنترل عملیات اپراتورهای مهاجرت و جهش از انتخاب رولت استفاده می‌کنند.

Zhiwei Ye و همکارانش [۴۴] از یادگیری مبتنی بر مخالفت نخبگان برای بهبود الگوریتم بهینه‌سازی پرورش ترکیبی (HBO) استفاده کردند تا جستجوی بهینه‌تری داشته باشد.

Malek Barhoush و همکارانش [۴۷] روش‌های یادگیری مبتنی بر مخالفت، یادگیری مبتنی بر مخالفت نخبگان و روش جستجوی همسایگی متغیر، الگوریتم ازدحام سالپ (SSA) را بهبود دادند.

K.G. Maheswari و همکاران [۴۹] از بهینه‌سازی تغذیه پرتوی اصلاح شده، استفاده کردند.

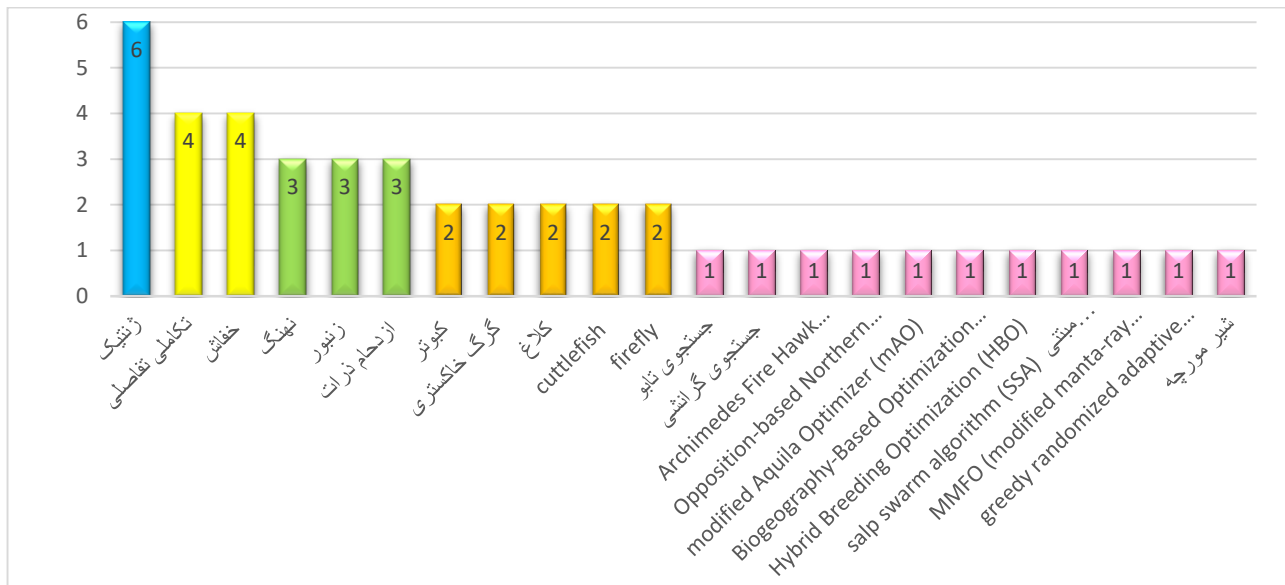
Silvio E. Quincozes و همکارانش [۵۲] از روش جستجوی تطبیقی تصادفی حریرانه (GRASP) برای انتخاب ویژگی استفاده کرده‌اند.

Junwen Chen و همکارانش [۶۱] از الگوریتم ترکیب شیر مورچه و محاسبات کوانتومی استفاده کرده‌اند.



[۲۰]، مجموعه داده AWID-CLS-R-Trn انتخاب شده است که در آن "R" یک مجموعه داده کاهش یافته را نشان می‌دهد. همچنین دارای مجموعه داده AWID-ATK-R-Trn است که در آن "ATK" شامل ۱۶ کلاس هدف است. ۱۶ نوع کلاس هدف حمله در

حملات شامل Web, Heartbleed, DoS, Brute Force SSH هستند. Brute Force FTP و DDoS و Botnet, Infiltration, Attack, AWID [۲۱]: مجموعه داده AWID [۲۵] شامل گروهی از مجموعه داده‌های بزرگ مانند AWID-CLS-R-Tst, AWID-CLS-R-Trn, AWID-CLS-F-Tst, AWID-CLS-F-Trn است. در آزمایش تحقیق



شکل ۴: تعداد تحقیق بر اساس الگوریتم‌های فراابتکاری در مطالعات بررسی شده

که از ضبط ترافیک شبکه استخراج شده است. SWat: داده‌های گرد-آوری شده از بستر آزمایش شامل ۱۱ روز کار مداوم است. داده‌های نرمال طی ۷ روز جمع‌آوری شد و داده با سناریوهای ۴۱ حمله ۴ روز جمع‌آوری شد [۵۲]. در طول کار، کلیه داده‌های ترافیک شبکه، حسگر و محرک گردآوری شد. ترافیک شبکه و تمام مقادیر به دست آمده از ۵۱ حسگر و محرک جمع‌آوری شده است. داده‌ها بر اساس رفتارهای عادی و غیرعادی برچسبگذاری شده‌اند. ISCX2012 مرکز امنیت اطلاعات تعالی این مجموعه داده را ارائه کرده است. ISCX شامل ۲۰ ویژگی و ۱۵۱۲۰۰۰ رکورد است که فعالیت‌های شبکه را به مدت ۷ روز دربرمی‌گیرد. [۵۸] HCRL-car hacking مجموعه داده‌های هک خودرو شامل حمله DoS، حمله فازی، جعل چرخ‌دنده درایو، و جعل RPM gauge است. مجموعه داده‌ها با ثبت ترافیک CAN از طریق پورت OBD-II از یک وسیله نقلیه واقعی درحالی‌که حملات تزریق پیام انجام می‌شد، ساخته شدند. مجموعه داده‌ها شامل ۳۰۰ نفوذ از تزریق پیام‌اند. هر نفوذ به مدت ۳ تا ۵ ثانیه انجام می‌شود و هر مجموعه داده در کل ۳۰ تا ۴۰ دقیقه از ترافیک CAN را شامل می‌شود [۵۹]. CIRA-CIC-DOHBrw-2020 مجموعه داده CIRA-CIC-DoHBrw-2020 ترافیک DoH عادی و مخرب را همراه با ترافیک غیر DoH ثبت می‌کند. این مجموعه داده دارای ۳۴ ویژگی و چهار کلاس است. تعداد نمونه‌های این مجموعه داده حدود ۱،۴ میلیون است [۶۴]. ADFA-LD مجموعه داده ADFA-LD دارای دو برچسب عادی و حمله است که حملاتش به گروه‌های Hydra-FTP, Hydra-adduser,

مجموعه داده "ATK" با جزئیات بیشتر مانند Honeypot, Evil Twins و EvilTwins و سایر حملات موجود است. داده‌های "CLS" دارای نسخه فشرده دسته‌های حمله هستند که عبارتند از تزریق، سیل، حمله جعل هویت و کلاس عادی که در مجموعه داده "ATK" فهرست شده اند [۲۰]. BoT-IoT: مجموعه داده BOT-IOT [۱۱۴] با طراحی یک محیط شبکه واقعی در آزمایشگاه Cyber Range در UNSW Canberra ایجاد شد. محیط شبکه ترکیبی از ترافیک عادی و بات نت را در خود جای داده است. این مجموعه داده دارای ۴۵ ویژگی و 3668522 رکورد است. فایل‌ها بر اساس دسته و زیرمجموعه حمله جدا شدند تا در فرآیند برچسب‌گذاری بهتر کمک شود. مجموعه داده شامل حملات DDoS, DoS, OS, Service Scan و Keylogging, Data Exfiltration است که حملات DDoS و DoS بر اساس پروتکل مورد استفاده بیشتر سازماندهی شده است. CSE-CIC-IDS 2018: مجموعه داده CSE-CIC-IDS 2018 مجموعه داده معیار برای سیستم‌های تشخیص نفوذ (IDS) در اینترنت اشیا (IoT) است [۳۸]. این مجموعه داده شامل بیش از ۸۰ میلیون رکورد، با ترکیبی از ترافیک عادی و مخرب از انواع دستگاه‌ها و پروتکل‌های IoT است. مجموعه داده شامل ترافیک ناشی از حملات شبیه‌سازی شده، مانند حملات Mirai, Botnet, Denial-of-Service (DoS) است. مجموعه داده CSE-CIC-IDS 2018 شامل ۸۰ ویژگی است

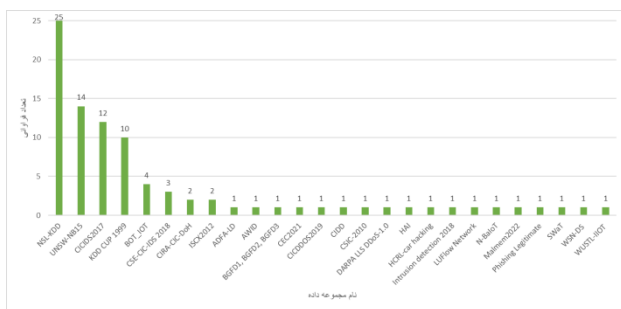
جدول ۴: مجموعه داده ها و فراوانی آنها

Dataset Name	تعداد	مراجع
ADFA-LD	1	[72]
AWID	1	34
BGFD1, BGFD2, BGFD3	1	50
BOT_IOT	4	23,51,64,37
CEC2021	1	44
CICDDOS2019	1	47
CICIDS2017	12	34,42,47,49,55,56,57,58,59,66,67,72
CIDD	1	45
CIRA-CIC-DoH	1	47,64
CSE-CIC-IDS 2018	3	38, 47, 60
CSIC-2010	1	49
DARPA LLS DDoS-1.0	1	49
HAI	1	44
HCRL-car hacking	1	59
Intrusion detection 2018	1	47
ISCX2012	2	58, 74
KDD CUP 1999	10	33, 34,35, 45,46, 50, 58, 68, 70, 71
LUFlow Network	1	47
N-BaIoT	1	47
Malmem2022	1	47
NSL-KDD	25	37,40,41,42,43,44,45,46, 47,48,50,54,58,60,61,62, 63,65,66,67,69,70,73,75,76
Phishing Legitimate	1	47
SWaT	1	52
UNSW-NB15	14	36,38,39,41,46,47,53,56, 58,59,63,64,66,70
UNSW-NB16	1	
WSN-DS	1	52
WUSTL-IIOT	1	44

در مجموع ۴۴ ویژگی و ۱,۵۷۹ نمونه دارد [۷۲]. مجموعه داده‌های دیگری وجود دارد که مقالات کمتری استفاده شده‌اند مانند CEC2021, CIDD, WUSTL-IIOT and HAI که در [۴۴] و- CIRA, CICDDOS2019, Intrusion detection 2018, Phishing [۴۷] Legitimate, Malmem2022, N-BaIoT, LUFlow Network و BGFD1 در [۴۹], DARPA LLS DDoS-1.0 و CSIC-2010 در [۴۹] و BGFD2 و BGFD3 در [۵۰] و ISCXURL-2016 در [۷۴]. جدول ۴ لیست مجموعه داده‌ها به همراه فراوانی آنها در مقالات بررسی شده را نشان می‌دهد.

مجموعه داده NSL-KDD با ۲۵ فراوانی بیشترین استفاده را دارد و سپس مجموعه داده UNSW-NB15 با ۱۴ فراوانی در رتبه دوم بیشترین آزمایش را داراست.

شکل ۵ نمودار فراوانی مجموعه داده را نشان می‌دهد.



شکل ۵: فراوانی مجموعه داده

بنابراین مجموعه داده‌های محبوب به ترتیب UNSW-NB15, NSL-KD, CSEC-CIC-IDS 2018, BOT\_IOT, KDD CUP 1999, CICIDS2017 هستند و دیگر مجموعه داده‌ها شامل ADFA-LD, AWID, BGFD1, BGFD2, BGFD3, CIDD, CICDDOS2019, CEC2021, HAI, DARPA LLS DDoS-1.0, CSIC-2010, CIRA-CIC-DoH, LUFlow Network, Intrusion detection 2018, HCRL-car hacking, WSN-DS, SWaT, Phishing Legitimate, Malmem2022, N-BaIoT و WUSTL-IIOT هستند.

## ۲.۵. معیارهای ارزیابی

در این بخش ابتدا معیارهای ارزیابی متداول را تعریف می‌کنیم سپس مقادیر به این معیارهای ارزیابی را در مقالات منتخب بررسی می‌کنیم.

مهم‌ترین معیار برای تعیین کارایی یک سیستم تشخیص نفوذ، معیار Accuracy می‌باشد. دقت کلی تشخیص نفوذ، به معنای میزان درستی سیستم تشخیص نفوذ (IDS) در تشخیص و شناسایی تلاش‌های نفوذ و حملات به شبکه یا سیستم است. این دقت معمولاً به عنوان نسبت تعداد حملات و نفوذهای صحیح شناسایی شده به مجموع حملات واقعی در سیستم تعریف می‌شود. به عبارتی، دقت تشخیص نفوذ نشان می‌دهد که چه تعداد از حملات واقعی توسط سیستم تشخیص نفوذ به درستی شناسایی شده‌اند. رابطه ۱ فرمول Accuracy را نشان می‌دهد.

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

Accuracy یک معیار کلی است که همه پیش‌بینی‌های صحیح را در نظر می‌گیرد، چه حمله باشند و چه نباشند.

در سیستم‌های تشخیص نفوذ، تعداد بالای پیش‌بینی‌های نادرست حمله می‌تواند منجر به هشدارهای بی‌مورد شود که باعث ایجاد اختلال و مصرف منابع می‌شود. در اینجا Precision می‌تواند معیار بهتری برای ارزیابی کارایی سیستم باشد. Precision به نسبت تعداد پیش‌بینی‌های مثبت صحیح به تعداد کل پیش‌بینی‌های مثبت اشاره دارد. به عبارت دیگر، Precision نشان می‌دهد که از میان تمام نمونه‌هایی که به عنوان حمله

$$FPR = \frac{FP}{FP+TN} \quad (5)$$

پیش‌بینی شده‌اند، چند

درصد واقعاً حمله بوده‌اند. رابطه ۲ فرمول این معیار را نشان می‌دهد. از آنجا که معیار Accuracy در شرایطی که کلاس‌ها به شدت نامتوازن هستند، می‌تواند گمراه‌کننده باشد بنابراین Precision می‌تواند دید بهتری از عملکرد مدل برای دسته‌بندی نمونه‌های مثبت ارائه دهد.

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

Recall یا Sensitivity یا نرخ تشخیص یا True Positive Rate (TPR): به نسبت تعداد نمونه‌های مثبت شناسایی شده به کل تعداد نمونه‌های مثبت واقعی اشاره دارد. به عبارت دیگر، Recall نشان می‌دهد که مدل چه تعداد از کل نمونه‌های مثبت را به درستی شناسایی کرده است. این معیار با رابطه (۳) محاسبه می‌شود.

$$Recall = \frac{TP}{TP+FN} \quad (3)$$

این معیار برای ارزیابی عملکرد مدل در شناسایی تمامی نمونه‌های مثبت اهمیت دارد. به عبارتی Recall یک معیار حیاتی برای ارزیابی کارایی مدل است زیرا نشان می‌دهد که مدل چقدر در شناسایی تمامی موارد مثبت، موفق است. شرایطی را در نظر بگیرید که حمله وجود دارد ولی مدل آن را شناسایی نمی‌کند در این صورت سیستم با عواقب جدی مواجه می‌شود. Recall به ما کمک می‌کند تا مطمئن شویم مدل تا چه حد، همه موارد مثبت را شناسایی می‌کند.

F-measure (که گاهی به عنوان F1-score شناخته می‌شود) یکی از

معیارهای مهم در ارزیابی عملکرد مدل‌های یادگیری ماشین و به ویژه در مسائل تشخیص نفوذ است. این معیار ترکیبی از دو معیار Precision و Recall است و به منظور ایجاد تعادل بین این دو معیار مورد استفاده قرار می‌گیرد که با رابطه (۴) محاسبه می‌شود. F-measure به تغییرات در هر دو معیار Precision و Recall حساس است و می‌تواند تصویری دقیق‌تر از عملکرد مدل در شناسایی صحیح نمونه‌های مثبت ارائه دهد.

$$F-measure = \frac{2*Precision*Recall}{Precision+Recall} \quad (4)$$

F-measure به خصوص زمانی مهم است که باید تعادلی بین Precision و Recall برقرار شود. این معیار برای مسائلی که نیازمند بالانس بین نرخ شناسایی صحیح (Recall) و نرخ مثبت‌های صحیح شناسایی شده (Precision) هستند، بسیار مناسب است. در مجموعه داده‌هایی که کلاس‌ها نامتوازن هستند، استفاده از F1-score می‌تواند معیار بهتری نسبت به Accuracy باشد، چرا که F1-score تأثیر خطاهای کلاسی که تعداد کمتری دارند را بهتر منعکس می‌کند.

نرخ مثبت کاذب (False Positive Rate - FPR) به نسبت تعداد نمونه‌هایی که به اشتباه، مثبت شناسایی شده‌اند به کل تعداد نمونه‌های واقعی منفی اشاره دارد (رابطه ۵). به عبارت دیگر، FPR نشان می‌دهد که چه درصدی از نمونه‌های منفی به غلط، مثبت تشخیص داده شده‌اند. نرخ مثبت کاذب بالا در سیستم‌های تشخیص نفوذ می‌تواند منجر به افزایش تعداد هشدارهای نادرست شود که باعث می‌شود اپراتورها زمان و منابع زیادی را صرف بررسی تهدیدات غیر واقعی کنند.

Mathews Correlation Coefficient (MCC) یک معیار آماری است که برای ارزیابی کیفیت دسته‌بندی‌های باینری (دودسته‌ای) استفاده می‌شود. این معیار یک مقدار بین -۱ و ۱ تولید می‌کند، که ۱ نشان‌دهنده دسته‌بندی کامل، ۰ نشان‌دهنده دسته‌بندی تصادفی و -۱ نشان‌دهنده دسته‌بندی کاملاً اشتباه است. MCC در مقایسه با معیارهای دیگر مانند دقت (Accuracy) و F1-Score جامع‌تر است زیرا همزمان تأثیرات True Positives (TP)، True Negatives (TN)، False Positives (FP) و False Negatives (FN) را در نظر می‌گیرد. MCC طبق رابطه (۶) به دست می‌آید.

$$MCC = \frac{(TP \times TN) - (FP \times FN)}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}} \quad (6)$$

جدول ۵: معیارهای ارزیابی مختلف در سیستم تشخیص نفوذ

نام معیار ارزیابی	مراجع	
Accuracy	[34,35,36,38,39,40,41,42,43,44,45,46,47,48,49,50,51,52,53,54,55,56,57,58,59,60,61,62,63,64,66,67,68,69,70,71,72,73,74,75,76]	
Precision	[34,114,38,39,40,42,44,45,47,48,49,50,52,53,57,58,59,60,65,67,72,74]	
TPR recall Sensitivity DR	[34,114,38,41,42,44,45,47,46,47,48,49,50,51,52,53,54,56,57,58,59,60,61,62,63,64,65,66,67,68,69,70,72,74,75]	
F-measure   F1-score	[33,34,114,38,41,42,44,45,46,47,48,49,50,52,53,54,59,60,61,65,67,70,74]	
FPR FAR	[33,34,35,36,38,40,45,46,48,49,51,58,61,62,63,65,66,67,68,69,70,75]	
time	زمان ساخت مدل	34
	training time	[38,45]
	testing time	35
	predicting time	51
	execute time	[33,40]
MCC	[38,39]	
number of selected features	[113,40,47,56,67,74,76]	
specificity  TNR	[38,39,48,49,58,69,72]	
AUC	[46,50,57]	
error rate Missed Rate (MR)	[48,67]	
G-mean	50	
95% CI MST	59	

زمان اجرای الگوریتم یکی از معیارهای کلیدی برای ارزیابی کارایی و عملکرد یک الگوریتم است. این معیار نشان‌دهنده مدت زمانی است که یک الگوریتم برای حل یک مسئله خاص نیاز دارد. در این معادلات:

- (TP) (True Positives) به تعداد حملات واقعی که به درستی شناسایی شده‌اند، اشاره دارد.

- (TN) (True Negatives) به تعداد دسترسی‌های مجاز که به درستی به‌عنوان غیرحمله شناسایی شده‌اند، اشاره دارد.

- (FP) (False Positives) به تعداد دسترسی‌های مجازی که به‌اشتباه به‌عنوان حمله شناسایی شده‌اند اشاره دارد.

- (FN) (False Negatives) به تعداد حملات واقعی که به‌اشتباه به‌عنوان دسترسی مجاز شناسایی شده‌اند، اشاره دارد.

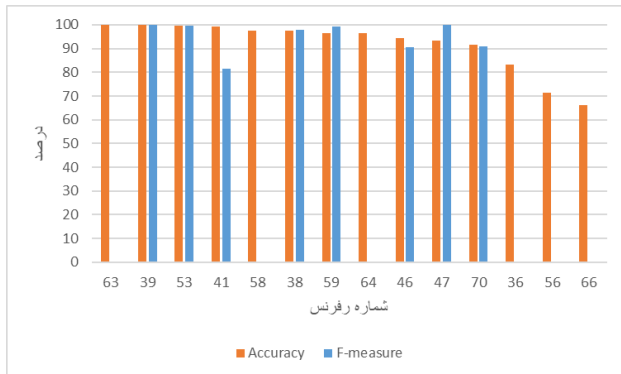
این معیارها کمک می‌کنند تا کارایی یک سیستم تشخیص نفوذ را ارزیابی کرده و نقاط قوت و ضعف آن را مشخص کنیم. جدول ۵ معیارهای ارزیابی مختلف در سیستم تشخیص نفوذ را نشان می‌دهد.

همه مقالات معیار ارزیابی Accuracy را بررسی کرده‌اند و اکثر مقالات معیارهای ارزیابی Precision، Recall و F-measure را دارند. معیارهای دیگری مانند false positive rate (FPR)، specificity، تعداد ویژگی‌های انتخابی، false alarm rate (FAR) و غیره نیز وجود دارند که کمتر به آن‌ها پرداخته شده است. معیار Accuracy روش‌های بررسی شده از ۶۶٪ تا ۱۰۰٪ است. معیارهای ارزیابی برای هر مجموعه داده‌ها در جدول ۶ آمده است. مقادیر خالی در جدول ۶ نشان‌دهنده این است که یا مقدار دقیقی برای معیار ارزیابی مشخص نشده است یا کلاً آن معیار در مقاله مذکور ارزیابی نشده است.

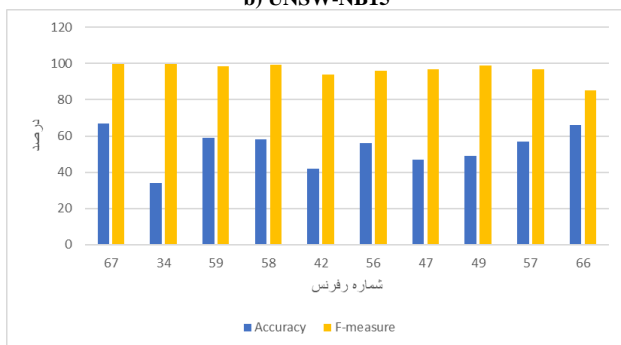
جدول ۶: معیارهای ارزیابی مقالات منتخب به تفکیک مجموعه داده‌ها

Dataset Name	مرجع	Accuracy	Precision	TPR recall Sensitivity DR	F-measure	FPR FAR	Dataset Name	مرجع	Accuracy	Precision	TPR recall Sensitivity DR	F-measure	FPR FAR
NSL-KDD	63	99.98		97.29		0.003527	CICIDS2017	49	97.96	98.46	98.98	98.97	0.451
	67	99.79	99.83	99.81	99.82	0.23		57	96.8	97.4	96.7		
	48	99.66	99.59	99.59	99.59	0.464		66	95.16		85.32		0.35
	62	99.44		99.36		0.6		34	99.9	99.9	99.9	99.9	0.001
	41	99.4	98.38	98.02	98.14			47	98.51	91.77	0.97	93.9	
	75	99.22		99.1		0.0064		55					
	58	99.16		99.38		0.015		72			95.91		
	66	98.99		88.82		0.3		KDD CUP 1999	46	99.82		99.7	97.23
	54	98.95	96	99.4	97.7		34		99.8	99.8	99.8	99.8	0.001
	61	98.51		98.56	99.07	0.625	68		99.6	88.5	96.2	92.2	0.4
	47	98.34	98.51	99.93	99.87		58		98.05		99.59		0.029
	40	97.8		98.76			70		96		98.2		0.076
	42	97.5	94.5	94.8	92.1		35		95.03		95.23		1.65
	46	96.93		97.5	91.9	0.112	33		90.28			77	0.005
	43	90.73					45						
	70	88.3		86.6	88.2	0.088	50						
	76	87.59					71				83.49		
	65	87.53	87.68	67.2	75.74	0.05	BOT_IOT	51	99.99	99.99	99.99	99.99	
	60	87.34	89.09	87.34	88.21			64	98.9				
	44	84.8649	88.12	60.17	60.93			46	97.37		99.7	96.7	
73	80					37			93.7	97.7	95.6		
69		100				CSE-CIC-IDS 2018	38	99.89	99.914	99.125	99.214	0.013	
50							47	99.95	0.9982	0.9852	0.98516		
45							60	99.87	99.88	99.87	99.88		
37		90.8	94.8	92.7		CIRA-CIC-DoH	47	99.43	99.94	99.94	99.915		
UNSW-NB15	63	99.86		98.51		1.021394	ISCX2012	64	98.94				
	39	99.79	98.99	98.49	99.88			58	99.14		98.98		0.004
	53	99.72	99.27	99.84	99.56	0.16	74	99.66	99.8	99.57	99.69		
	41	99.2	83.57	83.29	81.66		ADFA-LD	72	99.5	99.5	99.5	99.5	0.001
	58	97.63		98.18		0.033	AWID	34			94.44		
	38	97.535	97.674	98.945	97.995	0.105	BGFD1, BGFD2, BGFD3	50					
	59	96.5	98.6	100	99.3		CEC2021	44					
	64	96.48					CICDDOS 2019	47	97.19	84.24	96.8	96.979	
	46	94.45		96.7	90.7	11.1	CIDD	45					
	47	93.32	92.31	100	100		CSIC-2010	49	97.56	97.458	98.789	98.369	50.254
	70	91.7		89.4	90.9	0.034	Intrusion detection 2018	59	98	96.1	100	98	
	36	83.12				3.7	LUFlow Network	47	99%	0.9806	0.9818	0.9812	
	56	71.5429		80.5779			N-BaIoT	47	99%	0.9959	0.9939	0.994	

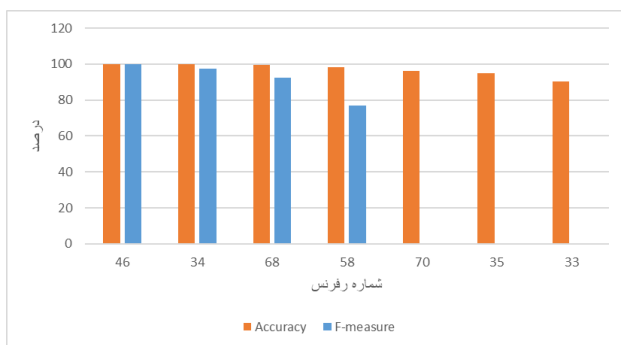
	66	66	64.9	3.85	Malmem2_022	47	100%	1	0.937	0.92893		
CICIDS2017	67	99.91	99.99	99.92	99.95	0.1	Phishing Legitimate	47	96%	0.968	0.9958	0.9959
	59	99.3	100	98.5	99.2		SWaT	52	99.65	97.16	96.78	96.97
	58	99.23		99.26		0.013	WSN-DS	52	99%	92.87	96.44	94.62
	42	99.1	97.2	94.1	94.5		WUSTL-IIoT	44		98.72	96.36	97.27
	56	98.71		96.17								



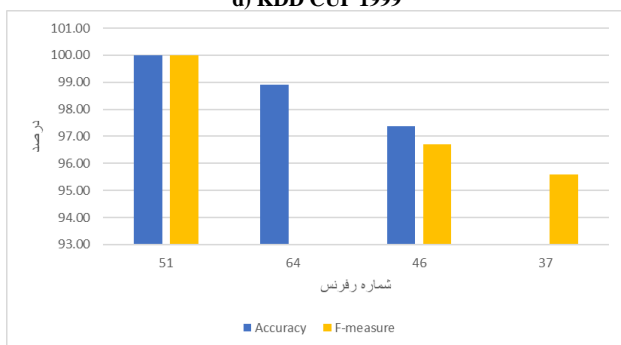
b) UNSW-NB15



c) CICIDS2017



d) KDD CUP 1999

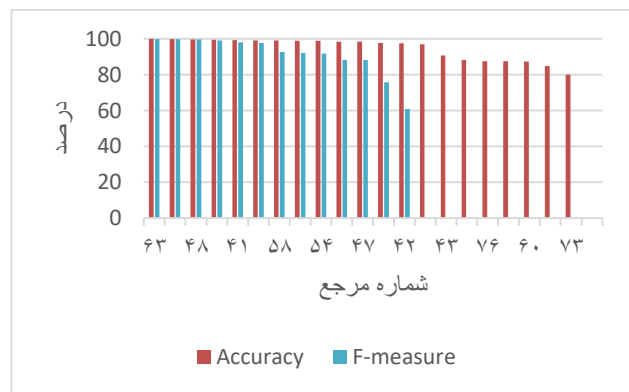


e) BOT\_IOT

شکل ۶: درصد دقت تشخیص و امتیاز F را برای مجموعه داده‌های a) NSL-KDD, b) UNSW-NB15, c) CICIDS2017, d) KDD CUP 1999, e) BOT\_IOT

در مجموعه داده NSL-KDD بیشترین دقت تشخیص ۹۹.۹۸٪ [۶۳] است که از الگوریتم کلاغ برای انتخاب ویژگی استفاده کرده‌است و کمترین مقدار دقت تشخیص ۸۰٪ [۷۳] است که الگوریتم نهنگ اصلاح شده را به‌عنوان انتخاب ویژگی به‌کار برده‌اند. در حالی که [۶۹] با ترکیب دو الگوریتم زنبور و ازدحام ذرات تطبیقی توانسته‌است مقدار Precision را به ۱۰۰٪ برساند به‌عبارتی، روش پیشنهادی آن‌ها همه حملات را به‌درستی تشخیص می‌دهد. مقدار recall در بین مقالاتی که روی مجموعه داده NSL-KDD آزمایش انجام داده‌اند بین ۹۹.۹۳٪ [۴۷] و ۶۰.۱۷٪ [۴۴] است. بنابراین استفاده از الگوریتم ازدحام salp مبتنی بر مخالفت توانسته بهترین مقدار recall را داشته‌باشد یعنی تا حد زیادی در شناسایی تمامی موارد مثبت، موفق بوده‌است. همچنین مقدار معیار F-measure برای این روش برابر با ۹۹.۸۷٪ است [۴۷].

کمترین مقدار F-measure متعلق به Hybrid Breeding Optimization (HBO) است که برابر با ۶۰.۹۳٪ [۴۴] است. تعداد مقالاتی که FPR را بررسی کردند تقریباً کم بوده‌است. بهترین نتیجه این معیار برای الگوریتم کلاغ [۶۳] با مقدار ۰.۰۰۴ است و بدترین نتیجه را الگوریتم شیر مورچه [۶۱] با مقدار ۰.۶۲۵ داشته‌است. شکل ۶ نمودار درصد دقت تشخیص و F-measure را برای مجموعه داده‌هایی که تعداد فراوانی آن‌ها بالاتر از ۴ است، نشان می‌دهد. چنانچه مرجعی در این شکل نباشد به این دلیل است که درصد مشخصی از معیارها در مقاله مربوطه بیان نشده‌است و یا مقدار آن را به‌صورت نمودار نشان داده‌است که به‌دلیل عدم وجود مقدار دقیق از آن صرف‌نظر کردیم.



a) NSL-KDD

در مجموعه داده UNSW-NB15 بیشترین دقت تشخیص ۹۹.۸۶٪ [۶۳] است که از الگوریتم کلاغ برای انتخاب ویژگی استفاده کرده است و کمترین مقدار دقت تشخیص ۶۶٪ [۶۶] است که الگوریتم ژنتیک چندهدفه را به کار برده اند. مقدار بیشترین Precision متعلق به الگوریتم کرم شبتاب باینری [۵۳] است و کمترین مقدار آن ۸۳.۵۷٪ است که برای الگوریتم کلاغ [۴۱] است. درحالی که [۵۹] با الگوریتم ژنتیک اصلاح شده توانسته است مقدار recall را به ۱۰۰٪ برساند و الگوریتم ژنتیک چندهدفه [۶۶] کمترین مقدار recall را دارد (۶۶٪). و [۴۷] با الگوریتم ازدحام سالپ مبتنی بر مخالفت توانسته مقدار recall و F-measure را به ۱۰۰٪ برساند و [۴۱] کمترین مقدار F-measure را در مجموعه داده UNSW-NB15 دارد. بهترین نتیجه معیار FPR برای الگوریتم خفاش چندهدفه [۵۸] با مقدار ۰.۰۳۳ و بدترین نتیجه برای الگوریتم کبوتر [۴۶] با ۱۱.۱ است.

دقت در مجموعه داده CICIDS2017 به طور کلی بالا است رنج درصد دقت تشخیص در این مجموعه داده از ۹۵.۱۶٪ تا ۹۹.۹۱٪ است. بیشترین دقت مربوط به الگوریتم ازدحام ذرات [۶۷] با ۹۹.۹۱٪ است البته این روش بیشترین مقدار recall را با ۹۹.۹۲٪ از بین همه متدهای این قسمت داراست و مقدار Precision آن برابر با ۹۹.۹۹٪ است. همچنین مقدار Precision در روش های مختلف برای CICIDS2017 بین ۹۱.۷۷٪ تا ۱۰۰٪ است که [۵۹] با الگوریتم ژنتیک اصلاح شده بیشترین مقدار Precision را دارد. مقدار F-measure از ۹۲.۹۰٪ تا ۹۹.۵٪ است که درصد خوبی از تشخیص حملات و ترافیک نرمال را نشان می دهد. بهترین نتیجه معیار FPR برای الگوریتم خفاش [۳۴] با مقدار ۰.۰۰۱ و پایین ترین نتیجه برای الگوریتم (modified manta-ray foraging optimization) MMFO [۴۹] با ۰.۴۵۱ است.

دقت تشخیص نفوذ برای مجموعه داده KDD CUP 1999 از ۹۰.۲۸٪ تا ۹۹.۸۲٪ است. الگوریتم کبوتر [۴۶] بیشترین و الگوریتم کرم شبتاب [۳۳] کمترین مقدار دقت را دارد. مقدار معیار Precision برای الگوریتم خفاش [۳۴]، ۹۹.۸٪ و برای ازدحام ذرات [۶۸] ۸۸.۵٪ است. مقدار معیارهای recall و F-measure در الگوریتم خفاش [۳۴]، برابر با ۹۹.۸٪ است که بیشترین مقدار این معیارها در مجموعه داده KDD CUP 1999 می باشد. کمترین مقدار معیار recall در این مجموعه داده [۷۱] ۸۳.۴۹٪ است. الگوریتم کرم شبتاب [۳۳] کمترین مقدار معیار F-measure را دارد.

الگوریتم ژنتیک [۵۱] روی مجموعه داده BOT\_IOT دارای بیشترین مقدار برای معیارهای دقت، صحت، فراخوانی و F-measure است و البته متدهای دیگر [۶۴، ۴۶، ۳۷] نیز درصدهای بالای ۹۳٪ را نشان می دهند. همچنین [۴۷، ۶۰، ۳۸] درصدهای بالای ۹۸٪ را برای معیارهای مختلف در مجموعه CSE-CIC-IDS 2018 اعلام کرده اند.

### ۵.۳. تعداد ویژگی های انتخاب شده

یکی از پارامترهایی که در انتخاب ویژگی اهمیت دارد تعداد ویژگی های انتخاب شده است. این تعداد به مجموعه ای از ویژگی ها اشاره دارد که از بین ویژگی های اصلی مجموعه داده انتخاب شده اند تا مدل یادگیری ماشین با استفاده از آن ها ساخته شود. به طور کلی، هیچ عدد خاصی برای تعداد بهینه ویژگی ها وجود ندارد. بهترین راه برای انتخاب تعداد ویژگی ها، آزمایش با مقادیر مختلف و ارزیابی عملکرد مدل بر روی داده ها است. تعداد ویژگی های انتخاب شده بسته به روش انتخاب ویژگی می تواند متفاوت باشد. جدول ۷ و شکل ۷ تعداد ویژگی های انتخاب شده را بر اساس مجموعه داده ها در روش های مختلف ارائه می دهند.

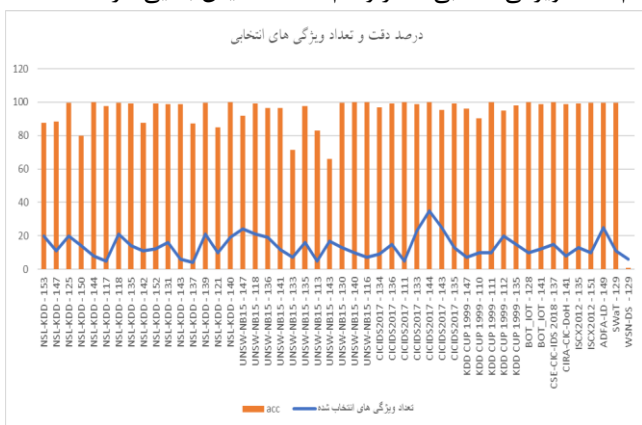
جدول ۷: تعداد ویژگی های انتخابی بر اساس مجموعه داده ها

Dataset Name	مرجع	تعداد ویژگی های انتخاب شده
NSL-KDD	40	11
	41	11
	44	21
	45	17
	48	6
	54	16
	58	12
	60	20
	62	20
	63	21
	65	14
	66	19
	67	10
	70	5
	73	8
75	14	
76	4	
UNSW-NB15	36	16
	39	24
	41	7
	53	19
	56	11.9
	58	13
	59	7
	63	21
	64	10
66	17	
70	5	
CICIDS2017	34	13

انتخاب‌کند که کمترین تعداد ویژگی بین روش‌های بررسی شده است. بعد از آن الگوریتم کبوتر [۷۰] با ۵ ویژگی انتخابی است. سپس الگوریتم تکاملی تفاضلی [۴۸] با ۶ ویژگی و الگوریتم بهینه‌سازی نهنگ اصلاح‌شده [۷۳] با ۸ ویژگی انتخابی، هستند.

برای مجموعه داده UNSW-NB15، الگوریتم‌های کبوتر [۷۰]، الگوریتم‌های ژنتیک تغییر یافته (MGA) و کلاغ [۴۱، ۵۹] به ترتیب ۵ و ۷ ویژگی را انتخاب کردند. Multi-Objective Evolutionary Feature Selection (MOEFS) [۵۷] و ژنتیک تغییر یافته [۵۹] به ترتیب ۵ و ۹ ویژگی از مجموعه داده CICIDS2017 را انتخاب کرده‌اند.

در مجموعه داده KDD CUP 1999 ابتدا الگوریتم کبوتر [۷۰] با ۷ ویژگی، سپس الگوریتم‌های کرم شب‌تاب و خفاش [۳۳، ۳۴] کمترین تعداد ویژگی را انتخاب کرده‌اند. بنابراین الگوریتم کبوتر [۷۰] برای سه مجموعه داده UNSW-NB15، NSL-KDD و KDD CUP 1999 کمترین تعداد ویژگی‌ها را انتخاب کرده‌است. برای مقایسه بیشتر شکل ۸ تعداد ویژگی‌های انتخابی و دقت تشخیص را نشان می‌دهد. همان‌طور که مشخص است در مجموعه داده NSL-KDD الگوریتم NSGA-II [۶۶] هم تعداد ویژگی انتخابی کمتر و هم دقت تشخیص بالایی دارد.



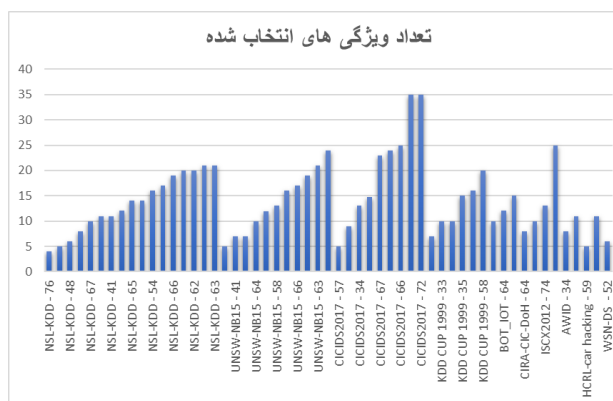
شکل ۸: تعداد ویژگی‌های انتخابی و دقت تشخیص

#### ۴.۵. توابع هدف

توابع هدف در الگوریتم‌های فراابتکاری نقش حیاتی در هدایت فرآیند بهینه‌سازی ایفای می‌کنند. این توابع، معیاری برای ارزیابی کیفیت هر راه‌حل پیشنهادی در فضای جستجو فراهم می‌کنند. انتخاب و طراحی مناسب تابع هدف می‌تواند تأثیر زیادی بر عملکرد و کارایی الگوریتم فراابتکاری داشته‌باشد.

معیار دقت نقش مهمی در تابع هدف مسئله تشخیص نفوذ دارد و مقالات زیادی [۳۳، ۳۸، ۴۰، ۴۲، ۴۳، ۴۹، ۵۱، ۵۳، ۵۴، ۵۵، ۵۷، ۵۹، ۶۱، ۶۵، ۶۸، ۷۰، ۷۱، ۷۲، ۷۳، ۷۶] از آن به‌تنهایی برای تابع هدف استفاده کرده‌اند. و تعدادی معیار دقت را با معیارهای دیگر ترکیب کرده‌اند. مانند [۷۴، ۶۶، ۴۱، ۷۵] که دقت و تعداد ویژگی انتخابی را ترکیب کردند. [۶۴] از رابطه ۷ برای ترکیب دقت و ماتریس همبستگی بین ویژگی‌ها به‌عنوان تابع هدف استفاده کردند.

	55	24
	56	15
	57	5
	58	35
	59	9
	66	25
	67	23
	72	35
KDD CUP 1999	33	10
	34	10
	35	15
	45	16
	58	20
	70	7
BOT_IOT	51	10
	64	12
CSE-CIC-IDS 2018	60	15
CIRA-CIC-DoH	64	8
ISCX2012	58	10
	74	13
ADFA-LD	72	25
AWID	34	8
CIDD	45	11
HCRL-car hacking	59	5
SWaT	52	11
WSN-DS	52	6



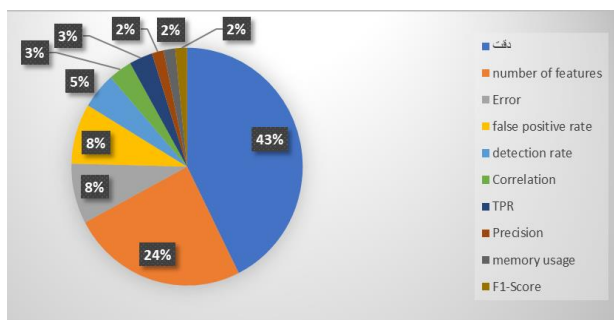
شکل ۷: تعداد ویژگی‌های انتخابی بر اساس مجموعه داده‌ها در روش‌های مختلف

الگوریتم گرگ خاکستری چندهدفه [۷۶] توانسته‌است ۴ ویژگی از

مجموعه

$$F_i = \frac{A_i + (1 - M_i)}{2} \quad (7) \quad \text{داده NSL-KDD}$$





شکل ۹: درصد استفاده از پارامترهای مختلف در توابع هدف مطالعات انجام شده

## ۶. چالش‌ها و شکاف‌های تحقیقاتی انتخاب ویژگی مبتنی بر الگوریتم‌های فراابتکاری

در این بخش به سوال سوم «چالش‌ها و شکاف‌های تحقیقاتی در الگوریتم‌های فراابتکاری موجود برای انتخاب ویژگی در سیستم‌های تشخیص نفوذ چیست؟» پاسخ داده می‌شود.

انتخاب ویژگی یکی از مراحل حیاتی در پیش‌پردازش داده‌ها و ساخت مدل‌های یادگیری ماشین است. این فرآیند با هدف کاهش تعداد ویژگی‌ها، حذف ویژگی‌های زائد و بهبود دقت مدل انجام می‌شود. الگوریتم‌های فراابتکاری به دلیل توانایی‌شان در جستجوی فضای بزرگ و پیچیده ویژگی‌ها، به‌طور گسترده‌ای در انتخاب ویژگی استفاده می‌شوند. با این حال، این رویکردها با چالش‌ها و شکاف‌های تحقیقاتی متعددی مواجه هستند که در ادامه به برخی از آن‌ها اشاره می‌کنم:

**پیچیدگی محاسباتی و زمان اجرا:** الگوریتم‌های فراابتکاری معمولاً به زمان محاسباتی زیادی نیاز دارند، به‌خصوص زمانی که داده‌ها دارای تعداد زیادی ویژگی باشند. این مسئله می‌تواند کارایی را کاهش دهد و در کاربردهای عملی محدودیت ایجاد کند. با توجه به بررسی‌های انجام‌شده، تعداد مقالات کمی به زمان ساخت مدل، زمان اجرا، زمان آموزش و تست توجه کردند و این پارامترها را جهت ارزیابی نتایج در نظر نگرفته‌اند.

**بهینه محلی:** الگوریتم‌های فراابتکاری ممکن است در بهینه‌های محلی گیر کنند و نتوانند به بهینه جهانی دست‌یابند. این مشکل به‌ویژه در مسائل پیچیده و با فضای جستجوی بزرگ مشهود است. اکثر مقالات سعی داشتند جستجوی محلی را بهبود دهند و تعداد کمی از الگوریتم‌های پایه و استاندارد استفاده کردند. البته بهینه‌کردن جستجوی محلی الگوریتم‌های فراابتکاری همچنان موضوع باز تحقیقات امروزی است.

**تعیین پارامترها:** الگوریتم‌های فراابتکاری معمولاً دارای پارامترهای متعددی هستند که تنظیم دقیق آن‌ها برای دستیابی به عملکرد بهینه ضروری است. تنظیم نادرست پارامترها می‌تواند منجر به عملکرد ضعیف الگوریتم شود. تعداد کمی از تحقیقات مذکور، پارامترهای الگوریتم فراابتکاری خود را با روش‌های مختلف به‌دقت تنظیم کرده‌اند.

که  $A_i$  دقت و  $M_i$  ماتریس همبستگی است. تابع هدف در [۶۲] ترکیبی از دقت و نرخ تشخیص است.

جدول ۸ خلاصه‌ای از پارامترهای توابع هدف را نشان می‌دهد. دقت بیشترین کاربرد را در تابع هدف تحقیقات گذشته دارد پس از آن تعداد ویژگی‌های انتخابی، میزان خطا و false positive rate هستند.

که  $A_i$  دقت و  $M_i$  ماتریس همبستگی است. تابع هدف در [۶۲] ترکیبی از دقت و نرخ تشخیص است. جدول ۸ خلاصه‌ای از پارامترهای توابع هدف را نشان می‌دهد. دقت بیشترین کاربرد را در تابع هدف تحقیقات گذشته دارد، پس از آن تعداد ویژگی‌های انتخاب‌شده، میزان خطا و false positive rate هستند.

جدول ۸: توابع هدف مقالات بررسی شده

پارامترها	مراجع
دقت	[۳۳، ۲۸، ۴۰، ۴۲، ۴۳، ۴۵، ۴۸، ۴۹، ۵۱، ۵۲، ۵۳، ۵۴، ۵۵، ۵۷، ۵۹، ۶۱، ۶۴، ۶۵، ۶۸، ۷۰، ۷۱، ۷۲، ۷۳، ۷۴، ۷۵، ۷۶]
Precision	39
Error	36, 47, 50, 56, 58
Correlation	34, 64
detection rate	35, 62, 75
false positive rate	35, 36, 44, 46, 58
number of features	36, 37, 41, 44, 46, 47, 48, 56, 58, 60, 63, 66, 67, 74, 75
TPR	44, 46,
memory usage	50
F1-Score	52

شکل ۹ نشان می‌دهد که ۴۳٪ از مطالعات از پارامتر دقت برای تابع هدف (به صورت تک‌هدفه یا چندهدفه) استفاده کردند که ۷۷٪ آن فقط پارامتر دقت را در نظر گرفته‌اند و مابقی مطالعات، ترکیب پارامتر دقت و دیگر پارامترها را به کار برده‌اند. دومین پارامتر پر استفاده، number of features است که ۲۴٪ از مطالعات به آن پرداخته‌اند. false positive rate و میزان خطا به‌طور مساوی ۸٪ و پارامتر detection rate ۵٪ مطالعات و پارامترهای Correlation و TPR به‌طور یکسان ۳٪ از توابع هدف مطالعات را به خود اختصاص داده‌اند. دیگر پارامترها هر کدام ۲٪ در توابع هدف مشارکت داشته‌اند.

فراابتکاری مختلف در تشخیص نفوذ را بررسی کردیم. الگوریتم‌های فراابتکاری مبتنی بر هوش جمعی پر استفاده‌ترین نوع الگوریتم فراابتکاری است. ۷۶٪ از مطالعات از الگوریتم‌های هوش جمعی استفاده کردند.

تقریباً ۲۸٪ مطالعات به مجموعه داده KDD99، که حدود ۲۰ سال پیش ایجاد شده است، وابسته هستند. این مجموعه داده به طور گسترده برای ارزیابی الگوریتم‌های یادگیری ماشین، به ویژه در تشخیص ترافیک مخرب از ترافیک عادی، استفاده شده است. با این حال، KDD99 دیگر نمی‌تواند ویژگی‌های ترافیک داده‌های مدرن را به خوبی منعکس کند. در حال حاضر، ترافیک چندرسانه‌ای مانند صدا و ویدیو اصلی‌ترین منبع درآمد ارائه‌دهندگان خدمات است و همچنین وسیله‌ای برای پوشش ترافیک مخرب محسوب می‌شود. این نوع ترافیک در مجموعه داده KDD99 گنجانده نشده است [۱۰].

تقریباً ۲۴٪ از مطالعات معیار دقت را برای ارزیابی نتایج متدشان انتخاب کردند و ۲۳٪ از آن‌ها  $TPR|recall|Sensitivity|DR$  را بررسی کرده‌اند. در حالی که معیارهای  $F1-score$  |  $FPR|FAR$ ،  $F-measure$  و Precision فقط ۱۳٪ بررسی نتایج مطالعات را به خود اختصاص داده‌اند. در صورتی که برای بررسی کامل نتایج باید این معیارها نیز تجزیه و تحلیل شوند. همچنین زمان اجرای الگوریتم در ۴٪ مطالعات آمده است که بسیار کم می‌باشد در حالی که پیچیدگی زمان اجرای الگوریتم یک چالش مهم برای مسأله تشخیص نفوذ است.

معیار دقت برای توابع هدف الگوریتم‌های مختلف، زیاد به کار برده شده است به طوری که ۴۲٪ از توابع هدف مطالعات شامل دقت بوده است که ۳۳٪ فقط معیار دقت را برای انتخاب ویژگی انتخاب کردند و ۱۰٪ از ترکیب دقت با دیگر پارامترها تشکیل شده است. البته بهتر است پارامترهای مهمی مانند  $RECALL$ ،  $PRECISION$ ،  $FPR$  و  $F1-Score$  نیز در تابع هدف مشارکت داشته باشند. توابع هدف در الگوریتم‌های فراابتکاری نقش کلیدی در هدایت فرآیند بهینه‌سازی دارند. طراحی مناسب و انتخاب صحیح تابع هدف می‌تواند تأثیر زیادی بر عملکرد و کارایی الگوریتم داشته باشد. با توجه به نوع مسئله و ویژگی‌های فضای جستجو، می‌توان تابع هدف مناسبی را انتخاب و بهینه‌سازی را به نحو مؤثری انجام داد.

برای کارهای آینده بررسی الگوریتم‌های طبقه‌بندی جهت تشخیص نفوذ نیاز است. مطالعه الگوریتم‌های یادگیری ماشین و طبقه‌بندی آن‌ها در سیستم‌های تشخیص نفوذ مؤثر خواهد بود. برای ادامه کار مجموعه داده‌هایی که شامل ترافیک چندرسانه‌ای و انواع جدید حملات مانند حملات مبتنی بر هوش مصنوعی هستند را بررسی خواهیم کرد. همچنین بررسی و ارزیابی الگوریتم‌های فراابتکاری جدید و نوآورانه می‌توانند به بهبود انتخاب ویژگی کمک کنند. بررسی تکنیک‌های ترکیبی روش‌های فیلتر و لاف انتخاب ویژگی نیز می‌تواند مؤثر باشد. انجام تحلیل‌های مقایسه‌ای بین روش‌های مختلف انتخاب ویژگی و تشخیص نفوذ در محیط‌های ابری مختلف و بررسی عملکرد آن‌ها در

**تعامل و همبستگی ویژگی‌ها:** انتخاب ویژگی‌هایی که به شدت با هم همبسته هستند یا برعکس، تعاملات پیچیده‌ای با هم دارند، می‌تواند عملکرد مدل را تحت تأثیر قرار دهد. الگوریتم‌های فراابتکاری به طور معمول نمی‌توانند این تعاملات پیچیده را به خوبی مدیریت کنند. بنابراین می‌توان از پارامتر همبستگی ویژگی‌ها در تابع هدف الگوریتم فراابتکاری مانند [۳۴، ۶۴] استفاده کرد.

**ناتوانی در تضمین بهترین راه‌حل:** الگوریتم‌های فراابتکاری تضمینی برای یافتن بهترین راه‌حل ندارند و تنها می‌توانند به یک راه‌حل تقریباً بهینه دست یابند. این مسئله می‌تواند در مسائل حساس به دقت مانند تشخیص نفوذ مشکل‌ساز شود. بنابراین بررسی تعداد معیارهای ارزیابی همزمان می‌تواند ما را به راه حل بهینه نزدیک‌تر کند.

**پیچیدگی تابع هدف:** تابع هدف باید به گونه‌ای طراحی شود که الگوریتم بتواند از بهینه‌های محلی فرار کرده و به سمت بهینه جهانی حرکت کند. از طرفی تابع هدف نباید بیش از حد پیچیده باشد، زیرا محاسبه مکرر آن می‌تواند زمان بر باشد. در مسائل چندهدفه، تعیین راه‌حلی که تعادل مناسبی بین اهداف مختلف برقرار کنند، یک چالش است.

#### شکاف‌های تحقیقاتی:

**توسعه الگوریتم‌های ترکیبی:** یکی از زمینه‌های تحقیقاتی جذاب، ترکیب الگوریتم‌های فراابتکاری با دیگر تکنیک‌ها به منظور بهره‌برداری از مزایای هر دو روش است. توسعه الگوریتم‌های ترکیبی می‌تواند به بهبود کارایی و دقت انتخاب ویژگی کمک کند.

**روش‌های تطبیقی و خودتنظیمی:** تحقیقات بیشتری در زمینه توسعه روش‌های تطبیقی و خودتنظیمی برای تنظیم پارامترهای الگوریتم‌های فراابتکاری مورد نیاز است. این روش‌ها می‌توانند به صورت خودکار بهترین پارامترها را برای هر مسئله خاص تنظیم کنند.

**الگوریتم‌های مقاوم در برابر بهینه‌های محلی:** نیاز به توسعه الگوریتم‌هایی است که بتوانند از بهینه‌های محلی فرار کنند و به بهینه‌های جهانی نزدیک‌تر شوند. استفاده از تکنیک‌هایی مانند حافظه و مکانیزم‌های جستجوی چندگانه می‌تواند در این زمینه مؤثر باشد.

**مدیریت تعاملات پیچیده ویژگی‌ها:** الگوریتم‌های جدیدی که بتوانند تعاملات پیچیده بین ویژگی‌ها را به خوبی مدیریت کنند و از اطلاعات تعاملی برای انتخاب ویژگی‌ها بهره‌برند، یک شکاف تحقیقاتی مهم است.

## 7. نتیجه‌گیری و راهکارهای آینده

سیستم‌های تشخیص نفوذ موجود، می‌تواند به طور مؤثر رفتارهای ناهنجاری را در شبکه شناسایی کند. با این حال، هنوز هم دارای نرخ تشخیص پایین و نرخ هشدار کاذب بالا به خصوص برای حملاتی با رکوردهای کمتر است و البته شکاف‌های تحقیقاتی وجود دارد که بررسی نکردیم. ما فقط ۴۳ مقاله انتخاب ویژگی مبتنی بر الگوریتم‌های

and Information Networks, 2, 107-119. <https://doi.org/10.1007/s41650-017-0033-7>.

12. Saadouni, R., Gherbi, C., Aliouat, Z., Harbi, Y., & Khacha, A. (2024). Intrusion detection systems for IoT based on bio-inspired and machine learning techniques: a systematic review of the literature. *Cluster Computing*, 1-27. <https://doi.org/10.1007/s10586-024-04388-5>.
13. Kalimuthan, C., & Renjit, J. A. (2020). Review on intrusion detection using feature selection with machine learning techniques. *Materials Today: Proceedings*, 33, 3794-3802. <https://doi.org/10.1016/j.matpr.2020.06.218>.
14. Agrawal, P., Abutarboush, H. F., Ganesh, T., & Mohamed, A. W. (2021). Metaheuristic algorithms on feature selection: A survey of one decade of research (2009-2019). *Ieee Access*, 9, 26766-26791. <https://doi.org/10.1109/ACCESS.2021.3056407>.
15. Nssibi, M., Manita, G., & Korbaa, O. (2023). Advances in nature-inspired metaheuristic optimization for feature selection problem: A comprehensive survey. *Computer Science Review*, 49, 100559. <https://doi.org/10.1016/j.cosrev.2023.100559>.
16. Pham, T. H., & Raahemi, B. (2023). Bio-inspired feature selection algorithms with their applications: a systematic literature review. *IEEE Access*. <https://doi.org/10.1109/ACCESS.2023.3272556>.
17. Bello, S. A., Oyedele, L. O., Akinade, O. O., Bilal, M., Delgado, J. M. D., Akanbi, L. A., ... & Owolabi, H. A. (2021). Cloud computing in construction industry: Use cases, benefits and challenges. *Automation in Construction*, 122, 103441. <https://doi.org/10.1016/j.autcon.2020.103441>.
18. Lansky, J., Ali, S., Mohammadi, M., Majeed, M. K., Karim, S. H. T., Rashidi, S., ... & Rahmani, A. M. (2021). Deep learning-based intrusion detection systems: a systematic review. *IEEE Access*, 9, 101574-101599. <https://doi.org/10.1109/ACCESS.2021.3097247>.
19. Melvin, A. A. R., Kathrine, G. J. W., Ilango, S. S., Vimal, S., Rho, S., Xiong, N. N., & Nam, Y. (2022). Dynamic malware attack dataset leveraging virtual machine monitor audit data for the detection of intrusions in cloud. *Transactions on Emerging Telecommunications Technologies*, 33(4), e4287. <https://doi.org/10.1002/ett.4287>.
20. Rostami, M., Berahmand, K., Nasiri, E., & Forouzandeh, S. (2021). Review of swarm intelligence-based feature selection methods. *Engineering Applications of Artificial Intelligence*, 100, 104210. <https://doi.org/10.1016/j.engappai.2021.104210>.
21. Solorio-Fernández, S., Carrasco-Ochoa, J. A., & Martínez-Trinidad, J. F. (2020). A review of unsupervised feature selection methods. *Artificial Intelligence Review*, 53(2), 907-948. <https://doi.org/10.1007/s10462-019-09682-y>.
22. Abdel-Basset, M., Abdel-Fatah, L., & Sangaiah, A. K. (2018). Metaheuristic algorithms: A comprehensive review. *Computational intelligence for multimedia big data on the cloud with engineering applications*, 185-231. <https://doi.org/10.1016/B978-0-12-813314-9.00010-4>.
23. Osaba, E., Villar-Rodriguez, E., Del Ser, J., Nebro, A. J., Molina, D., LaTorre, A., ... & Herrera, F. (2021). A tutorial on the design, experimentation and application of metaheuristic algorithms to real-world optimization

شرایط مختلف و با داده‌های مختلف نیز جهت کار آینده پیشنهاد می‌شود. البته تحقیق در مورد روش‌های بهینه‌سازی پارامترها برای الگوریتم‌های انتخاب ویژگی و تشخیص نفوذ، به‌ویژه در محیط‌های ابری با مقیاس بزرگ مورد نیاز است.

## منابع

1. Thakkar, A., & Lohiya, R. (2022). A survey on intrusion detection system: feature selection, model, performance measures, application perspective, challenges, and future research directions. *Artificial Intelligence Review*, 55(1), 453-563. <https://doi.org/10.1007/S10462-021-10037-9>.
2. Ghaffari, A., & Hossinnezhad, R. (2022). Intrusions detection system in the cloud computing using heterogeneity detection technique. *Intelligent Multimedia Processing and Communication Systems (IMPCS)*, 3(1), 37-46. <https://srb.sanad.iau.ir/en/Article/903516>
3. Roknaldini, M., & Noroozi, E. (1402). Presenting A Hybrid Method of Deep Neural Networks to Prevent Intrusion in Computer Networks. *Intelligent Multimedia Processing and Communication Systems (IMPCS)*, 4(4), 57-65. <http://sanad.iau.ir/fa/Article/903465>
4. Kaya, İ. (2009). RETRACTED: A genetic algorithm approach to determine the sample size for control charts with variables and attributes. <https://doi.org/10.1016/j.eswa.2008.12.011>
5. Mohammad Akhlaghpour. (2023, apr). Using the Modified Colonial Competition Algorithm to Increase the Speed and Accuracy of the Intelligent Intrusion Detection System. (IMPCS), (pp. 1-10). <https://doi.org/10.1016/j.cie.2019.106040>
6. Khanduja, N., & Bhushan, B. (2021). Recent advances and application of metaheuristic algorithms: A survey (2014–2020). *Metaheuristic and evolutionary computation: algorithms and applications*, 207-228. [https://doi.org/10.1007/978-981-15-7571-6\\_10](https://doi.org/10.1007/978-981-15-7571-6_10).
7. Dokeroglu, T., Deniz, A., & Kiziloz, H. E. (2022). A comprehensive survey on recent metaheuristics for feature selection. *Neurocomputing*, 494, 269-296. <https://doi.org/10.1016/j.neucom.2022.04.083>.
8. Maldonado, J., Riff, M. C., & Neveu, B. (2022). A review of recent approaches on wrapper feature selection for intrusion detection. *Expert Systems with Applications*, 198, 116822. <https://doi.org/10.1016/j.eswa.2022.116822>.
9. Sharma, S., Kumar, V., & Dutta, K. (2024). Multi-objective optimization algorithms for intrusion detection in IoT networks: A systematic review. *Internet of Things and Cyber-Physical Systems*. <https://doi.org/10.1016/j.iotcps.2024.01.003>.
10. Reddy, D. K. K., Nayak, J., Behera, H. S., Shanmuganathan, V., Viriyasivatav, W., & Dhiman, G. (2024). A Systematic Literature Review on Swarm Intelligence Based Intrusion Detection System: Past, Present and Future. *Archives of Computational Methods in Engineering*, 1-68. <https://doi.org/10.1007/s11831-023-10059-2>.
11. Balasaraswathi, V. R., Sugumaran, M., & Hamid, Y. (2017). Feature selection techniques for intrusion detection using non-bio-inspired and bio-inspired optimization algorithms. *Journal of Communications*

- method for network intrusion detection. *Computers & Security*, 102, 102164. <https://doi.org/10.1016/j.cose.2020.102164>
37. Rani, B. S., Vairamuthu, S., & Subramanian, S. (2024). Archimedes Fire Hawk Optimization Enabled Feature Selection with Deep Maxout for Network Intrusion Detection. *Computers & Security*, 103751. <https://doi.org/10.1016/j.cose.2024.103751>
  38. Turukmane, A. V., & Devendiran, R. (2024). M-MultiSVM: An efficient feature selection assisted network intrusion detection system using machine learning. *Computers & Security*, 137, 103587. <https://doi.org/10.1016/j.cose.2023.103587>
  39. Aljehane, N. O., Mengash, H. A., Hassine, S. B., Alotaibi, F. A., Salama, A. S., & Abdelbagi, S. (2024). Optimizing intrusion detection using intelligent feature selection with machine learning model. *Alexandria Engineering Journal*, 91, 39-49. <https://doi.org/10.1016/j.aej.2024.01.073>
  40. Abualigah, L., Ahmed, S. H., Almomani, M. H., Zitar, R. A., Alsoud, A. R., Abuhaija, B., ... & Elaziz, M. A. (2024). Modified Aquila Optimizer Feature Selection Approach and Support Vector Machine Classifier for Intrusion Detection System. *Multimedia Tools and Applications*, 1-27. <https://doi.org/10.1007/s11042-023-17886-2>
  41. Jayalatchumy, D., Ramalingam, R., Balakrishnan, A., Safran, M., & Alfarhood, S. (2024). Improved Crow Search-based Feature Selection and Ensemble Learning for IoT Intrusion Detection. *IEEE Access*. <https://doi.org/10.1109/ACCESS.2024.3372859>
  42. Jiang, T., Fu, X., & Wang, M. (2024). BBO-CFAT: Network intrusion detection model based on BBO algorithm and hierarchical transformer. *IEEE Access*. <https://doi.org/10.1109/ACCESS.2024.3386405>
  43. Sangaiah, A. K., Javadpour, A., Ja'fari, F., Pinto, P., Zhang, W., & Balasubramanian, S. (2023). A hybrid heuristics artificial intelligence feature selection for intrusion detection classifiers in cloud of things. *Cluster Computing*, 26(1), 599-612. <https://doi.org/10.1007/s10586-022-03629-9>
  44. Ye, Z., Luo, J., Zhou, W., Wang, M., & He, Q. (2023). An ensemble framework with improved hybrid breeding optimization-based feature selection for intrusion detection. *Future Generation Computer Systems*. <https://doi.org/10.1016/j.future.2023.09.035>
  45. Subramani, S., & Selvi, M. (2023). Multi-objective PSO based feature selection for intrusion detection in IoT based wireless sensor networks. *Optik*, 273, 170419. <https://doi.org/10.1016/j.jlleo.2022.170419>
  46. Alghanam, O. A., Almobaideen, W., Saadeh, M., & Adwan, O. (2023). An improved PIO feature selection algorithm for IoT network intrusion detection system based on ensemble learning. *Expert Systems with Applications*, 213, 118745. <https://doi.org/10.1016/j.eswa.2022.118745>
  47. Barhoush, M., Abed-alguni, B. H., & Al-qudah, N. E. A. (2023). Improved discrete salp swarm algorithm using exploration and exploitation techniques for feature selection in intrusion detection systems. *The Journal of Supercomputing*, 79(18), 21265-21309. <https://doi.org/10.1007/s11227-023-05444-4>
  - problems. *Swarm and Evolutionary Computation*, 64, 100888. <https://doi.org/10.1016/j.swevo.2021.100888>
  24. Dehghani, M., Montazeri, Z., Trojovská, E., & Trojovský, P. (2023). Coati Optimization Algorithm: A new bio-inspired metaheuristic algorithm for solving optimization problems. *Knowledge-Based Systems*, 259, 110011. <https://doi.org/10.1016/j.knsys.2022.110011>
  25. Priyadarshini, J., Premalatha, M., Čep, R., Jayasudha, M., & Kalita, K. (2023). Analyzing physics-inspired metaheuristic algorithms in feature selection with K-nearest-neighbor. *Applied Sciences*, 13(2), 906. <https://doi.org/10.3390/app13020906>
  26. Jia, H., & Lu, C. (2024). Guided learning strategy: A novel update mechanism for metaheuristic algorithms design and improvement. *Knowledge-Based Systems*, 286, 111402. <https://doi.org/10.1016/j.knsys.2024.111402>
  27. Kwakye, B. D., Li, Y., Mohamed, H. H., Baidoo, E., & Asenso, T. Q. (2024). Particle guided metaheuristic algorithm for global optimization and feature selection problems. *Expert Systems with Applications*, 248, 123362. <https://doi.org/10.1016/j.eswa.2024.123362>
  28. Di Mauro, M., Galatro, G., Fortino, G., & Liotta, A. (2021). Supervised feature selection techniques in network intrusion detection: A critical review. *Engineering Applications of Artificial Intelligence*, 101, 104216. <https://doi.org/10.1016/j.engappai.2021.104216>
  29. Nazir, A., & Khan, R. A. (2021). A novel combinatorial optimization based feature selection method for network intrusion detection. *Computers & Security*, 102, 102164. <https://doi.org/10.1016/j.cose.2020.102164>
  30. Slowik, A., & Kwasnicka, H. (2020). Evolutionary algorithms and their applications to engineering problems. *Neural Computing and Applications*, 32, 12363-12379. <https://doi.org/10.1007/s00521-020-04832-8>
  31. Beni, G. (2020). Swarm intelligence. *Complex Social and Behavioral Systems: Game Theory and Agent-Based Models*, 791-818. [https://doi.org/10.1007/978-1-0716-0368-0\\_530](https://doi.org/10.1007/978-1-0716-0368-0_530)
  32. Prajapati, V. K., Jain, M., & Chouhan, L. (2020, February). Tabu search algorithm (TSA): A comprehensive survey. In *2020 3rd International Conference on Emerging Technologies in Computer Engineering: Machine Learning and Internet of Things (ICETCE)* (pp. 1-8). *IEEE*. <https://doi.org/10.1109/ICETCE48199.2020.9091743>
  33. Selvakumar, B., & Muneeswaran, K. (2019). Firefly algorithm based feature selection for network intrusion detection. *Computers & Security*, 81, 71-155. <https://doi.org/10.1016/j.cose.2018.11.005>
  34. Zhou, Y., Cheng, G., Jiang, S., & Dai, M. (2020). Building an efficient intrusion detection system based on feature selection and ensemble classifier. *Computer networks*, 174, 107247. <https://doi.org/10.1016/j.comnet.2020.107247>
  35. Mohammadi, S., Mirvaziri, H., Ghazizadeh-Ahsae, M., & Karimipour, H. (2019). Cyber intrusion detection by combined feature selection algorithm. *Journal of information security and applications*, 44, 80-88. <https://doi.org/10.1016/j.jisa.2018.11.007>
  36. Nazir, A., & Khan, R. A. (2021). A novel combinatorial optimization based feature selection

- using hybrid Naïve Bayes—Decision table and multi-objective evolutionary feature selection. *Computer Communications*, 188, 133-144. <https://doi.org/10.1016/j.comcom.2022.03.009>.
58. Ghanem, W. A. H., Ghaleb, S. A. A., Jantan, A., Nasser, A. B., Saleh, S. A. M., Ngah, A., ... & Abiodun, O. I. (2022). Cyber intrusion detection system based on a multiobjective binary bat algorithm for feature selection and enhanced bat algorithm for parameter optimization in neural networks. *IEEE Access*, 10, 76318-76339. <https://doi.org/10.1109/ACCESS.2022.3192472>.
  59. Aksu, D., & Aydin, M. A. (2022). MGA-IDS: Optimal feature subset selection for anomaly detection framework on in-vehicle networks-CAN bus based on genetic algorithm and intrusion detection approach. *Computers & Security*, 118, 102717. <https://doi.org/10.1016/j.cose.2022.102717>
  60. Zhao, R., Mu, Y., Zou, L., & Wen, X. (2022). A hybrid intrusion detection system based on feature selection and weighted stacking classifier. *IEEE Access*, 10, 71414-71426. <https://doi.org/10.1109/ACCESS.2022.3186975>.
  61. Chen, J., Qi, X., Chen, L., Chen, F., & Cheng, G. (2020). Quantum-inspired ant lion optimized hybrid k-means for cluster analysis and intrusion detection. *Knowledge-Based Systems*, 203, 106167. <https://doi.org/10.1016/j.knosys.2020.106167>.
  62. Kunhare, N., Tiwari, R., & Dhar, J. (2022). Intrusion detection system using hybrid classifiers with meta-heuristic algorithms for the optimization and feature selection by genetic algorithm. *Computers and Electrical Engineering*, 103, 108383. <https://doi.org/10.1016/j.compeleceng.2022.108383>.
  63. Khanna, A., Rani, P., Garg, P., Singh, P. K., & Khamparia, A. (2022). An enhanced crow search inspired feature selection technique for intrusion detection based wireless network system. *Wireless Personal Communications*, 127(3), 2021-2038. <https://doi.org/10.1007/s11277-021-08766-9>.
  64. Halim, Z., Yousaf, M. N., Waqas, M., Sulaiman, M., Abbas, G., Hussain, M., ... & Hanif, M. (2021). An effective genetic algorithm-based feature selection method for intrusion detection systems. *Computers & Security*, 110, 102448. <https://doi.org/10.1016/j.cose.2021.102448>.
  65. Almasoudy, F. H., Al-Yaseen, W. L., & Idrees, A. K. (2020). Differential evolution wrapper feature selection for intrusion detection system. *Procedia Computer Science*, 167, 1230-1239. <https://doi.org/10.1016/j.procs.2020.03.438>.
  66. Khammassi, C., & Krichen, S. (2020). A NSGA2-LR wrapper approach for feature selection in network intrusion detection. *Computer Networks*, 172, 107183. <https://doi.org/10.1016/j.comnet.2020.107183>
  67. Elmasry, W., Akbulut, A., & Zaim, A. H. (2020). Evolving deep learning architectures for network intrusion detection using a double PSO metaheuristic. *Computer Networks*, 168, 107042. <https://doi.org/10.1016/j.comnet.2019.107042>
  68. Ogundokun, R. O., Awotunde, J. B., Sadiku, P., Adeniyi, E. A., Abiodun, M., & Dauda, O. I. (2021). 48. Faris, M., Mahmud, M. N., Salleh, M. F. M., & Alsharaa, B. (2023). A differential evolution-based algorithm with maturity extension for feature selection in intrusion detection system. *Alexandria Engineering Journal*, 81, 178-192. <https://doi.org/10.1016/j.aej.2023.09.032>
  49. Maheswari, K. G., Siva, C., & Nalinipriya, G. (2023). Optimal cluster based feature selection for intrusion detection system in web and cloud computing environment using hybrid teacher learning optimization enables deep recurrent neural network. *Computer Communications*, 202, 145-153. <https://doi.org/10.1016/j.comcom.2023.02.003>
  50. Rabash, A. J., Nazri, M. Z. A., Shapui, A., & Hasan, M. K. (2023). Non-Dominated Sorting Genetic Algorithm based Dynamic Feature Selection for Intrusion Detection System. *IEEE Access*. <https://doi.org/10.1109/ACCESS.2023.3328395>
  51. Mohy-Eddine, M., Guezzaz, A., Benkirane, S., & Azrou, M. (2023). An efficient network intrusion detection model for IoT security using K-NN classifier and feature selection. *Multimedia Tools and Applications*, 82(15), 23615-23633. <https://doi.org/10.1007/s11042-023-14795-2>
  52. Quincozes, S. E., Passos, D., Albuquerque, C., Mossé, D., & Ochi, L. S. (2022). An extended assessment of metaheuristics-based feature selection for intrusion detection in CPS perception layer. *Annals of Telecommunications*, 77(7), 457-471. <https://doi.org/10.1007/s12243-022-00912-z>.
  53. Saheed, Y. K. (2022). A binary firefly algorithm-based feature selection method on high dimensional intrusion detection data. In *Illumination of Artificial Intelligence in Cybersecurity and Forensics* (pp. 273-288). Cham: Springer International Publishing. [https://doi.org/10.1007/978-3-030-93453-8\\_12](https://doi.org/10.1007/978-3-030-93453-8_12).
  54. Balogun, B. F., Gbolagade, K. A., Arowolo, M. O., & Saheed, Y. K. (2021). A hybrid metaheuristic algorithm for features dimensionality reduction in network intrusion detection system. In *Computational Science and Its Applications—ICCSA 2021: 21st International Conference, Cagliari, Italy, September 13–16, 2021, Proceedings, Part IX 21* (pp. 101-114). Springer International Publishing. [https://doi.org/10.1007/978-3-030-87013-3\\_8](https://doi.org/10.1007/978-3-030-87013-3_8).
  55. Sharma, M., Saini, S., Bahl, S., Goyal, R., & Deswal, S. (2021). Modified bio-inspired algorithms for intrusion detection system. In *International Conference on Innovative Computing and Communications: Proceedings of ICICC 2020, Volume 1* (pp. 185-201). Springer Singapore. [https://doi.org/10.1007/978-981-15-5113-0\\_14](https://doi.org/10.1007/978-981-15-5113-0_14).
  56. Stankovic, M., Antonijevic, M., Bacanin, N., Zivkovic, M., Tanaskovic, M., & Jovanovic, D. (2022, October). Feature selection by hybrid artificial bee colony algorithm for intrusion detection. In *2022 International Conference on Edge Computing and Applications (ICECAA)* (pp. 500-505). IEEE. <https://doi.org/10.1109/ICECAA55415.2022.9936116>
  57. Panigrahi, R., Borah, S., Pramanik, M., Bhoi, A. K., Barsocchi, P., Nayak, S. R., & Alnumay, W. (2022). Intrusion detection in cyber-physical environment

- An enhanced intrusion detection system using particle swarm optimization feature extraction technique. *Procedia Computer Science*, 193, 504-512. <https://doi.org/10.1016/j.procs.2021.10.052>
69. Velliangiri, S., & Karthikeyan, P. (2020). Hybrid optimization scheme for intrusion detection using considerable feature selection. *Neural Computing and Applications*, 32(12), 7925-7939. <https://doi.org/10.1007/s00521-019-04477-2>
  70. Alazzam, H., Sharieh, A., & Sabri, K. E. (2020). A feature selection algorithm for intrusion detection system based on pigeon inspired optimizer. *Expert systems with applications*, 148, 113249. <https://doi.org/10.1016/j.eswa.2020.113249>
  71. Maheswari, S., & Arunesh, K. (2020, September). Unsupervised Binary BAT algorithm based Network Intrusion Detection System using enhanced multiple classifiers. In *2020 International Conference on Smart Electronics and Communication (ICOSEC)* (pp. 885-889). IEEE. <https://doi.org/10.1109/ICOSEC49089.2020.9215453>
  72. Vijayanand, R., & Devaraj, D. (2020). A novel feature selection method using whale optimization algorithm and genetic operators for intrusion detection system in wireless mesh network. *IEEE Access*, 8, 56847-56854. <https://doi.org/10.1109/ACCESS.2020.2978035>
  73. Ravindranath, V., Ramasamy, S., Somula, R., Sahoo, K. S., & Gandomi, A. H. (2020, July). Swarm intelligence based feature selection for intrusion and detection system in cloud infrastructure. In *2020 IEEE congress on evolutionary computation (CEC)* (pp. 1-6). IEEE. <https://doi.org/10.1109/CEC48606.2020.9185887>
  74. Al Ogaili, R. R. N., Alomari, E. S., Alkorani, M. B. M., Alyasseri, Z. A. A., Mohammed, M. A., Dhanaraj, R. K., ... & Karuppayah, S. (2023). Malware cyberattacks detection using a novel feature selection method based on a modified whale optimization algorithm. *Wireless Networks*, 1-17. <https://doi.org/10.1007/s11276-023-03606-z>
  75. Alzubi, Q. M., Anbar, M., Alqattan, Z. N., Al-Betar, M. A., & Abdullah, R. (2020). Intrusion detection system based on a modified binary grey wolf optimisation. *Neural computing and applications*, 32, 6125-6137. <https://doi.org/10.1007/s00521-019-04103-1>.
  76. Alamiyedy, T. A., Anbar, M., Alqattan, Z. N., & Alzubi, Q. M. (2020). Anomaly-based intrusion detection system using multi-objective grey wolf optimisation algorithm. *Journal of Ambient Intelligence and Humanized Computing*, 11(9), 3735-3756. <https://doi.org/10.1007/s12652-019-01569-8>.