مجله کنترل
ISSN (print) 2008-8345
ISSN (online) 2538-3752
مقاله پژوهشی
جلد۱۷، شماره ۲، تابستان۱۴۰۲ صفحه ۱۲۷-۱۱۳

مروری بر روش‌های تحلیل هجوم هشدار در فرآیندهای صنعتی: شناسایی الگو و بررسی شباهت

محمد حسین روحی[۱]، ایمان ایزدی[۲]

[۱] دستیار تحقیقاتی، دانشکدهٔ مهندسی برق و کامپیوتر، دانشگاه صنعتی اصفهان، ااصفهان، ایران roohi@ualberta.ca

[۲] دانشیار، دانشکدهٔ مهندسی برق و کامپیوتر، دانشگاه صنعتی اصفهان،اصفهان، ایران iman.izadi@iut.ac.ir

دریافت: ۱۴۰۲/۰۵/۰۵ ویرایش: ۱۴۰۲/۰۶/۰۲ پذیرش: ۱۴۰۲/۰۶/۲۰

چکیده: در این مقاله، مرور روش‌های مختلف شناسایی الگو و تجزیه و تحلیل شباهت در سیستم‌های هشدار و تاثیر این روش‌ها در بهبود کارایی این سیستم‌ها، با تمرکز بر هجوم هشدار ارائه خواهد شد. با تجزیه و تحلیل انواع مختلف هشدارها و الگوهای آنها، امکان پیش‌بینی‌های دقیق‌تر، واکنش به موقع و بهبود نگهداری سیستم فراهم می‌شود. همچنین در این مقاله به پیشرفت‌های اخیر فناوری که این روش‌ها را تقویت کرده‌اند اشاره می‌شود. به طور خاص نقش پیشرفت‌های اخیر در حوزه هوش مصنوعی و یادگیری ماشین در بهبود روش‌های شناسایی الگو و تجزیه و تحلیل شباهت به ویژه در زمان هجوم هشدار بررسی خواهد شد. استفاده از این فناوری‌های نوین در سیستم‌های مدیریت هشدار سنتی راه را برای بهبود قابل توجه در ایمنی صنعتی هموار می‌کند. در این راستا نمونه‌هایی از صنایع مختلف برای مشخص شدن کاربرد عملی و تأثیر این روش‌ها عنوان خواهد شد.

کلمات کلیدی: سیستم‌های مدیریت هشدار، هجوم هشدار، شناسایی الگو، تجزیه و تحلیل شباهت، ایمنی صنعتی.

A review of alarm flood analysis methods in industrial processes: pattern recognition and similarity analysis

Mohammad Hossein Roohi, Iman Izadi

Abstract: In this paper, recently proposed methods of pattern recognition and similarity analysis in alarm management systems are reviewed, and the impact of these methods on improving the efficiency of these systems, with emphasis on alarm floods, is discussed. Various types of alarms and their patterns are analyzed, highlighting the potential for more precise predictions, timely response strategies, and improvements in system maintenance. The technological advancements that have strengthened these methods are explained. More specifically, the role of recent advancements in artificial intelligence and machine learning in enhancing pattern recognition and similarity analysis, especially during alarm floods, is studied. The combination of these modern technologies with traditional alarm management systems paves the way for significant improvements in industrial safety. Examples from various industries are provided to demonstrate the practical application and impact of these methods.

Keywords: Alarm management systems, Alarm flood, Pattern recognition, Similarity analysis, Industrial safety.

۱- مقدمه

یک سیستم مدیریت هشدار[۱]، جزئی حیاتی در صنایع مختلف است که برای اطلاع‌رسانی در مورد وقوع رویدادها یا وضعیت‌های غیرطبیعی طراحی شده است [۱، ۲، ۳]. طبق استاندارد صنعتی ISA-18.2 سیستم هشدار به این صورت تعریف می‌شود: «یک سیستم هشدار، مجموعه‌ای از سخت‌افزار و نرم‌افزار است که حالت هشدار را تشخیص می‌دهد، نشانه آن حالت را به اپراتورها ارائه می‌دهد و تغییرات در حالت هشدار را ثبت می‌کند» [۴]. یک سیستم هشدار مناسب می‌تواند در افزایش ایمنی کارکنان، حفاظت از تجهیزات، کاهش خسارت‌ها و بهبود بازده فرآیندهای صنعتی نقش مهمی ایفا کند. این سیستم‌ها معمولاً اطلاعات دریافت شده از حسگرها و متغیرهای سیستم کنترل را تجزیه و تحلیل کرده و زمانی که یک وضعیت غیرطبیعی را تشخیص دهند، اعلان‌های هشدار مربوطه را تولید می‌کنند.

هدف اصلی سیستم هشدار اطلاع‌رسانی صحیح و اثربخش در مورد وقوع رویدادها و موقعیت‌های غیرطبیعی در حداقل زمان ممکن است. این رویدادها می‌توانند شامل عملکرد نادرست، خطاها، خطرات ایمنی، انحراف متغیرهای فرآیندی یا سیستم کنترل و به طور کلی هر نوع مشکل یا خطر مربوط به فرآیند مورد نظر باشند. با استفاده از سیستم هشدار، اپراتورها می‌توانند واکنش مناسب را نسبت به وقوع رویدادها نشان داده و اقدامات لازم را برای کنترل و رفع به‌موقع مشکلات انجام دهند. سیستم‌های مدیریت هشدار (به ویژه در شرایط بحرانی که اهمیت این سیستم‌ها دو چندان می‌شود)، ممکن است با مشکلاتی مواجه شوند که می‌تواند به سردرگمی اپراتور و عدم انجام به‌موقع اقدامات مورد نیاز منجر شود. برخی از این مشکلات متداول شامل موارد زیر است:

۱. هشدارهای ناصحیح[۲]: هشدارهایی در مورد رویدادها یا وضعیت‌هایی هستند که در واقعیت ناظر به حالت غیرطبیعی فرآیند نیستند. این نوع هشدارها ممکن است به دلایل فنی، خطای‌تشخیص یا تنظیمات اشتباه سیستم مدیریت هشدار رخ دهند.

۲. هشدارهای مزاحم[۳]: مربوط به هشدارها در مورد رویدادهای غیرمهم یا خطاهای موقتی است که تأثیر مهمی بر عملکرد فرآیند ندارند. این هشدارها ممکن است تمرکز اپراتورها را از هشدارهای واقعی و مهم منحرف کند.

۳. هجوم هشدار[۴]: وقوع مجموعه‌ای از هشدارها در یک زمان کوتاه و بیش از مقدار قابل قبول است و می‌تواند اپراتور را در مورد تشخصی وضعیت واقعی و یافتن راه حل مناسب و سریع برای آن سردرگم کند.

در ادامه ابتدا پیرامون دو مورد اول توضیحات بیشتری ارائه شده و سپس بر هجوم هشدار به عنوان موضوع محوری این مقاله تمرکز می‌شود.

یک هشدار نادرست زمانی رخ می‌دهد که یک هشدار بدون دلیل مشخص یا بر اثر خطای سیستم اعلام می‌شود. این مورد می‌تواند باعث کاهش اعتماد اپراتورها نسبت به سیستم مدیریت هشدار شود. دو معیار کلیدی در بررسی اثربخشی سیستم‌های هشدار نرخ هشدار نادرست[۵] و نرخ هشدار از دست رفته[۶] هستند [۵]. نرخ هشدار نادرست تعداد هشدارهای نادرست را نسبت به کل تعداد هشدارها محاسبه می‌کند. نرخ بالای هشدار نادرست می‌تواند باعث شود که اپراتورها هشدارهای مهم واقعی را نادیده بگیرند. از سوی دیگر نرخ هشدار از دست رفته تعداد وقایعی که باید اعلام می‌شدند اما نشده‌اند را محاسبه می‌کند. از دست رفتن یک هشدار می‌تواند در بهترین حالت باعث افت کارایی فرآیند و در بدترین حالت می‌تواند منجر به خرابی سیستم و حتی فاجعه انسانی شود. برخی از روش‌های اصلی برای مقابله با هشدارهای نادرست شامل این موارد است:

۱. منطقی‌سازی هشدارها[۷]: به معنای بازبینی منظم هشدارهای موجود است تا هشدارهای غیرضروری حذف شده و اطمینان حاصل شود که هشدارهای باقی‌مانده به درستی تنظیم و بر اساس اهمیتشان اولویت‌بندی شده‌اند [۶].

۲. فیلتر کردن هشدارها: هشدارهای تولید شده بر اساس مشخصه‌های مختلفی مانند اولویت، نوع فرآیند، سطح خطر، زمان و تغییرات وضعیت دسته‌بندی و ارزیابی می‌شوند. سپس، با استفاده از قوانین و الگوریتم‌هایی هشدارهایی که با معیارها و شرایط تعیین شده سازگاری ندارند، تصفیه و حذف می‌شوند.

۳. تجزیه و تحلیل پیشرفته داده‌ها: با استفاده از ابزارهای تجزیه و تحلیل داده‌ها می‌توان درک بهتری از دلایل وقوع هشدارهای نادرست به دست آورد و راهکارهایی برای کاهش آن‌ها ارائه داد. بررسی شباهت میان هجوم‌های هشدار، استخراج الگو در سیل‌های هشدار و کشف هشدارهایی که باعث وقوع دیگر هشدارها شده‌اند در این دسته‌بندی قرار می‌گیرند [۷]. تمرکز این مقاله بر روش‌های مبتنی بر تجزیه و تحلیل شباهت و تطابق الگوها در هجوم‌های هشدار است. در ادامه توضیح مختصری پیرامون این دو روش داده می‌شود. روش‌های مبتنی بر تحلیل شباهت بر اساس مقایسه الگوهای هشدار با یکدیگر و یافتن ویژگی‌های مشترک و شباهت‌های آن‌ها است. در این روش‌ها، الگوهای هشدار مختلف با استفاده از رویکردهایی مانند مقایسه الگوها بر اساس معیارهای فضایی یا زمانی مورد بررسی قرار می‌گیرند. با مشخص کردن شباهت میان الگوها می‌توان ویژگی‌های مشترک و تفاوت‌های بین الگوها را شناسایی کرد و از آن‌ها برای تشخیص الگوهای ناخواسته و پر خطر استفاده کرد.

---

[۱] Alarm management system
[۲] False alarms
[۳] Nuisance alarms
[۴] Alarm flood
[۵] False Rate Alarm (FAR)
[۶] Missed Alarm Rate (MAR)
[۷] Alarm rationalization

تطبیق الگو به معنای جستجو و تشخیص الگوهای مشخص در داده‌های هشدار است. در این روش‌ها، الگوهای مشخص و از پیش‌تعیین شده به عنوان الگوهای معتبر هشدار در نظر گرفته می‌شوند. سپس، با استفاده از الگوریتم‌های تطبیق الگو، داده‌های هشدار مورد بررسی قرار گرفته و با الگوهای مشخص شده مقایسه می‌شوند. اگر هشدارها با این الگوها مطابقت داشته باشند، به عنوان هشدار معتبر شناخته می‌شوند. بنابراین، تحلیل شباهت بیشتر بر روی مقایسه الگوها و شباهت آن‌ها تمرکز دارد، در حالی که تطبیق الگو بر روی تطابق هشدارهای دریافتی با الگوهای مشخص و پیش‌تعیین شده تمرکز دارد.

در ادامه این مقاله به بررسی مفاهیم اساسی و مبانی نظری این حوزه پرداخته شده و انواع مختلف روش‌های تشابه‌سنجی و تطابق الگو مورد بررسی قرار می‌گیرد. همچنین، نمونه‌های کاربردی از این روش‌ها در زمینه‌های مختلف ارائه و نقاط قوت و ضعف هر روش بیان می‌شود.

۲ – اهمیت مساله

گزارش یک مطالعه جامع شامل صنایع متنوعی مانند نفت و گاز، پتروشیمی و برق توسط [۸] نشان می‌دهد که اکثر سیستم‌های هشدار صنعتی در عمل با عدم کارایی مناسب روبرو هستند. برای تبیین بیشتر، معیارهای کلی‌ای از کارایی سیستم‌های هشدار در جدول ۱ خلاصه شده‌اند. این جدول به صراحت نشان می‌دهد که بین سیستم‌های صنعتی مورد بررسی و معیاری که توسط استاندارد صنعتی (EEMUA-191 [۹]) ارائه شده، تفاوت معناداری مشاهده می‌شود.

جدول ۱: مرور عمکرد سیستم‌های هشدار [۸]

سنجش عملکرد	EEMUA-191	نفت و گاز	پتروشیمی	برق
میانگین هشدارها در ساعت	۶	۶۳	۴۵	۸۴
میانگین هشدارهای ایستاده	۹	۰۵	۰۰۱	۵۶
بیشترین هشدارها در ساعت	۰۶	۰۲۳۱	۰۸۰۱	۰۰۱۲
درصد توزیع اولویت (پایین/ متوسط / بالا)	۰۸/۵۱/۵	۵۲/۰۴/۵۳	۵۲/۰۴/۵۳	۵۲/۰۴/۵۳

با توجه به اهمیت موضوع هجوم‌های هشدار، در سال‌های اخیر روش‌های مختلفی برای مقابله با این مشکل پیشنهاد شده‌اند. این روش‌ها را می‌توان به صورت کلی به ایده‌های مبتنی بر کاهش نرخ هشدارها برای مهار هجوم‌های هشدار، مقایسه هجوم‌های هشدار برای تسهیل در تحلیل، استخراج الگوهای هجوم هشدار، تحلیل علل اصلی و کمک به اپراتورها برای افزایش آگاهی از وضعیت دسته‌بندی کرد [۱۰]. در ادامه به چند نمونه از پژوهش‌های صورت گرفته در این زمینه اشاره می‌شود.

۱. برای مقابله با پدیده هجوم هشدار، اولین راه حل، کاهش تعداد هشدارهای تولید شده از طریق بهینه‌سازی موازنه‌ی بین نرخ هشدار اشتباه و هشدار از دست رفته است. در این راستا روش‌های مبتنی بر باند مرده[۱] [۱۱، ۱۲]، تایمرهای تاخیر[۲] [۱۳، ۱۴] و فیلترهای هشدار [۱۵، ۱۶، ۱۷] معرفی شده‌اند. به علاوه تاثیر کنترل کننده و طراحی بهینه آن برای بهبود عملکرد سیستم هشدار در [۱۸، ۱۹] مورد بررسی قرار گرفته است.

۲. تحلیل علت اصلی هشدار[۳] می‌تواند با شناسایی و رفع مشکلات اصلی که منجر به بروز هجوم‌های هشدار می‌شوند به مهار هجوم‌های هشدار کمک کند. در عین حال، این تحلیل می‌تواند به اپراتورها کمک کند تا متوجه شوند که کدام هشدارها نیاز به توجه فوری دارند و کدام هشدارها نتیجه‌ی هشدارهای دیگر هستند [۷، ۲۰]. از میان این پژوهش‌ها می‌توان به روش‌های مبتنی بر علیت گرانگر[۴] [۲۱، ۲۲]آنتروپی انتقال[۵] [۲۳، ۲۴] و روش‌های مبتنی بر مدل‌های گرافی احتمالاتی [۲۶، ۲۶] اشاره کرد.

۳. تحلیل تشابه و بررسی الگوهای موجود در هجوم‌های هشدار می‌تواند در مقابله با هجوم هشدار موثر باشد، چرا که این رویکرد به شناسایی و درک علل اصلی هجوم‌های هشدار کمک می‌کند. بررسی تشابه بین هجوم‌های هشدار مختلف می‌تواند الگوهایی که به وقوع هجوم هشدار منجر می‌شوند، آشکار کند. با تشخیص این الگوها، می‌توان رویکردها و راه‌حل‌های موثرتری برای جلوگیری یا کاهش هجوم‌های هشدار ارائه داد. به عبارت دیگر، تحلیل تشابه علت اصلی می‌تواند به سیستم هشدار کمک کند تا بفهمد کدام هشدارها ممکن است همزمان یا به ترتیب به وقوع پیوندند، و بر اساس آن، اقدامات مناسبی را انجام دهد تا هجوم‌های هشدار را کنترل کند.

در ادامه این مقاله ابتدا هجوم هشدار و روش تعیین آن را معرفی کرده و سپس پژوهش‌های انجام شده به منظور کاهش این پدیده و مقابله با آن با استفاده از تشخیص شباهت و کشف الگو مورد بررسی قرار می‌گیرد.

۳ – تشخیص وقوع هجوم هشدار

یک سیستم هشدار با مجموعه متغیرهای هشدار

$$A = \{\alpha_i, i = 1,2,\cdots,|A|\}$$

را در نظر بگیرید که در آن $|.|$ کاردینالیتی مجموعه را نشان می‌دهد. در این مجموعه هر هشدار $\alpha_i \in A$ از یک سیگنال هشدار $x_{a_i}(t)$ ساخته شده و به صورت زیر تولید می‌شود:

$$x_{a_i}(t) = \begin{cases} 1 & \text{اگر } \tilde{x}_{a_i}(t) \in X_{\mathbf{ab}}, \\ 0 & \text{اگر } \tilde{x}_{a_i}(t) \in X_{\mathbf{n}}. \end{cases}$$

[۱] Deadband
[۲] Delay timers
[۳] Alarm root cause
[۴] Granger causality
[۵] Transfer entropy

که در آن X_n و X_{ab} به ترتیب ناظر به محدوده عملکرد طبیعی و غیرطبیعی متغیر فرآیندی $\tilde{x}_{a_i}$ هستند. در بسیاری از موارد هشدارهای تکرار شونده[۱] می‌تواند منجر به شناسایی غلط هجوم هشدار شود بنا براین برای جلوگیری از اعلام اشتباه هجوم هشدار، ابتدا باید هشدارهای تکرار شونده حذف شوند. روش‌های مختلفی برای حذف این دسته از هشدارها شامل فیلترها، تایمرهای تاخیر و باند مرده معرفی شده‌اند [۱۳، ۱۲] که از این میان، تایمرهای تاخیر می‌توانند مستقیماً بر سیگنال‌های هشدار اعمال شوند. با استفاده از یک تایمر تاخیر خاموش[۲] با λ نمونه، هشدار $A \in \alpha_i$ به صورت زیرتعریف می‌شود:

$$\alpha_i(t) = \begin{cases} 1 & \text{اگر } x_{ai}(t) = 1 \text{ و } \forall k \in \{t-\lambda, t-\lambda+1, \dots, t-1\}, \quad x_{a_i}(k) = 0, \\ 0 & \text{در غیر این صورت.} \end{cases}$$

تشخیص وجود هجوم هشدار می‌تواند براساس نرخ هشدار باشد که یک شاخص کلیدی متداول برای ارزیابی عملکرد یک سیستم هشدار است. نرخ هشدار $\zeta(t)$ در زمان t در پنجره زمانی به طول T به صورت زیر تعریف می‌شود:

$$\zeta(t) = \sum_{i=1}^{|A|} \sum_{k=t-T+1}^{t} x_{ai}(k). \qquad (۱)$$

حال براساس نرخ هشدار $\zeta(t)$، می‌توان زمان شروع و پایان هجوم هشدار را با مقایسه $\zeta(t)$ با آستانه‌ی از پیش تعیین شده معین کرد. متغیر ψ که وجود هجوم هشدار را نشان می دهد، به صورت زیر تعریف می‌شود:

$$\psi(t) = \begin{cases} 1 & \text{اگر } \zeta(t) \geq \Gamma_s \text{ و } \psi(t-1) = 0, \\ 0 & \text{اگر } \zeta(t) < \Gamma_e \text{ و } \psi(t-1) = 1, \\ \psi(t-1) & \text{در غیر این صورت.} \end{cases} \qquad (۲)$$

که در آن ۱ و ۰ به ترتیب وجود و عدم وجود هجوم هشدار را نشان می دهد. نمونه اولیه $\psi(0)$ به 0 مقداردهی شده است.

براساس استاندارد ISA-18.2 [۴]، آستانه‌های مرجع برای شناسایی شروع و پایان هجوم هشدار، به ترتیب، ۱۰ و ۵ هشدار در یک دوره ۱۰ دقیقه‌ای به ازای هر اپراتور است. در اینجا، دو آستانه معادله (۲) با $t_s =$ 10 و $t_e = 5$ براساس پنجره‌ی زمانی با اندازه $T = 10$ معادل دقیقه ۶۰۰ ثانیه در (۱) هستند.

زمان شروع و پایان هر هجوم هشدار می‌تواند براساس متغیر ψ مشخص شود. یک هجوم هشدار در زمان t_s آغاز می‌شود اگر

$$\psi(t_s) = 1 \text{ و } \psi(t_s - 1) = 0,$$

که نشان دهنده‌ی رسیدن نرخ هشدار $\zeta(t)$ به آستانه‌ی ۱۰ هشدار در یک دوره‌ی ۱۰ دقیقه‌ای است، و در لحظه‌ی t_e پایان می‌یابد اگر

$$\psi(t_e) = 0 \text{ و } \psi(t_e - 1) = 1,$$

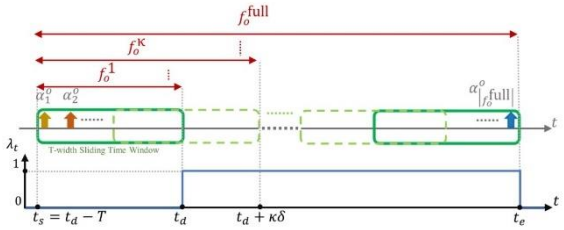
که نشان دهنده‌ی این است که نرخ هشدار $\zeta(t)$ به زیر آستانه‌ی ۵ هشدار در یک دوره ۱۰ دقیقه‌ای بر می گردد. اکنون می‌توان با استفاده از [۲۷] و بر اساس توضیحات فوق هجوم هشدار را برای حالت آفلاین و آنلاین به صورتی که در ادامه توضیح داده می‌شود مشخص کرد.

۳-۱ تشخیص آفلاین

در تشخیص آفلاین هجوم هشدارها، می‌توان از لاگ‌های A&E که داده‌های ضبط شده هشدار را شامل می‌شوند استفاده کرد. با اعمال پنجره‌ی زمانی متحرک بر روی لاگ A&E، هجوم‌های هشدار به عنوان دنباله‌های زمان‌دار در نظر گرفته شده و در یک مجموعه داده هجوم هشدار ذخیره می‌شوند. به این صورت هر دنباله ضبط شده هجوم هشدار به شکل زیر ثبت می‌شود:

$$f_k = \left\{ \left(\alpha_1^k, t_{\alpha_1}^k\right), \left(\alpha_2^k, t_{\alpha_2}^k\right), \dots, \left(\alpha_{|f_k|}^k, t_{\alpha_{|f_k|}}^k\right) \right\}.$$

در اینجا، $f_k \in F$ هجوم هشدار kام در مجموعه داده ضبط شده هجوم هشدار F است، که $\alpha_m^k \in A; k = 1,2,\dots,|F|$ برچسب هشدار mام در f_k است، که $m = 1,2,\dots,|f_k|$؛ $t_{\alpha_m}^k$ مهر زمانی فعال‌شدن مرتبط با α_m^k است که متعلق به بازه زمانی $[t_s^k, t_e^k]$ است؛ و t_s^k و t_e^k به ترتیب زمان‌های شروع و پایان هجوم هشدار f_k هستند.



شکل ۱: تشخیص آنلاین هجوم هشدار [۲۷]

۳-۲ تشخیص آنلاین

شناسایی هجوم هشدار در زمان واقعی و به صورت آنلاین با استفاده از یک پنجره‌ی زمانی با عرض T قابل پیاده سازی است.

این فرایند در شکل ۱ نمایش داده شده و طبق معادله زیر توصیف شده است:

[۱] Chattering alarm

[۲] off-delay-timer

$$
\begin{aligned}
f_o^1 &= \left\{(\alpha_1^o, t_S), (\alpha_2^o, t_{\alpha_2}^o), \dots, \left(\alpha_{|f_o^1|}^o, t_{\alpha_{|f_\delta^1|}}^o\right)\right\},\\
&\vdots\\
f_o^\kappa &= \left\{(\alpha_1^o, t_S), (\alpha_2^o, t_{\alpha_2}^o), \dots, \left(\alpha_{|f_o^1|}^o, t_{\alpha_{|f_\delta^1|}}^o\right), \dots, \left(\alpha_{|f_o^\kappa|}^o, t_{\alpha_{|f_\delta^\kappa|}}^o\right)\right\},\\
&\vdots\\
f_{full}^o &= \left\{(\alpha_1^o, t_S), (\alpha_2^o, t_{\alpha_2}^o), \dots, \left(\alpha_{|f_o^1|}^o, t_{\alpha_{|f_\delta^1|}}^o\right), \dots, \left(\alpha_{|f_o^\kappa|}^o, t_{\alpha_{|f_\delta^\kappa|}}^o\right), \dots, \left(\alpha_{\left|f_o^{full}\right|}^o, t_e\right)\right\}.
\end{aligned}
$$

برای زمان t_d که $\psi(t)$ طبق (۲) از صفر به یک تغییر پیدا می‌کند، هجوم هشدار آنلاین اول f_1^o شناسایی می‌شود. در هر به‌روزرسانی، نمونه‌های قبلی هشدار خارج از بازه زمانی T حذف و هشدارهای جدید اضافه می‌شوند تا نرخ هشدار محاسبه گردد. تا زمانی که $\psi(t)$ برابر با یک باشد، هشدار آنلاین به‌روز می‌شود و هنگامی که از یک به صفر تغییر می‌کند، پایان هجوم هشدار شناسایی می‌شود. به‌روزرسانی‌های متفاوت یک هجوم هشدار آنلاین با استفاده از برچسب‌های زمانی متناظر، ضبط می‌شوند. در اینجا، f_κ^o و f_{full}^o نمایانگر به‌روزرسانی κ مین و آخرین هجوم هشدار هستند. همچنین $t_{\alpha_r}^o$ زمان وقوع هشدار t_s و α_r^o و t_e به ترتیب برچسب زمانی هشدارهای اول و آخر آخرین هجوم هشدار هستند.

٤ - نمودارهای مرتبط با هجوم هشدار

در این بخش، برخی از نمودارهایی که منظره کلی از هجوم‌های هشدار شناسایی شده و الگوهای هشدار را فراهم می‌کند ارائه می‌شود. سه نمودار اصلی برای تصویر سازی هجوم‌های هشدار وجود دارد:

• نمودار شباهت هشدار (ASCM)[1] همبستگی خوشه‌ای هشدارها را نشان می‌دهد.

• نمودار میله‌ای سه بعد: نرخ وقوع هشدارها را در یک نمودار سه بعدی نشان می‌دهد. همچنین شامل یک نمودار برای مقایسه نرخ هشدارها با آستانه هجوم هشدار است.

• نمودار مارپیچی[2] داده‌های توالی زمانی را بر روی یک مارپیچ که هر دور آن متناظر با یک شبانه روز است نشان می‌دهد.

نمودار ASCM در واقع به تصویر کشیدن شباهت‌ها و تفاوت‌های بین هشدارهای مختلف است. یک نمونه از این نمودار در شکل ۲ نشان داده شده‌است. رنگ‌های مختلف در نمودار نشان‌دهنده درجه شباهت هستند، به طوری که رنگ‌های تیره‌تر نمایانگر شباهت بیشتری هستند. همچنین، گروه‌بندی هشدارهای مشابه به برجسته‌سازی الگوها و شباهت‌ها کمک می‌کند. این نمودار امکان تحلیل سریع و دقیق هجوم‌های هشدار را فراهم می‌آورد و به شناسایی و درک بهتر الگوها و روابط میان هشدارها کمک می‌کند، و بنابراین ابزار مفیدی برای تحلیل هجوم هشدار است.

ترسیم نمودار ASCM شامل چهار مرحله زیر است [۲۸]:

۱. پر کردن هر دنباله دودویی منحصر به فرد هشدار با ۱ های اضافی: به منظور در نظر گرفتن تاخیرهای ارتباطی و تاخیرهای زمانی متفاوت بین هشدارهای منحصر به فرد، هر دنباله دودویی می‌تواند با ۱ های اضافی پر شود تا تعداد وقوع هشدارهای موجود را افزایش دهد. به عنوان نمونه، برای هر وقوع هشدار در یک دنباله دودویی، پنج ۱ به هر سمت وقوع واقعی اضافه می‌شود که مجموعاً یازده ۱ به هر وقوع هشدار متناظر می‌شود. به عبارت دیگر، بازه تأثیر هر وقوع هشدار در دامنه زمانی به ۱۱ ثانیه افزایش‌می‌یابد.

۲. محاسبه شاخص شباهت بین دنباله‌های دودویی پرشده متناظر با هر جفت منحصر به فرد از هشدارها: با توجه به خصوصیات دنباله‌های دودویی، شاخص شباهت جکارد[3] یک انتخاب متداول برای محاسبه شباهت دنباله‌های هشدار است و به صورت زیر محاسبه می‌شود:

$$S_{Jaccard}(A,B) = \max_{l\in L} \frac{a(l)}{a(l)+b(l)+c(l)},$$

که در آن $a(l)$ تطابق‌ها $(\alpha_i = 1, \beta_{i+l} = 1)$، $b(l)$ تعداد عدم تطابق‌های نوع اول $(\alpha_i = 1, \beta_{i+l} = 0)$ و $c(l)$ تعداد عدم تطابق‌های نوع دوم $(\alpha_i = 0, \beta_{i+l} = 1)$ هستند، و $\forall i \in [1-l, N]$ برای $l \le 0$ و $\forall i \in [1, N-l]$ برای $l > 0$. همچنین α_i و β_j ها هشدارهای دو دنباله هشدار A و B هستند که طبق مرحله قبل ساخته شده‌اند.

۳. مرتب‌سازی منحصر به فرد هشدارها بر اساس میزان شباهت با دیگر هشدارهای منحصر به فرد: برای تصویرسازی کاربردی‌تر، سطرها و ستون‌ها بر اساس ترتیب خوشه‌بندی سلسله‌مراتبی مرتب می‌شوند.

۴. ترسیم ماتریس شباهت مرتب‌شده: ماتریس همبستگی با سطرها و ستون‌ها مجدد مرتب‌شده، مانند شکل ۲ با استفاده از رنگ‌ها کدگذاری می‌شود. همچنین، نام‌های برچسب‌های هشدار در محور افقی حذف شده‌اند زیرا ترتیب آن‌ها همانند ترتیب در محور عمودی است.

در شکل ۳ یک نمونه از نمودار میله‌ای سه بعدی نشان داده شده است. هر میله نمایانگر تعداد هشدار برای یک برچسب هشدار در یک بازه‌ی زمانی ۱۰ دقیقه‌ای است. منحنی سیاه نرخ هجوم هشدار است و خطهای سبز و قرمز رنگ نشان‌دهنده آستانه‌های مرجع متناظر با ۱ و ۱۰ هشدار در یک بازه زمانی ۱۰ دقیقه‌ای هستند. بنابراین قسمت‌هایی از نمودار که منحنی سیاه از آستانه قرمز رنگ فراتر می‌رود نشان دهنده هجوم هشدار است و به صورت همزمان نمودارهای میله‌ای متناظر با این محدوده‌ها نمایان‌گر برچسب‌های هشدار و میزان تاثیر آنها در هجوم هشدار مربوطه هستند.

[1] Alarm Similarity Color Map
[2] Spiral graph
[3] Jaccard

شکل ۴ یک نمودار مارپیچی نشان داده شده است که رنگ سبز مربوط به نرخ پایین‌تر و رنگ نارنجی مربوط به نرخ بالاتر وقوع هشدار است. همچنین رنگ قرمز نمایان‌گر هجوم هشدار است و با توجه به زمان‌های مشخص شده در نمودار می‌توان زمان وقوع و پایان آن را مشاهده کرد. همچنین این نمودار این قابلیت را دارد که نشان دهد هجوم‌های هشدار بیشتر در چه زمان‌هایی از روز رخ می‌دهند.

۵ - روش‌های مقابله با هجوم هشدار

کشف الگوها و بررسی شباهت در هشدارهای متعدد، به عنوان یکی از روش‌های پیشرفته تحلیل و بررسی داده، در سیستم‌های مدیریت هشدار مورد استفاده قرار می‌گیرند. این فرآیند به تجزیه و تحلیل ساختارهای پر تکرار در داده‌های هشدار به منظور شناسایی الگوهای محدود و غیرتصادفی و یا یافتن تشابه بین هجوم‌های‌هشدار مختلف می‌پردازد. پیش‌پردازش داده‌ها، که شامل آماده‌سازی داده‌های اولیه برای استخراج الگوها است، از مراحل ابتدایی واساسی در فرآیند کاوش الگو و بررسی شباهت داده هشدار محسوب می‌شود. این مرحله می‌تواند حتی تا ۶۰ درصد از کل فرآیند زمان کاوش را شامل شود. مرجع [۳۱] به بررسی مراحل اصلی پیش‌پردازش از جمله مسائل مرتبط با هشدارهای تکرار شونده و داده‌های گم‌شده می‌پردازد. همچنین باید تفاوت بین ساختار و قالب‌های مختلف داده‌های هشدار در نظر گرفته شود تا فرآیند پیش‌پردازش بهینه‌تر و دقیق‌تر انجام شود.

مراحل اصلی پیش‌پردازش داده‌های هشدار به شرح زیر است:

۱. جمع‌آوری و ادغام داده‌ها: در این مرحله، داده‌های مربوط به یک بازه زمانی خاص و یک قسمت خاص از سیستم هشدار مورد نظر جمع‌آوری شده و در یک فایل با قالب مشترک ادغام می‌شوند.

۲. حذف رکوردهای غیرضروری: رکوردهای تکراری، هشدارهایی که به طور مستقیم به پایگاه داده فرستاده شده‌اند و به اپراتور نشان داده نشده‌اند، پیغام‌هایی که هشدار نیستند و رکوردهای با ستون‌های از دست رفته حذف می‌شوند.

۳. یکپارچه‌سازی قالب‌های داده: قالب برخی از ستون‌ها باید به یک قالب استاندارد تبدیل شوند تا توسط برنامه داده‌کاوی خوانده شود.

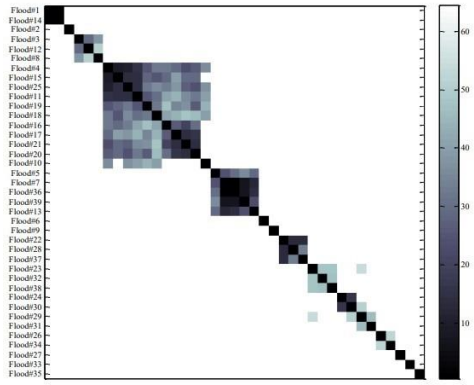
۴. حذف ستون‌های غیرضروری: ستون‌هایی که برای داده‌کاوی لازم نیست حذف می‌شوند.

۵. ایجاد شناسه‌های هشدارهای منحصر به فرد: برای هر هشدار منحصر به فرد یک شناسه ایجاد می‌شود.

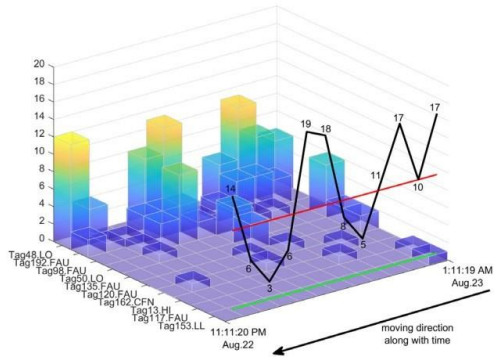
۶. مرتب‌سازی داده‌ها: هشدارها و اعلان‌های برگشت به حالت طبیعی[۱] باید بر اساس برچسب زمانی به ترتیب صعودی مرتب شوند.

۷. حذف هشدارهای قدیمی: هشدارهایی که برای مدت زمان بسیار طولانی فعال بوده‌اند باید حذف شوند.

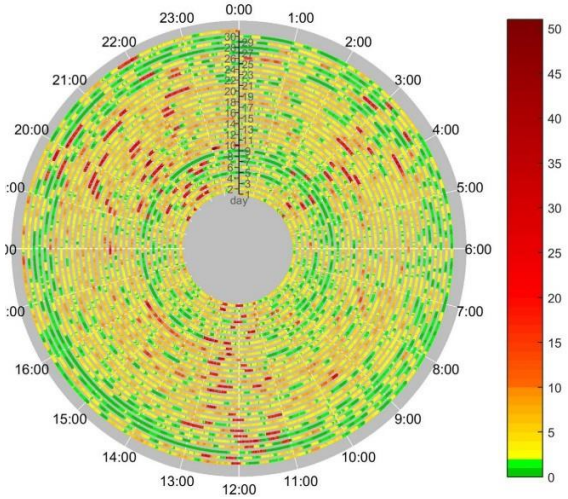
۸. حذف هشدارهای تکرار شونده: هشدارهایی که در یک بازه زمانی کوتاه بارها و بارها فعال و غیرفعال‌می‌شوند، باید تغییر پیدا کرده و به تعداد کمتری هشدار تبدیل شوند.



شکل ۲: نمودار ASCM [۲۹]



شکل ۳: نمودار میله‌ای سه بعدی [۳۰]



شکل ۴: نمودار مارپیچی [۳۰]

[۱] RTNs

٦ - بررسی شباهت در هجوم‌های هشدار

هر هشدار در هجوم‌های هشدار توسط برچسب منحصر به فرد و مهر زمان توصیف می‌شود. یک جفت از دنباله‌های هجوم هشدار می‌تواند صورت $A = [\alpha_1, \alpha_2, \ldots, \alpha_M]$ و $B = [\beta_1, \beta_2, \ldots, \beta_N]$ نمایش داده شود، که در آن‌ها M و N تعداد هشدارهای ظاهر شده در دنباله‌های A و B را نشان می‌دهند. هر عنصر در A یا B می‌تواند با برچسب هشدار منحصر به فرد، مهر زمان و اولویت آن نمایش داده شود. برای مثال، α_i نشان‌دهنده-i امین هشدار از هجوم هشدار A است، و می‌تواند با ویژگی‌های خود به صورت $\alpha_i = [e_i, t_i]$ نمایش داده شود که در آن به ترتیب، e_i و t_i برچسب هشدار و مهر زمان α_i هستند.

در مواجهه با یک هجوم هشدار، یک مسئله این است که چگونه آن را با دنباله‌های ضبط شده مقایسه کرده و مشابه‌ترین را پیدا کنیم تا بتوانیم هشدارها در هجوم هشدار را پیش‌بینی کرده و همچنین در تحلیل علت اصلی از آن استفاده کنیم. همانطور که پیش‌تر در تشخیص هجوم هشدار توضیح داده شد برخلاف تحلیل آفلاین، در تحلیل آنلاین یک دنباله هجوم هشدار می‌تواند با زمان افزایش یابد. بنابراین، تحلیل شباهت آنلاین نوعی مقایسه است که به صورت تکراری با افزایش هشدارها به صورت مداوم انجام می‌شود.

در زمینه مدیریت هشدار، تحلیل شباهت بین دنباله‌های هجوم هشدار با یافتن بخش‌های مشابه یا تطبیق برچسب‌های هشدار مشترک از هر دو دنباله صورت می‌پذیرد. مقاله [۳۲] از فاصله‌ی اقلیدسی بین ماتریس‌های فرکانس هجوم‌های هشدار برای مقایسه تشابه آن‌ها استفاده می‌کند. ماتریس‌های فرکانس $N \times N$ هجوم‌های هشدار، که N تعداد کل هشدارهای ممکن سیستم است، نشان می‌دهد که چه تعداد از هشدار A مستقیماً پس از هشدار B در یک هجوم هشدار رخ می‌دهد. همچنین، از فاصله جکارد برای اندازه‌گیری تشابه بیشتر بین هجوم‌ها، خوشه‌بندی سلسله‌مراتبی و تغییر زمان پویا استفاده شده است. فاصله جکارد یک معیار آماری برای مقایسه تشابه و تفاوت بین دو مجموعه است. این شاخص از نسبت تعداد اعضای مشترک دو مجموعه به تعداد اعضای منحصر به فرد در هر دو مجموعه حاصل می‌شود.

مرجع [۳۳] یک روش برای خوشه‌بندی هجوم هشدار بر اساس همزمانی هشدارها ارائه می‌دهد. ایده اصلی در این روش تبدیل توالی هشدار به نوعی نقشه هم‌زمانی هشدارها است تا به عنوان ویژگی برای خوشه‌بندی استفاده شود. سپس برای خوشه‌بندی نقشه‌های همزمانی معرفی شده از روش‌های ماشین بردار پشتیبان[۱] و k همسایه‌ی نزدیک‌تر[۲] استفاده می‌شود.

مقاله [۳۴] یک استراتژی سلسله‌مراتبی برای طبقه‌بندی و پیش‌بینی هجوم هشدار بر اساس ویژگی‌های ساختاری آنها ارائه می‌دهد. ساختار هجوم هشدار به عنوان مجموعه‌ای از تمام وابستگی‌های مبتنی بر علیت بین متغیرهای هشدار در نظر گرفته می‌شود. در این مقاله ابتدا، یک استخراج کننده ویژگی ساختار[۳] ارائه شده است تا وابستگی‌های اندک هشدار را بررسی کند و هجوم هشدار را به ویژگی‌های ساختاری در فضای برداری تبدیل کند. سپس، یک استراتژی سلسله‌مراتبی با استفاده از ویژگی‌های حاصل شده توسعه داده شده است.

روش‌های متداول تطبیق دنباله، مانند الگوریتم Smith-Waterman و الگوریتم Needleman-Wunsch، تطبیق‌های محلی و عمومی بهینه را همراه با امتیازات شباهت بین دنباله‌های هجوم هشدار ارائه می‌دهند. امتیاز شباهت بالا نشان‌دهنده حضور تعداد زیادی از برچسب‌های هشدار مشترک در هر دو دنباله هجوم هشدار در همان ترتیب زمانی است. در عمل، بازگشت یک خطا معمولا منجر به وقوع هجوم هشدارهای مشابه می‌شود که شامل گروه تقریبا یکسانی از هشدارها در ترتیب توالی خاصی است.

۶-۱ الگوریتم Smith-Waterman

الگوریتم Smith-Waterman ابتدا در سال ۱۸۹۱ توسط Smith و Waterman معرفی شده است [۳۵]. هدف این الگوریتم «یافتن یک جفت بخش از دو دنباله طولانی است که هیچ جفت دیگری از بخش‌ها شباهت بیشتری ندارند». این الگوریتم ابتدا در [۳۶] برای بررسی شباهت زیر دنباله‌ها در هجوم‌های هشدار مورد استفاده قرار گرفت. برای آشنایی کلی با این روش فرض کنیم دو توالی زیر را داریم:

$$A = [3,2,8,1,9]$$
$$B = [3,1,9,2]$$

یک تطبیق محلی برای این دو توالی می‌تواند به صورت زیر باشد:

$$A = [3,2,8,1,9,-,-]$$
$$B = [3,-,-,1,9,2,-]$$

در این دو توالی فاصله‌ها[۴] در مکان‌های مختلف قرار گرفته‌اند تا تطابق بهتری ایجاد شود. بنابراین باید یک روش امتیازدهی معرفی شود که تعداد اضافه کردن فاصله‌ها را (به وسیله اختصاص دادن امتیاز منفی) محدود کند تا تطابق بهینه مشخص شود.

حال برای بررسی الگوریتم Smith-Waterman دو دنباله با طول‌های دلخواه را به صورت زیر در نظر بگیرید:

$$A = [\alpha_1, \alpha_2, \ldots, \alpha_M]$$
$$B = [\beta_1, \beta_2, \ldots, \beta_N]$$

که α_m و β_n نمونه‌هایی از هشدارهای تعریف شده در سیستم هشدار هستند و نمادهای $A_{i:p}$ و $B_{j:q}$ زیردنباله‌هایی از دنباله‌های A و B را نشان می‌دهند:

$$A_{i:p} = [\alpha_i, \alpha_{i+1}, \ldots, \alpha_p]$$
$$B_{j:q} = [\beta_j, \beta_{j+1}, \ldots, \beta_q]$$

که در آن $1 \le i \le p \le M$ و $1 \le j \le q \le N$. نماد $I(A_{i:p}, B_{j:q})$ شاخص شباهت جفت بخش $(A_{i:p}, B_{j:q})$ را نشان می‌دهد. هدف اصلی ما جستجو برای جفت بخش بهینه با بالاترین شاخص شباهت در میان تمام جفت بخش‌های ممکن است. این بیشینه شاخص شباهت به عنوان $S(A,B)$ نمایانگر شباهت بین دنباله‌های A و B است. در حالت خاصی که دو دنباله به طور کامل از یکدیگر متفاوت هستند، $S(A,B)$ برابر با 0 مقداردهی شده و شاخص شباهت $S(A,B)$ به صورت زیر مشخص می‌شود:

[۱] SVM
[۲] KNN
[۳] Structure Feature Extractor (SFE)
[۴] gaps

$$S(A,B)=\max_{1\le i\le p\le M,1\le j\le q\le N}\left(I\left(A_{i:p},B_{j:q}\right),0\right)$$

الگوریتم یک روش سیستماتیک برای تعیین شاخص شباهت $S(A,B)$ و ترتیب‌بندی محلی متناظر آن ارائه می‌دهد. این الگوریتم ماتریس شاخص H را ایجاد می‌کند که عنصر $H_{p+1,q+1}$ بیشینه شاخص را معین می‌کند. اگر شاخص شباهت مثبتی وجود نداشته باشد، $H_{p+1,q+1}=0$ خواهد بود. به عبارت دیگر:

$$H_{p+1,q+1}=\max_{1\le i\le m,1\le j\le n}\left(I\left(A_{i:p},B_{j:q}\right),0\right)$$

وقتی یک یا هر دوی بخش‌ها خالی باشند، شاخص شباهت جفت بخش‌ها برابر با 0 است. به عبارت دیگر، $H_{1,q}=0$ و $H_{p,1}=0$ برای هر p و q. مقدار $H_{p+1,q+1}$ می‌توان با استفاده از معادله زیر به صورت بازگشتی با جریمه یکنواخت δ محاسبه کرد:

$$H_{p+1,q+1}=max\left(H_{p,q}+s\left(\alpha_p,\beta_q\right),H_{p,q+1}+\delta,H_{p+1,q}+\delta,0\right) \quad (۳)$$

حال به صورت خلاصه الگوریتم را می‌توان به شرح زیر توصیف کرد:

- ساختن ماتریس $H\in R(M+1)\times(N+1)$ و مقداردهی اولیه سطر و ستون اول H به صفر.
- محاسبه مقادیر دیگر در ماتریس H از گوشه سمت چپ بالا تا گوشه راست پایین با استفاده از معادله (۳).
- شناسایی بیشینه مقدار در ماتریس H که نمایانگر شاخص شباهت دو دنباله $S(A,B)$ است.
- دنبال کردن المان‌های ماتریس از این بیشینه مقدار تا رسیدن به یک مقدار صفر مسیر ترتیب‌بندی محلی بهینه را نشان می‌دهد.

برای سازگار کردن این الگوریتم به منظور پیاده‌سازی بر روی داده‌های هشدار نیاز به اعمال تغییراتی هست و [۳۶] به این موضوع اختصاص یافته است. در این مقاله تغییراتی برای الگوریتم برای توالی‌های هشدار با برچسب زمانی شامل بازتعریف امتیاز شباهت یک جفت نمادین به امتیاز شباهت یک جفت هشدار پیشنهاد شده است. به علاوه در این مقاله دو مفهوم جدید معرفی شده است، بردار فاصله زمانی

$$\boldsymbol{d}_m=[d_m^1,d_m^2,\dots,d_m^K]^T$$

و بردار وزن زمانی

$$\boldsymbol{w}_m=[f(d_m^1),f(d_m^2),\dots,f(d_m^K)]^T$$

بردار فاصله زمانی برای هر هشدار تعریف شده و اطلاعاتی در مورد فاصله زمانی بین هشدار m ام و نزدیک‌ترین هشدار با نوع k را شامل می‌شود. اگر هیچ هشداری از نوع k وجود نداشته باشد، فاصله زمانی بی‌نهایت است. سپس بردار وزن زمانی تعریف می‌شود که با تابع وزن زمانی $f(x)$ ارتباط دارد که اگر فاصله زمانی به بی‌نهایت برسد، وزن به صفر می‌رسد. همچنین در [۳۶]، از تابع گاوسی مقیاس‌شده $f(x)=e^{-x^2/2^2}$ به عنوان تابع وزن استفاده می‌شود.

در [۳۷] برای بررسی الگو بین بیش از دو دنباله روش معرفی شده در [۳۶] تعمیم داده شده است. این مقاله رابطه

$$S'(A,B,C)=max\{\frac{S(B,A)+S(C,A)}{2},\frac{S(A,B)+S(C,B)}{2},\frac{S(A,C)+S(B,C)}{2}\}$$

را به عنوان شاخص شباهت بین سه دنباله A، B و C در نظر گرفته و در ادامه روشی مشابه [۳۶] را برای بررسی مشابهت بین این دنباله‌ها معرفی می‌کند.

مقاله [۳۸] روش محاسبه جریمه فاصله خالی که در [۳۶] پیشنهاد شده بود را تغییر داده و با معیارهای اندازه‌گیری فاصله زمانی مرتبط کرده‌اند تا تاثیرات هشدارهای نامرتبط بر نتایج تطابق کاهش یابد؛ همچنین یک استراتژی افزایشی برای محاسبه فاصله زمانی و ماتریس‌ها هنگام تطابق توالی‌ها معرفی شده است که به افزایش سرعت اجرای الگوریتم کمک می‌کند.

در [۳۹]، روش سیستماتیکی برای تطابق الگوی هشدارهای مشابه در فرآیندهای متفاوت یک سیستم هشدار ارائه شده است. این روش شامل دو گام اصلی است، ابتدا تعمیم نمایش هشدارها و سپس تحلیل شباهت بین انبوه هشدارها بر اساس روش Smith–Waterman تعمیم یافته. خوشه‌های پیدا شده از هجوم‌های مشابه می‌توانند به یافتن علل ریشه‌ای مشترک کمک کنند، که منجر به حل عمومی برای رفع هجوم‌های هشدار از فرآیندهای متفاوت می‌شود.

در [۴۰] دنباله‌های هجوم هشدار به صورت ترکیبی از برچسب‌های هشدار و شدت اهمیت آن برچسب هشدار در نظر گرفته شده است. سپس الگوریتم Smith-Waterman با اصلاح استراتژی امتیاز براساس اولویت‌های هشدار و یک روش تجزیه و تحلیل دنباله مبتنی بر امتیاز شباهت برای کاهش هزینه محاسباتی پیشنهاد داده شده است.

۲-۶ الگوریتم Needleman-Wunsch

الگوریتم Needleman-Wunsch، که ابتدا در [۴۱] معرفی شده و بر خلاف الگوریتم SWA، یک تطابیق عمومی[۱] ارائه می‌دهد. در این الگوریتم، یک ماتریس ماتریس به نام H ایجاد می‌شود که هر عنصر آن، یعنی

[۱] Global alignment

$H_{p+1,q+1}$، برای یک جفت از هشدارها یعنی α_p و β_q بر اساس رابطه زیر محاسبه می‌شود:

$$H_{p+1,q+1} = \max\{H_{p,q} + S(\alpha_p, beta_q), H_{p,q+1} + \delta, H_{p+1,q} + \delta\}$$

که در آن، $p = 1,2,\dots,M$ و $q = 1,2,\dots,N$ است. مقادیر اولیه سطر اول و ستون اول ماتریس H به صورت زیر تعریف می‌شود:

$$H_{1,1} = 0، H_{p+1,1} = p\delta، H_{1,q+1} = q\delta$$

پس از محاسبه ماتریس H ، روند پیگیری به عقب از گوشه سمت راست پایینی ماتریس شروع شده و ادامه می‌یابد تا به گوشه سمت چپ بالایی برسد و در طول این روند، مطابقت بهینه پیدا می‌شود.

در [۴۲]، یک لیست هشدار به دو روش مختلف نمایش داده شده است: به عنوان یک دنباله از هشدارها و به عنوان یک بردار دودویی که هشدارهای موجود (و غیرموجود) در لیست هشدار را نشان می‌دهد. سپس یک دنباله‌ی الگو و یک بردار الگو برای هر هشدار استخراج می‌شود. در این روش دنباله‌ی الگو با استفاده از الگوریتم Needleman-Wunsch مشخص می‌شود و شامل ترتیب هشدارهایی است که در حداقل نصف لیست هشدارها مربوط به یک حالت غیرطبیعی سیستم در همان ترتیب زمانی وجود دارد.

۳-۶ مقایسه NWA و SWA

در مقاله [۴۳] مقایسه‌ای بین دو روش Needleman-Wunsch و Smith-Waterman با استفاده از یک نمونه واقعی صنعتی انجام شده‌است. در SWA تطابق‌یابی محلی صورت می‌پذیرد، یعنی الگوریتم به دنبال یافتن زیر دنباله‌ی محلی بهینه بین دو هجوم هشدار می‌گردد. NWA از سوی دیگر، تطابق عمومی بهینه، یعنی تطابق از ابتدا تا انتها را مدنظر قرار می‌دهد. از سوی دیگر در NWA، امتیاز تشابه بین برچسب‌های هشدار می‌تواند مثبت یا منفی باشد. در SWA، امتیاز تشابه منفی وجود ندارد و در واقع با صفر جایگزین می‌شود. همچنین برای یک جفت هجوم هشدار، طول تطابق محلی بهینه با استفاده از SWA همیشه کوچکتر از طول تطابق عمومی بهینه با استفاده از NWA خواهد بود. باید توجه کرد که ممکن است در نتایج تطابق با استفاده از NWA فاصله‌های بیشتری وجود داشته باشد. به صورت کلی NWA برای هجوم‌های هشدار با طول‌های تقریباً مشابه کاربردی‌تر است.

با توجه به مقایسه دو الگوریتم، هر کدام مزیت‌ها و محدودیت‌های خود را دارند. در کاربردهای واقعی، باید با دقت در انتخاب روش مناسب، به خصوص برای تحلیل شباهت آنلاین هجوم‌های هشدار، برخورد کرد. می‌توان به دو جنبه زیر توجه نمود:

۱. نتایج بررسی دنباله‌های مختلف هجوم هشدار در [۴۳] نشان میدهد NWA در تطابق بیشتر هشدارها بهتر عمل کرد، اما در مقایسه با SWA، تعداد فاصله‌های بیشتری ایجاد کرد. به عبارتی، SWA توانست تراز محلی با کمترین فاصله را ارائه کند و کار با آن در یافتن برچسب‌های هشدار مشابه از دنباله‌های هجوم هشدار با طول بیشتر، ساده‌تر است.

۲. تعداد هشدارهای تکراری (که بعضاً ناشی از نویز یا اختلال هستند) در هجوم هشدار ممکن است بر کیفیت تطابق دنباله تاثیر بگذارد. به دلیل وجود هشدارهای تکراری، برای SWA پیدا کردن یک تراز محلی خوب بین دنباله‌ها سخت می‌شود، زیرا هشدارهای تکراری می‌توانند به تعداد زیادی فاصله منجر شوند، که باعث می‌شود امتیاز در ماتریس شاخص H سریعاً به صفر برسد.

۷ - خوشه‌بندی زودهنگام هجوم هشدار

با توجه به اینکه اکثریت هشدارهای فعال شده در ابتدای یک هجوم هشدار معمولاً نقش مهم‌تری در آن دارند، یک روش خوشه‌بندی به نام تحلیل اجزاء تضعیف شده به صورت نمایی[۱] در [۴۴] پیشنهاد شده است. هر هجوم هشدار به صورت یک بردار دودویی استخراج می‌شود، به صورتی که اطلاعات زمان‌های نسبی فعال شدن در قالب تضعیف نمایی جاسازی شده است. با اختصاص وزن‌های بیشتر به هشدارهای اولیه، EACA تلاش می‌کند تا در مراحل اولیه یک هجوم هشدار، خوشه‌بندی قابل قبولی ارائه کند.

این مقاله ضریب تضعیف را با پارامتر λ مشخص می‌کند که با توجه به ویژگی‌های فرآیند سیستم هشدار مورد نظر مشخص می‌شود و این پارامتر نقش کلیدی در عملکر روش ارائه شده دارد. در این روش، داده‌های هشدار ثبت‌شده از (A&E) به داده‌های دودویی تبدیل می‌شوند، به نحوی که ۱ وضعیت غیرطبیعی و ۰ نشان‌دهنده وضعیت طبیعی باشد. در این صورت هر هجوم هشدار تبدیل به یک بردار دودویی $y \in \mathbb{R}^m$ می‌شود، که m تعداد کل هشدارهای ممکن در سیستم نظارت را مشخص می‌کند. این هشدارها در بردار y کد می‌شوند و هر ورودی نشان‌دهنده حضور یا عدم حضور یک هشدار است. زمان‌های فعال شدن هشدارها در بردار زمان نسبی τ ثبت می‌شود و اگر برخی از هشدارها فعال نشوند، زمان نسبی مربوط به آنها صفر مقداردهی می‌شود. در نتیجه خواهیم داشت

$$z = y \circ exp(-\lambda\tau)$$

که در آن، علامت $\circ$ ضرب هادامارد (یعنی، ضرب عنصر به عنصر) بین دو ماتریس را نشان می‌دهد و $exp()$ تابع نمایی عنصر به عنصر را مشخص می‌کند.

فراهم کردن روشی کارآمد برای تعیین λ برای خوشه‌بندی اولیه دقیق ضروری است. [۴۴] روشی را برای یادگیری λ از داده‌های هجوم هشدار برچسب زده ضبط شده پیشنهاد می‌کند. در کاربردهای عملی، فراهم کردن داده‌های برچسب‌دار کافی برای آموزش آفلاین موثر، دشوار و زمان‌بر است. برای حل این مشکل، یک روش بر مبنای مدل ترکیب گاوسی GMM ([۴۵]) پیشنهاد شده است.

[۱] EACA

۸ -الگوریتم BLAST

الگوریتم BLAST ([۴۶]) یکی دیگر از ابزارهایی است که برای کشف شباهت بین دنباله‌های هشدار مورد استفاده محققین قرار گرفته است. اصلی‌ترین مزیت این الگوریتم، دستیابی سریع به تطابق‌های دنباله‌ها با یافتن مناطقی که شباهت‌های بالایی دارند و کاهش فضای جستجو است. الگوریتم BLAST از دو مرحله اصلی تشکیل شده است:

۱. تخمین: دو دنباله مورد نظر به زیردنباله‌های کوتاه با اندازه ثابت تقسیم شده و یک جدول جستجو برای همه ترکیب‌های ممکن ایجاد می‌شود. با نمایه‌گذاری واژگان کوتاه در جدول جستجو، بخش‌های مشابه در دنباله‌های مورد نظر شناسایی می‌شوند. جفت زیر دنباله‌ها با امتیازهای شباهتی که بیش از آستانه تعیین شده است، به عنوان "تخمین‌ها" نگه‌داشته شده و بقیه زیر دنباله‌ها حذف می‌شود.

۲. گسترش: تمام تخمین‌های شناسایی شده به دو طرف گسترش می‌یابند تا امتیاز شباهت به زیر آستانه‌ای کاهش یابد. تطابق‌هایی که دارای امتیازهای شباهت بالاتر از آستانه خاصی هستند، به‌عنوان «جفت‌های با امتیاز بالا[۱]» نامیده می‌شوند. HSP های با بالاترین امتیاز به عنوان مشابه‌ترین بخش‌ها بین دنباله‌های مورد نظر مشخص می‌شوند.

در [۴۷] از یک الگوریتم تطابق دنباله‌های هشدار مشابه را برای پیش‌بینی و پیشگیری از هجوم‌های هشدار استفاده می‌شود. الگوریتم پیشنهاد شده بر پایه جستجوی تطابق محلی BLAST است. در این مقاله، استراتژی امتیازدهی اولویت‌محور، الگوریتم پیشنهادی را حساس‌تر نسبت به هشدارهای با اولویت مطرح می‌شود و مکانیزم پیش‌تطابق بر پایه مجموعه‌ها، محاسبات غیرضروری را با حذف همه هجوم‌های هشدار و برچسب‌های غیرمرتبط کاهش می‌دهد. همچنین، روش BLAST اصلاح شده برای هجوم‌های هشدار انطباق داده شده‌اند که فضای جستجو را کاهش می‌دهد. طبق مطالعه موردی انجام شده در [۴۷] در مقایسه با الگوریتم Smith–Waterman سریع‌تر عمل می‌کند.

در [۴۸]، الگوریتم BLAST برای استفاده از فاصله Levenshtein به عنوان شاخص شباهت دو دنباله هشدار تطبیق یافته است. به طور خلاصه، فاصله Levenshtein بین دو دنباله حداقل تعداد ویرایش‌های یک عنصر (درج، حذف یا جایگزینی) است که برای تغییر یک دنباله به کلمه دنباله لازم است.

۸-۱ استخراج و تطبیق الگو

استخراج الگو با هدف کشف الگوهای جالب و متداول در داده‌های ضبط شده هشدار صورت می‌پذیرد. این الگوها می‌توانند برای غیر فعال کردن هشدار، تجزیه و تحلیل علت، و پشتیبانی تصمیم‌گیری استفاده شوند. استخراج الگو و تحلیل شباهت هجوم هشدار که پیش‌تر در مورد آن توضیح دادیم دو روش متفاوت برای بررسی داده‌های هجوم هشدار هستند که هر کدام به منظور خاصی طراحی شده‌اند. استخراج الگو مربوط به شناسایی ترتیب‌ها و توالی‌های خاصی از هشدارها است، در حالی که تحلیل شباهت به مقایسه و محاسبه تشابه‌ها و تفاوت‌های میان هشدارها می‌پردازد. در این بخش پژوهش‌های انجام شده در زمینه استخراج الگوهای هشدار را مورد بررسی قرار می‌دهیم.

روش PrefixSpan توسط [۴۹] معرفی شده و در [۵۰] برای مشخص کردن الگوهای پر تکرار در هجوم هشدار مورد استفاده قرار گرفته است. به صورت خلاصه PrefixSpan به شرح زیر عمل می‌کند:

۱. الگوریتم ابتدا تمام آیتم‌های تکرارشده را در توالی‌های هشدار پیدا می‌کند و آن‌ها را به عنوان الگوهای نماینده برای گسترش در نظر می‌گیرد.

۲. برای هر کدام از این الگوها، یک پایگاه داده ایجاد می‌کند که شامل تمام توالی‌هایی است که این الگو در آن‌ها ظاهر می‌شود.

۳. الگوریتم به صورت بازگشتی برای هر پایگاه داده، دو مرحله اول را تکرار می‌کند تا دیگر الگوی نماینده‌ی برای گسترش وجود نداشته باشد.

در [۵۱]، یک الگوریتم PrefixSpan مبتنی بر علیت افزایشی[۲] ارائه شده است. تا الگوهای هشدار پر تکرار و توالی را از هجوم هشدار استخراج کند. الگوی هشدار پر تکرار با علیت باویژگی‌های ترتیبی و تاخیر زمانی هشدارهای توالی بیان می‌شود.

در [۵۲]، روشی برای استخراج الگوهای هشدار از پایگاه داده هجوم هشدار ارائه شده است. این روش بر اساس الگوریتم PrefixSpan بهبود یافته است. ابتدا، استراتژی پیش تطبیق مبتنی بر اولویت برای خوشه‌بندی دنباله‌های مشابه ارائه شده است. سپس، الگوریتم PrefixSpan با در نظر گرفتن برچسب‌های زمانی بهبود یافته است تا تحمل نامعینی کوتاه مدت ترتیب در دنباله‌های هجوم هشدار را فراهم کند. همچنین، یک روش فشرده‌سازی الگوی هشدار برای استخراج بیشتر اطلاعات الگو ارائه شده است تا الگوهای نماینده را مشخص کند.

یک چالش مهم در استخراج الگوی هشدارها این است که هشدارهای خاص و با اولویت بالاتر می‌توانند به راحتی در استخراج الگو نادیده گرفته شوند زیرا معمولاً کمتر به وجود می‌آیند این در حالیست که در اغلب موارد این هشدار های اهمیت بالاتری دارند. با توجه به این مسئله، در [۵۳] برای حل مشکل ابهام توالی و خروجی های تکراری و همچنین در نظرگرفتن هشدارهای با اولویت بالاتر، یک روش استخراج الگوی هشدار بر اساس الگوریتم بهبودیافته PrefixSpan ارائه می‌شود. در ابتدا، یک استراتژی پیش‌تطابق مبتنی بر اولویت ارائه شده است تا توالی‌های مشابه را به طور پیشگویانه خوشه‌بندی کند. در دومین مرحله، PrefixSpan بهبود یافته را با در نظر گرفتن زمان‌های ثبت تطابق برای تطبیق با ابهام ترتیب کوتاه مدت در دنباله‌های هجوم هشدار بهبود می‌دهد. در سومین مرحله، یک روش فشرده‌سازی الگوی هشدار برای استخراج دقیق‌تر اطلاعات الگو جهت تولید الگوهای نماینده ارائه شده است.

[۱] HSP

[۲] Incremental causality

مرجع [۵۴] روشی را برای استخراج الگوهای هشدار با توجه به اولویت از دنباله‌های هجوم‌های هشدار ضبط شده پیشنهاد می‌کند. در ابتدا طراحی یک الگوریتم استخراج الگوی توالی با توجه به اولویت برای استخراج الگوهای توالی هشدار با استفاده از یک الگوریتم استخراج الگوی توالی بسته با اطلاعات اولویت هشدار مورد بررسی قرار گرفته است. استراتژی فشرده‌سازی الگویی پیشنهاد شده است تا دنباله‌های مشابه را به هدف تولید الگوهای جدید ترکیب کند.

روش پیشنهاد شده در [۵۰] که پیش‌تر آن‌را توضح دادیم الگوهای دنباله‌ای هشدار را تشخیص می‌دهند، اما ممکن است الگوهای تکراری شبیه یکدیگر وجود داشته باشند. با توجه به این مورد، در [۵۵] یک روش مبتنی بر ترکیب ClaSP و Top-K که دو روش استخراج الگو هستند پیشنهاد می‌شود. روش Top-K به جای تجزیه و تحلیل همه‌ی هشدارها، فقط k هشدار با بالاترین اهمیت و اولویت را بررسی و مورد بررسی قرار می‌دهد. برای انجام این کار، ابتدا هشدارها براساس یک معیار مشخص مانند اهمیت، اولویت یا ویژگی‌های دیگر رتبه‌بندی می‌شوند. سپس k هشدار با بالاترین امتیازها و اولویت‌ها انتخاب می‌شوند و تحلیل بیشتری روی این هشدارها انجام می‌شود.

در [۵۶] مقایسه ای بین روش افزونگی دوجهتی[۱] (BIDE) و PrefixSpan در یافتن الگوهای متداول در در هجوم هشدار صورت گرفته است. الگوریتم BIDE [(۷)] یک روش برای استخراج دنباله‌های پر تکرار است. این الگوریتم به صورت کارآمدی دنباله‌های متداول بسته را از داده‌های دنباله‌ای استخراج می‌کند.

این الگوریتم به طور خلاصه از دو مرحله اصلی تشکیل شده است:

۱. افزایش رو به جلو[۲]: در این مرحله، دنباله‌های کوچک‌تر از یک عنصر و دارای پشتیبانی کافی تشخیص داده می‌شوند. سپس دنباله‌های جدیدی با اضافه کردن یک عنصر به انتهای دنباله‌های موجود، تولید می‌شوند و پشتیبانی آن‌ها محاسبه می‌شود. اگر پشتیبانی دنباله جدید کافی باشد، دنباله‌ی بسته‌ی متداول تشخیص داده می‌شود.

۲. افزایش رو به عقب: در این مرحله، ترکیب دنباله‌ها از انتها به ابتدا انجام می‌شود. این عمل باعث شناسایی دنباله‌های متداول بسته می‌شود که با حذف یک یا چند عنصر آخر، دنباله

دیگری تولید نمی‌شود و همچنین پشتیبانی آن‌ها کافی باشد.

به کمک این دو مرحله و استفاده از روش‌های بهینه‌سازی مانند ساختار درختی، الگوریتم BIDE قادر به مشخص کردن دنباله‌های متداول بسته از داده‌های دنباله‌ای است. به صورت خلاصه می‌توان گفت BIDE با ترکیب دنباله‌های کوچک‌تر به دنباله‌های بزرگ‌تر الگوها را پیدا می‌کند در حالیکه PrefixSpan به صورت بازگشتی به دنبال تمام دنباله‌های زیرمجموعه از داده‌ها می‌گردد که یک زیردنباله مشخص در آن‌ها وجود دارد. پیچیدگی زمانی BIDE به صورت کلی به تعداد دنباله‌های کوچک در داده‌ها مربوط است و PrefixSpan در نیز به تعداد دنباله‌ها و طول آن‌ها وابسته است. در مقایسه موردی انجام شده توسط [۵۶] الگوریتم BIDE سریعتر از الگوریتم PrefixSpan الگوها را عنوان می‌کند.

مقاله [۵۸] یک روش سیستماتیک مبتنی بر CloFAST [(۵۹)] برای استخراج الگوی هجوم هشدار پیشنهاد می‌کند که قابلیت تحمل جابه‌جایی در ترتیب هشدارها و حذف الگوهای تکراری را دارد.

در [۶۰]، یک روش تطابق دنباله به صورت تدریجی ارائه شده است که کمک می‌کند دنباله هجوم‌های هشدار در حال وقوع را با دنباله‌های ضبط شده قبلی با استفاده از مراحل افزایشی مقایسه کند بدون اینکه هر بار که هشدارهای جدید ظاهر می‌شوند مجدداً تطابق کاملی انجام دهد. همچنین، یک استراتژی شاخص‌گذاری معرفی شده است که بخش‌های بی‌اثر هشدارها را از تطابق الگو حذف می‌کند.

۹ روش‌های مبتنی بر پردازش کلمات

به جای استفاده از دنباله‌های دودویی هشدار برای اندازه‌گیری میزان همبستگی بین هشدارها، می‌توان از روش‌های تعبیه کلمات[۳] یک روش تعبیر واژگان است استفاده کرد تا هشدارهای ضبط شده به بردارهایی نگاشت شوند. در این روش به عنوان یک تکنیک پردازش زبان طبیعی، واژه‌هایی که به طور متناوب در نزدیکی یکدیگر در یک متن ظاهر می‌شوند، به صورت متنظار در فضای برداری نیز در نزدیکی هم قرار خواهند گرفت. بنابراین، فاصله بین بردارهای هشدار می‌تواند میزان همبستگی بین هشدارها را نمایان سازد.

Word2Vec که در [۶۱] معرفی شده است یک روش برای تعبیه کلمات داده‌های هشدار یک فرآیند برای کد کردن اطلاعات متنی هشدار ارائه می‌کند. با استفاده از این روش، هشدارهای مختلف به صورت بردارهایی نمایش داده می‌شوند تا بتوان روابط بین هشدارها را بیشتر بررسی کرد. این روش شامل دو مدل کلاسیک یعنی مدل مجموعه پیوسته کلمات[۴] (CBOW) و مدل Skip-gram است.

رابطه ریاضی CBOW به صورت زیر قابل نمایش است:

$$\max \sum_{w \in W} \log P\big(w|context(w)\big),$$

که در آن w کلمه هدف و $context(w)$ کلمات اطراف آن هستند. در Skip-gram، برعکس CBOW، با توجه به یک کلمه مرکزی، کلمات اطراف آن پیش‌بینی می‌شوند. رابطه ریاضی آن به صورت زیر است:

$$\max \sum_{w \in W} \sum_{u \in context(w)} \log P(u|w),$$

که در آن w کلمه مرکزی و u کلمات اطراف w هستند.

با استفاده از این الگوریتم‌ها، هر کلمه به صورت یک بردار در یک فضای n-بعدی نمایش داده می‌شود که n اندازه ویژگی‌ها یا پارامترهایی است که در آموزش مدل استفاده شده‌اند. مدل CBOW هر کلمه را بر اساس کلمات پس‌زمینه پیش‌بینی می‌کند، در حالی که مدل Skip-gram کلمات پس‌زمینه را بر اساس کلمه فعلی پیش‌بینی می‌کند. به دلیل حساسیت به

[۱] Bi-Directional Extension
[۲] forward growth
[۳] word embedding
[۴] Continuous bag-of-words

کلمات کم تکرار، اغلب مدل Skip-gram برای کد کردن داده‌های هشدار استفاده می‌شود. ورودی مدل کل متن، یعنی یک لاگ A&E، و خروجی بردارهای کلمه است. هدف آموزش مدل Skip-gram یافتن بردارهای کلمه است که به پیش‌بینی کلمات پس‌زمینه، یعنی هشدارها در لاگ A&E، کمک می‌کند.

مقاله [۶۲] یک روش خوشه‌بندی هشدار را ارائه می‌دهد که در آن از Word2Vec برای تبدیل هر هشدار به یک بردار متناظر استفاده می‌کند. سپس هشدارها با استفاده از ترکیبی از روش‌های K-means و AHC[۱] خوشه‌بندی شده‌اند. الگوریتم K-means روش خوشه‌بندی است و در این روش ابتدا تعداد خوشه‌ها را مشخص کرده و مرکزهای اولیه خوشه‌ها را تعیین می‌کنیم. سپس داده‌ها به نزدیک‌ترین مرکز خوشه تخصیص داده می‌شوند و مرکزهای خوشه با میانگین داده‌های خود به‌روزرسانی می‌شوند. این روند تا زمانی ادامه دارد که دیگر مرکزهای خوشه‌ها تغییر نکنند و در نهایت داده‌ها به خوشه‌هایی بهینه تقسیم می‌شوند.

AHC یک روش دیگر برای خوشه‌بندی است که با شروع از تک خوشه‌ها و ادغام تدریجی آن‌ها، یک ساختار سلسله مراتبی از آن‌ها ایجاد می‌کند. در این روش، ابتدا هر داده به عنوان یک خوشه مجزا در نظر گرفته می‌شود. سپس خوشه‌ها با توجه به شباهت‌هایشان ادغام می‌شوند تا در نهایت یک درخت سلسله مراتبی از خوشه‌ها به دست آید.

در نهایت در [۶۲] پس از ترکیب K-means و AHC و خوشه‌بندی هشدارها، نتایج را با استفاده از روش MDS[۲] در یک نمودار دو بعدی نشان می‌دهد. MDS یا تجزیه و تحلیل چندبعدی یک روش است که از طریق تبدیل فضای بعد بالای داده‌ها به فضای بعد کمتر (دو بعد در [۶۲])، نمایشی از فواصل میان داده‌ها را ارائه می‌کند.

در [۶۳]، ابتدا از یک روش تغییر یافته مبتنی بر Bag-of-Words برای تبدیل هجوم‌های هشدار به بردارها استفاده می‌کند. سپس یک مدل مبتنی بر یادگیری ماشینی برای دسته‌بندی هجوم‌های هشدار به صورت آنلاین ارائه می‌کند. یک دسته‌بندی کلاسیک مبتنی بر یادگیری ماشینی طبق فرض مجموعه بسته کار می‌کند، به طوری که همه داده‌های جدیدی که در حال دسته‌بندی هستند باید از دسته‌هایی باشند که در مجموعه داده ضبط شده قبلی ظاهر شده‌اند. این مورد ممکن است همواره درست نباشد و حالت‌های جدید نیز رخ دهد. دسته‌بندی مجموعه باز به روشی اشاره دارد که در آن می‌توان با اضافه کردن گزینه «رد کردن»، روش دسته‌بندی را قادر به اجتناب از دسته‌بندی‌های نادرست می‌کند. این مکانیزم در واقع دسته‌بندی ای که با امتیاز کم دسته‌بندی شده است را حذف می‌کند.

در [۶۴] نیز از یک روش مبتنی بر CBOW برای کدگذاری‌داده‌های هجوم هشدار استفاده می‌شود و سپس، از فاصله جابجایی-تقسیم-ادغام[۳] (MSM) بین بردارها برای اندازه‌گیری شباهت هشدارهای استفاده می‌شود. "جابه‌جایی" به معنای تغییر مکان یا انتقال اجزاء تصویر یا دنباله از یک موقعیت به موقعیت دیگر است. "تقسیم" به معنای تقسیم تصویر یا دنباله به بخش‌های کوچک‌تر می‌باشد. "ادغام" به معنای ادغام یا ترکیب بخش‌های مختلف تصویر یا دنباله به یکدیگر است. فاصله "جابجایی-تقسیم-ادغام" برای مقایسه‌ی دو تصویر یا دنباله، ابتدا تغییرات جابجایی، تقسیم و ادغامی که برای تبدیل یک تصویر یا دنباله به دیگری لازم است، محاسبه می‌شود. سپس این تغییرات به عنوان یک معیار برای میزان تفاوت یا شباهت بین دو تصویر یا دنباله استفاده می‌شود.

در [۶۵]، روش جدیدی برای تطابق الگو بر اساس Word2Vec و پیچش پویای زمان[۴] برای شناسایی گروه‌هایی از هجوم‌های هشدار مشابه ارائه شده است. سپس، شباهت‌ها بین دنباله‌های هجوم‌های هشدار تبدیل شده به بردار با استفاده از DTW محاسبه می‌شود و گروه‌هایی از هجوم‌های هشدار مشابه از طریق خوشه‌بندی شناسایی می‌شوند. به عبارت دیگر، DTW یک تطبیق غیرخطی بین مشاهدات دو دنباله پیدا می‌کند و بنابراین حساس به اختلاف‌ها در زمان وقوع هشدارها (جلوتر یا عقب‌تر بودن یک هشدار در دو دنباله) نیست. این روش از دو مرحله اصلی تشکیل شده است. ابتدا ماتریس فاصله بین نقاط دو سری زمانی ایجاد می‌شود. سپس ماتریس تطابقی بر اساس ماتریس فاصله ایجاد می‌شود که کمترین هزینه برای مسیر تطابقی را نشان می‌دهد. با پیدا کردن مسیری با کمترین هزینه و تطابق دو سری زمانی، می‌توان شباهت‌ها و تفاوت‌ها را میان دو سری زمانی تشخیص داد.

در [۶۶]، ابتدا گراف وزن‌دار داده‌های هجوم هشدار تشکیل داده می‌شود به به نحوی که، تعداد گره‌ها برابر با تعداد هشدارهای منحصر به فرد است، و ماتریس شامل وزن‌های از شدت رابطه‌ی هشدارها با یکدیگر است سپس، گراف وزن‌دار به بردارهای منحصر به فرد برای هر هشدار با استفاده از الگوریتم Node2Vec تبدیل می‌شود. Node2Vec یک الگوریتم یادگیری نمایش برای گراف‌ها است که ابتدا در [۶۷] معرفی شده است و کمک می‌کند تا گره‌های یک گراف را به فضای برداری منتقل کنیم. در نهایت در [۶۶] این بردارها با استفاده از روش پیشنهادی خوشه‌بندی می‌شوند.

نتیجه‌گیری

در این مقاله روش‌های مختلفی برای شناسایی، تحلیل، بررسی و نهایتا مقابله با هجوم‌های هشدار مورد بررسی و ارزیابی قرار گرفت. این مطالعه نشان داد که این چالش پیچیده در محیط‌های صنعتی به طور گسترده‌ای در پژوهش‌ها مورد توجه قرار گرفته و از اهمیت ویژه‌ای برخوردار است. روش‌های مختلفی مانند استفاده از الگوریتم‌های داده‌کاوی، تجزیه و تحلیل سیگنال‌ها، تکنیک‌های تجزیه و تحلیل زمانی و تجزیه و تحلیل تصاویر به منظور تشخیص و پیش‌بینی هجوم‌های هشدار مطرح شدند. هر یک از این روش‌ها دارای مزایا و محدودیت‌های خود بوده و انتخاب روش مناسب بر

[۱] Agglomerativ Hierarchical Clustering
[۲] Multi-Dimension Scaling
[۳] Move-Split-Merge
[۴] Dynamic time warping

اساس ویژگی‌ها و نیازهای محیط صنعتی مورد استفاده، مورد تاکید قرار گرفته است.

مراجع

[1] F. E. Mustafa, I. Ahmed, A. Basit, S. H. Malik, A. Mahmood, P. R. Ali, et al., "A review on effective alarm management systems for industrial process control: barriers and opportunities," International Journal of Critical Infrastructure Protection, p.100599, 2023.

[2] J. Wang, F. Yang, T. Chen, and S. L. Shah, "An overview of industrial alarm systems: Main causes for alarm overloading, research status, and open problems," IEEE Transactions on Automation Science and Engineering, vol.13, no.2, pp.1045–1061, 2015.

[3] I.Izadi, S. L. Shah, D. S. Shook, and T. Chen, "An introduction to alarm analysis and design," IFAC-PapersOnLine, vol.42, no.8, pp.645–650, 2009.

[4] ISA. Management of Alarm Systems for the Process Industries. International Society of Automation, 2009.

[5] J. Xu, J. Wang, I. Izadi, and T. Chen, "Performance assessment and design for univariate alarm systems based on FAR, MAR, and AAD," IEEE Transactions on Automation Science and Engineering, vol.9, no.2, pp.296–307, 2011.

[6] B. R. Hollifield and E. Habibi. Alarm management: A comprehensive guide: Practical and proven methods to optimize the performance of alarm management systems. ISA, 2011.

[7] H. S. Alinezhad, M. H. Roohi, and T. Chen, "A review of alarm root cause analysis in process industries: Common methods, recent research status and challenges," Chemical Engineering Research and Design, 2022.

[8] D. H. Rothenberg. Alarm Management for Process Control: a Best-practice Guide for Design, Implementation, and Use of Industrial Alarm Systems. Momentum Press, 2009.

[9] EEMUA. Alarm Systems-A Guide to Design, Management and Procurement, vol.191. EEMUA Publication, 2013.

[10] B. Zhou. Advanced Methods for Alarm Monitoring and Alarm Flood Analysis Based on Industrial Data. University of Alberta, 2021.

[11] A.Tulsyan and R. B. Gopaluni, "Univariate model-based deadband alarm design for nonlinear processes," Industrial & Engineering Chemistry Research, vol.58, no.26, pp.11295–11302, 2019.

[12] M. S. Afzal, T. Chen, A. Bandehkhoda, and I. Izadi, "Analysis and design of time- deadbands for univariate alarm systems," Control Engineering Practice, vol.71, pp.96– 107, 2018.

[13] N. A. Adnan, I. Izadi, and T. Chen, "On expected detection delays for alarm systems with deadbands and delay-timers," Journal of Process Control, vol.21, no.9, pp.1318–1331, 2011.

[14] N. A. Adnan, Y. Cheng, I. Izadi, and T. Chen, "Study of generalized delay-timers in alarm configuration," Journal of Process Control, vol.23, no.3, pp.382–395, 2013.

[15] Y. Cheng, I. Izadi, and T. Chen, "Optimal alarm signal processing: Filter design and performance analysis," IEEE Transactions on Automation Science and Engineering, vol.10, no.2, pp.446–451, 2013.

[16] M. H. Roohi and T. Chen, "Generalized moving variance filters for industrial alarm sys tems," Journal of Process Control, vol.95, pp.75–85, 2020.

[17] M. H. Roohi and T. Chen, "Performance assessment and design of quadratic alarm filters," IFAC-PapersOnLine, vol.53, no.2, pp.494–499, 2020.

[18] M. H. Roohi, T. Chen, and I. Izadi, "Control and alarm interplay and robust state-feedback synthesis with an alarm performance constraint," Industrial & Engineering Chemistry Re- search, vol.59, no.38, pp.16708–16719, 2020.

[19] M. H. Roohi, T. Chen, Z. Guan, and T. Yamamoto, "A new approach to design alarm filters using the plant and controller knowledge," Industrial & Engineering Chemistry Research, vol.60, no.9, pp.3648–3657, 2021.

[20] V. Rodrigo, M. Chioua, T. Hagglund, and M. Hollender, "Causal analysis for alarm flood reduction," IFAC-PapersOnLine, vol.49, no.7, pp.723–728, 2016.

[21] T. Yuan and S. J. Qin, "Root cause diagnosis of plant-wide oscillations using granger causality," Journal of Process Control, vol.24, no.2, pp.450–459, 2014.

[22] Q. Chen, X. Lang, S. Lu, N. ur Rehman, L. Xie, and H. Su, "Detection and root cause analysis of multiple plant-wide oscillations using multivariate nonlinear chirp mode de- composition and multivariate granger causality," Computers & Chemical Engineering, vol.147, p.107231, 2021.

[23] P. Duan, F. Yang, T. Chen, and S. L. Shah, "Direct causality detection via the trans- fer entropy approach," IEEE Transactions on Control Systems Technology, vol.21, no.6, pp.2052–2066, 2013.

[24] Q.-Q. Meng, Q.-X. Zhu, H.-H. Gao, Y.-L. He, and Y. Xu, "A novel scoring function based on family transfer entropy for Bayesian networks learning and its application to industrial alarm systems," Journal of Process Control, vol.76, pp.122–132, 2019.

[25] Q.-X. Zhu, W.-J. Ding, and Y.-L. He, "Novel multimodule Bayesian network with cyclic structures for root cause analysis: Application to complex chemical processes," Industrial & Engineering Chemistry Research, vol.59, no.28, pp.12812–12821, 2020.

[26] M. H. Roohi, P. Ramazi, and T. Chen, "Towards accurate root-alarm identification: The causal Bayesian network approach," in International Conference on Control and Fault-Tolerant Systems, pp.169–174, IEEE, 2021.

Electronics and Communication; Network and Computer Technology, vol.12167, pp.723–728, SPIE, 2022.
[41] S. B. Needleman and C. D. Wunsch, "A general method applicable to the search for simi larities in the amino acid sequence of two proteins," Journal of Molecular Biology, vol.48, no.3, pp.443–453, 1970.
[42] S. Charbonnier, N. Bouchair, and P. Gayet, "Fault template extraction to assist operators during industrial alarm floods," Engineering Applications of Artificial Intelligence, vol.50, pp.32–44, 2016.
[43] M. R. Parvez, W. Hu, and T. Chen, "Comparison of the smith-waterman and Needleman- Wunsch algorithms for online similarity analysis of industrial alarm floods," in IEEE Electric Power and Energy Conference, pp.1–6, IEEE, 2020.
[44] J. Shang and T. Chen, "Early classification of alarm floods via exponentially attenuated component analysis," IEEE Transactions on Industrial Electronics, vol.67, no.10, pp.8702–8712, 2019.
[45] H. S. Alinezhad, J. Shang, and T. Chen, "Early classification of industrial alarm floods based on semi-supervised learning," IEEE Transactions on Industrial Informatics, vol.18, no.3, pp.1845–1853, 2021.
[46] S. F. Altschul, W. Gish, W. Miller, E. W. Myers, and D. J. Lipman, "Basic local alignment search tool," Journal of Molecular Biology, vol.215, no.3, pp.403–410, 1990.
[47] W. Hu, J. Wang, and T. Chen, "A local alignment approach to similarity analysis of industrial alarm flood sequences," Control Engineering Practice, vol.55, pp.13–25, 2016.
[48] Y. Xu, W. Tan, and T. Li, "An alarm flood pattern matching algorithm based on modified blast with Leveshtein distance," in International Conference on Control, Automation, Robotics and Vision, pp.1–6, IEEE, 2016.
[49] J. Han, J. Pei, B. Mortazavi-Asl, H. Pinto, Q. Chen, U. Dayal, and M. Hsu, "Prefixspan: Mining sequential patterns efficiently by prefix-projected pattern growth," in Inter- national Conference on Data Engineering, pp.215–224, IEEE, 2001.
[50] T. Niyazmand and I. Izadi, "Pattern mining in alarm flood sequences using a modified Prefixspan algorithm," ISA Transactions, vol.90, pp.287–293, 2019.
[51] J. Wang, R. Jia, J. Zhou, and M. Zhou, "Mining sequential alarm pattern based on the incremental causality Prefixspan algorithm," IEEE Transactions on Artificial Intelligence, 2022.
[52] Q.-X. Zhu, C. Jin, Y.-L. He, and Y. Xu, "Pattern mining of alarm flood sequences using an improved Prefixspan algorithm with tolerance to short-term order ambiguity," Industrial & Engineering Chemistry Research, vol.60, no.11, pp.4375–4384, 2021.
[53] S. Yang, T. Zhang, Y. Zhai, K. Wang, G. Zhao, Y. Tu, and L. Cheng, "Frequent alarm pattern mining of industrial alarm flood sequences by an
[27] H. S. Alinezhad, J. Shang, and T. Chen, "Open set online classification of industrial alarm floods with alarm ranking," IEEE Transactions on Instrumentation and Measurement, vol.72, pp.1–11, 2022.
[28] S. R. Kondaveeti, I. Izadi, S. L. Shah, T. Black, and T. Chen, "Graphical tools for routine assessment of industrial alarm systems," Computers & Chemical Engineering, vol.46, pp.39–47, 2012.
[29] K. Ahmed, I. Izadi, T. Chen, D. Joe, and T. Burton, "Similarity analysis of industrial alarm flood data," IEEE Transactions on Automation Science and Engineering, vol.10, no.2, pp.452–457, 2013.
[30] W. Hu, A. W. Al-Dabbagh, T. Chen, and S. L. Shah, "Design of visualization plots of industrial alarm and event data for enhanced alarm management," Control Engineering Practice, vol.79, pp.50–64, 2018.
[31] Z. Mannani, I. Izadi, and N. Ghadiri, "Preprocessing of alarm data for data mining," Industrial & Engineering Chemistry Research, vol.58, no.26, pp.11261–11274, 2019.
[32] T. Niyazmand and I. Izadi, "Identification and clustering of alarm floods in a natural gas processing plant," in Iranian Conference on Electrical Engineering, pp.656– 660, IEEE, 2017.
[33] M. Lucke, M. Chioua, C. Grimholt, M. Hollender, and N. F. Thornhill, "Online alarm flood classification using alarm coactivations," IFAC-PapersOnLine, vol.51, no.18, pp.345–350, 2018.
[34] C. Tian, P. Song, C. Zhao, and J. Ding, "Structure feature extraction for hierarchical alarm flood classification and alarm prediction," IEEE Transactions on Automation Science and Engineering, 2023.
[35] T. F. Smith, M. S. Waterman, et al., "Identification of common molecular subsequences," Journal of Molecular Biology, vol.147, no.1, pp.195–197, 1981.
[36] Y. Cheng, I. Izadi, and T. Chen, "Pattern matching of alarm flood sequences by a modified smith–waterman algorithm," Chemical Engineering Research and Design, vol.91, no.6, pp.1085–1094, 2013.
[37] S. Lai and T. Chen, "Methodology and application of pattern mining in multiple alarm flood sequences," IFAC-PapersOnLine, vol.48, no.8, pp.657–662, 2015.
[38] S. Lai, F. Yang, and T. Chen, "Online pattern matching and prediction of incoming alarm floods," Journal of Process Control, vol.56, pp.69–78, 2017.
[39] B. Zhou, W. Hu, K. Brown, and T. Chen, "Generalized pattern matching of industrial alarm flood sequences via word processing and sequence alignment," IEEE Transactions on Industrial Electronics, vol.68, no.10, pp.10171–10179, 2020.
[40] C. Li, Y. Tu, S. Gu, Y. Zheng, X. Yang, C. Li, Y. Ke, and J. Hu, "Pattern matching of alarm sequences by using an improved smith-waterman algorithm," in International Conference on

improved Prefixspan algorithm," Processes, vol.11, no.4, p.1169, 2023.
[54] W. Hu, Z. Wang, and J. Wang, "A priority-aware sequential pattern mining method for detection of compact patterns from alarm floods," Journal of Process Control, vol.129, p.103041, 2023.
[55] Z. Wang, W. Hu, W. Cao, and M. Wu, "Detection of sequential alarm patterns in complex industrial facilities using clasp and top-k algorithms," in Chinese Control Conference, pp.4671–4676, IEEE, 2021.
[56] C. Belavadi, V. S. Sardar, and S. S. Chaudhari, "Alarm pattern recognition in continuous process control systems using data mining," International Journal of Computing, vol.21, no.3, pp.333–341, 2022.
[57] J. Wang and J. Han, "Bide: Efficient mining of frequent closed sequences," in International Conference on Data Engineering, pp.79–90, IEEE, 2004.
[58] B. Zhou, W. Hu, and T. Chen, "Pattern extraction from industrial alarm flood sequences by a modified Clofast algorithm," IEEE Transactions on Industrial Informatics, vol.18, no.1, pp.288–296, 2021.
[59] F. Fumarola, P. F. Lanotte, M. Ceci, and D. Malerba, "Clofast: closed sequential pattern mining using sparse and vertical id-lists," Knowledge and Information Systems, vol.48, pp.429–463, 2016.
[60] M. R. Parvez, W. Hu, and T. Chen, "Real-time pattern matching and ranking for early prediction of industrial alarm floods," Control Engineering Practice, vol.120, p.105004, 2022.
[61] X. Rong, "word2vec parameter learning explained," arXiv preprint arXiv:1411.2738, 2014.
[62] S. Cai, L. Zhang, A. Palazoglu, and J. Hu, "Clustering analysis of process alarms using word embedding," Journal of Process Control, vol.83, pp.11–19, 2019.
[63] H. S. Alinezhad, J. Shang, and T. Chen, "A modified bag-of-words representation for industrial alarm floods," in International Symposium on Advanced Control of Industrial Processes, pp.331–336, IEEE, 2022.
[64] X. Zhang, W. Hu, A. W. Al-Dabbagh, and W. Cao, "Similarity analysis of industrial alarm floods based on word embedding and move-split-merge distance," in International Conference on Industrial Cyber-Physical Systems, pp.1–6, IEEE, 2023.
[65] W. Hu, X. Zhang, J. Wang, G. Yang, and Y. Cai, "Pattern matching of industrial alarm floods using word embedding and dynamic time warping," Journal of Automatica Sinica, vol.10, no.4, pp.1096–1098, 2023.
[66] H. Khaleghy and I. Izadi, "Detection of correlated alarms using graph embedding," in International Conference on Signal Processing and Intelligent Systems, pp.1–7, IEEE, 2021.
[67] Grover and J. Leskovec, "Node2vec: Scalable feature learning for networks," in ACM International Conference on Knowledge Discovery and Data Mining, pp.855–864, 2016.