# پنهان‌سازی مجموعه عناصر حساس از طریق حذف تراکنش‌های حساس مرتب‌سازی شده با الگوریتم ژنتیک چند هدفه

کبری قاسمی[۱]، دانشجوی کارشناسی ارشد؛ بهزاد زمانی دهکردی[۲]، استادیار؛ فرساد زمانی بروجنی[۳]، استادیار

۱- دانشکده فنی و مهندسی - واحد اصفهان (خوراسگان) - دانشگاه آزاد اسلامی - اصفهان - ایران - k.ghasemi.p@gmail.com
۲- دانشکده فنی و مهندسی - واحد شهرکرد - دانشگاه آزاد اسلامی - شهرکرد - ایران - bzamani@iaushk.ac.ir
۳- دانشکده فنی و مهندسی - واحد اصفهان (خوراسگان) - دانشگاه آزاد اسلامی - اصفهان - ایران - farsad.zamani@yahoo.com

**چکیده:** قواعد انجمنی، برای یافتن ارتباط پنهان و وابستگی‌های میان مجموعه عناصر مختلف در پایگاه داده به کار می‌روند که در قالب قانون استخراج می‌شوند؛ اما مشکل این روش، افشاء اطلاعات حساس و تهدید محرمانگی اطلاعات می‌باشد. فرایند ایمن‌سازی داده‌ها باوجود تحقیقات گسترده در این حوزه به‌عنوان یک مسئله NPHard در نظر گرفته می‌شود. این مقاله با استفاده از الگوریتم ژنتیک چندهدفه و نیز رویکرد مبتنی بر پشتیبان، سعی در کاهش پشتیبانی مجموعه عناصر حساس موجود در پایگاه داده تراکنشی دارد. روش پیشنهادی با حذف تراکنش‌هایی که شامل عناصر حساس هستند، باعث کاهش پشتیبانی عناصر حساس به کمتر از حداقل آستانه پشتیبانی شده که ایمن‌سازی پایگاه داده را به همراه دارد. روش پیشنهادی در هر تکرار، تنها با یک‌بار پویش تراکنش‌های حساس به‌جای پویش کل تراکنش‌های پایگاه داده، باعث افزایش سرعت و کاهش هزینه‌های اجرا می‌گردد. همچنین برای کاهش عوارض ناشی از پنهان‌سازی، تراکنش‌ها بر اساس کمترین طول یا بیشترین عنصر حساس و کمترین عنصر غیر حساس مرتب‌سازی می‌شوند.

**واژه‌های کلیدی:** قواعد انجمنی، پنهان‌سازی مجموعه عناصر حساس، الگوریتم‌های ژنتیک چندهدفه.

# Hiding the Sensitive Itemsets through the Ordered Sensitive Transactions Deletion via Multi-Objective Genetic Algorithms

K. Ghasemi[1], MSc Student; B. Zamani Dehkordi[2], Assistant Professor; F. Zamani Boroujeni[3], Assistant Professor

1- Faculty of Engineering, Isfahan (Khorasgan) Branch, Islamic Azad University, Isfahan, Iran. k.ghasemi.p@gmail.com
2- Faculty of Computer Engineering, Shahrekord Branch, Islamic Azad University, Shahrekord, Iran. bzamani@iaushk.ac.ir
3- Faculty of Engineering, Isfahan (Khorasgan) Branch, Islamic Azad University, Isfahan, Iran. farsad.zamani@yahoo.com

**Abstract**: Association rules are used to find hidden relationships and dependencies among different itemsets in the database that is extracted in the form of the rule, but the problem with this approach is the discovery of sensitive information and the treatment of information privacy. The sanitization process data is considered as a NPHard problem. In this article, we try to reduce support the sensitive itemsets in the transactional database using multi-objective genetic algorithms and the support-based approach. The proposed approach with the transaction deletion that includes sensitive itemsets leads to less support sensitive itemsets than the minimum support threshold and leads to the database sanitization. In each iteration of our method leads to increase the speed and reduce the performance criteria by one time of the scanning of the sensitive transaction instead of scanning the entire database of transactions. In addition, to reduce the effects of hiding the sensitive itemsets, the transactions sort based on the shortest length, the most sensitive itemsets and the least non-sensitive itemsets.

**Keywords:** Association rules, hiding the sensitive itemsets, multi-objective genetic algorithm.

تاریخ ارسال مقاله: ۱۳۹۵/۱۰/۱۱
تاریخ اصلاح مقاله: ۱۳۹۵/۱۲/۲۳
تاریخ پذیرش مقاله: ۱۳۹۶/۰۵/۰۹
نام نویسنده مسئول: بهزاد زمانی دهکردی
نشانی نویسنده مسئول: ایران - شهرکرد - رحمتیه - دانشگاه آزاد شهرکرد - دانشکده فنی و مهندسی.

## ۱- مقدمه

حفظ حریم خصوصی افراد از موضوعات بسیار بااهمیت برای افراد و سازمان‌ها، در حوزه‌های مختلفی ازجمله داده‌کاوی مطرح‌شده است، به‌گونه‌ای که با انجام تکنیک‌های مختلف داده‌کاوی و تحلیل داده‌ها، احتمال افشای اطلاعات و تهدید حریم خصوصی کاهش یابد. بااینکه مزایای استفاده از داده‌کاوی بر همگان روشن و مشخص است و اعمال فرایند داده‌کاوی بر روی مجموعه بسیار بزرگی از اطلاعات شخصی، موجب دسترسی به مدل و الگوها می‌شود اما جمع‌آوری و تحلیل داده‌های شخصی حساس، منجر به ایجاد نگرانی‌هایی درزمینه‌ی حریم خصوصی می‌شود. ازاین‌رو مسئله افشای اطلاعات حساس تبدیل به یکی از مهم‌ترین چالش‌های داده‌کاوی شده است[۱].

مسئله حفظ حریم خصوصی در داده‌کاوی[۱] در سال ۲۰۰۰ با انتشار دو مقاله [۲، ۳] با همین عنوان مطرح گردید و تحقیقات در این زمینه همچنان ادامه دارد. فرایند حفظ حریم خصوصی، برای جلوگیری از دسترسی افراد غیرمجاز به مجموعه داده‌های حساس، آن‌ها را به نگارش‌های تغییریافته تبدیل می‌کند. از سوی دیگر، فرایند حفظ حریم خصوصی با پنهان‌سازی این داده‌ها، موجب کاهش بهره‌وری مجموعه داده‌های اصلاح‌شده می‌شود. کاهش بهره‌وری به سطح مشخصی، مانع از تحلیل دقیق داده‌ها می‌گردد که باهدف اصلی داده‌کاوی متناقض خواهد بود، بنابراین هدف اصلی حفظ حریم خصوصی در داده‌کاوی، توسعه الگوریتم‌های اصلاح و پنهان‌سازی داده‌های اصلی است، به صورتی که داده‌های خصوصی بعد از فرایند داده‌کاوی همچنان خصوصی باقی بمانند. روش‌های مختلفی برای حفظ حریم خصوصی در فرایند داده‌کاوی ارائه‌شده است که پنهان‌سازی قواعد انجمنی از مهم‌ترین این روش‌ها می‌باشد.

بسیاری از روش‌های ارائه‌شده برای پنهان‌سازی قواعد انجمنی مبتنی بر روش‌های اکتشافی می‌باشند. الگوریتم‌های اکتشافی دارای کارایی و مقیاس‌پذیری بالا بوده و در بسیاری از تحقیقات استفاده می‌شوند، اما از تأثیرات جانبی ناخواسته رنج می‌برند و یک جواب تقریبی برای مسئله می‌یابند؛ زیرا این الگوریتم‌ها بر اساس بهترین تصمیمات محلی کار می‌کنند، که لزوماً به بهترین جواب عمومی نمی‌رسند. درصورتی‌که روش‌های تکاملی از انتخاب تصادفی استفاده می‌کنند که این انتخاب تصادفی همواره بر روی نتایج خروجی الگوریتم تأثیر مستقیم خواهد داشت و باعث دستیابی به بهترین جواب عمومی می‌شوند.

الگوریتم ژنتیک در کاربردهای مختلفی از قبیل تخمین توابع هسته ماشین بردار پشتیبان در رتبه بندی صفحات وب [۴]، بهینه سازی مسائل پویا [۵] و کاهش تعداد ویژگی در شناسایی بیماری صرع [۶] به‌کار گرفته شده است. در این مقاله ما با استفاده از الگوریتم ژنتیک به‌عنوان یک روش تکاملی، تراکنش‌ها را حذف می‌کنیم. الگوریتم پیشنهادی با استفاده از حذف تراکنش‌ها و رویکرد مبتنی بر پشتیبان، سعی در کاهش پشتیبانی مجموعه عناصر حساس موجود در پایگاه داده تراکنشی دارد؛ از طریق حذف تراکنش‌هایی که شامل عناصر حساس هستند، تنها با یک‌بار پویش تراکنش‌های حساس پایگاه داده، باعث کاهش پشتیبانی عناصر حساس به کمتر از حداقل آستانه پشتیبانی می‌شود و عمل ایمن‌سازی پایگاه داده را انجام می‌دهد.

ساختار ادامه مقاله به این صورت است که کارهای مرتبط با پنهان‌سازی قوانین انجمنی در بخش ۲ موردبررسی قرار می‌گیرند. بخش ۳ به بیان تعاریف اولیه و متغیرهای مورداستفاده می‌پردازد. الگوریتم پیشنهادی در بخش ۴ معرفی می‌گردد. بخش ۵ به معرفی یک مثال می‌پردازد. نتایج آزمایشات در بخش ۶ گزارش می‌شود که الگوریتم پیشنهادی با الگوریتم‌های متداول مورد مقایسه قرار می‌گیرد و در انتها در بخش ۷ بحث و نتیجه‌گیری آورده می‌شود.

## ۲- کارهای مرتبط با پنهان‌سازی قوانین انجمنی

به‌طورکلی الگوریتم‌های پنهان‌سازی قواعد انجمنی به‌عنوان الگوریتم‌های مطرح درزمینه حفظ حریم خصوصی قواعد انجمنی به چهار دسته اصلی رویکرد اکتشافی[۲]، رویکرد دقیق[۳]، رویکرد مبتنی بر مرز[۴] و سایر رویکردها تقسیم می‌شوند، که در ادامه به بررسی این روش‌ها پرداخته می‌شود.

### ۲-۱- رویکرد اکتشافی

این رویکرد شامل روش‌هایی است که فضای حالت مسئله یا همان پایگاه داده را به‌صورت کامل پیمایش نمی‌کنند بلکه سعی دارند با اجرای یک یا چند سیاست هوشمندانه پایگاه داده و تراکنش‌های موجود در آن را بسیار محدود کنند تا سرعت پنهان‌سازی قوانین انجمنی حساس را با کمترین عوارض جانبی افزایش دهند.

در مرجع [۷]، اولین بار یک الگوریتم تجربی برای پنهان‌سازی قوانین انجمنی حساس از طریق کاهش پشتیبانی مجموعه عناصر تولیدکننده قوانین حساس به کمتر از حداقل پشتیبانی تعیین‌شده توسط کاربر ارائه دادند. کاهش میزان پشتیبانی مجموعه عناصر با استفاده از گراف توری[۵] پایگاه داده صورت می‌گیرد. در مرجع [۸]، مسئله پنهان‌سازی را تبدیل به ترکیب مسائل پنهان‌سازی قوانین حساس و پنهان‌سازی مجموعه عناصر حساس نمودند و سه الگوریتم تجربی برای پنهان کردن قوانین انجمنی حساس بر مبنای کاهش پشتیبانی یا اطمینان قوانین و نه هردو مطرح کردند.

در مرجع [۹، ۱۰]، برای اولین بار استفاده از مقادیر نامعلوم را بجای تبدیل صفر به یک و بالعکس را به‌منظور پنهان‌سازی قوانین انجمنی حساس پیشنهاد دادند. تکنیک‌های ذکرشده در این مقاله برای کاربردهایی به کار می‌رود که مقادیر ویژگی‌ها یا محرمانه هستند یا موجود نیستند. بنابراین به‌جای این مقادیر، مقدار نامعلومی مانند علامت سؤال قرار می‌گیرد به صورتی که اثرات جانبی روی قوانین غیر حساس حداقل شود. در مرجع[۱۱]، پنج الگوریتم اکتشافی مختلف ارائه‌شده است که سه الگوریتم 1.a و 1.b و 2.a مبتنی بر کاهش اطمینان قواعد هستند و دو الگوریتم دیگر به نام 2b و 2c مبتنی بر کاهش پشتیبانی مجموعه عناصر هستند به‌طوری‌که پشتیبانی مجموعه‌های عناصر

تولیدکننده قوانین حساس را تا زمانی که کمتر از حداقل پشتیبانی شود کاهش می‌دهند. در مرجع [۱۲]، سه الگوریتم اکتشافی برای پنهان‌سازی قوانین انجمنی پیشنهادشده است. الگوریتم Aggregate مستعدترین تراکنش و الگوریتم Disaggregate مستعدترین عناصر را جهت حذف انتخاب می‌کند و الگوریتم Hybrid ترکیبی از مزایای هر دو الگوریتم موجود را در خود دارد. هر سه رویکرد برای پنهان کردن عناصر حساس از تکنیک کاهش پشتیبانی استفاده می‌کنند.

در مرجع [۱۳]، الگوریتمی به نام DSRRC[۶] معرفی شده است. این الگوریتم روی قوانینی که مقدم و تالی آن‌ها تکی است، کار می‌کند و با خوشه‌بندی قوانین بر اساس عنصر مشترک سمت راست قاعده‌ها، قوانین حساس را پنهان می‌کند. نقاط قوت این الگوریتم نداشتن شکست در عملیات پنهان‌سازی و پایین بودن دفعات اسکن پایگاه داده می‌باشد. در مرجع [۱۴]، الگوریتم DSRRC را اصلاح‌کرده و نام آن را ADSRRC[۷] می‌گذارد. در الگوریتم ADSRRC تنها یک‌بار عمل مرتب‌سازی تراکنش‌ها انجام می‌گیرد. به‌علاوه آن‌ها الگوریتمی به نام RRLR[۸] پیشنهاد کردند که می‌توان آن را بهبودیافته DSRRC دانست. در این الگوریتم برای پنهان کردن قوانین حساس هم پشتیبانی و هم اطمینان قوانین حساس کاهش داده می‌شود. و قوانینی را که دارای چند عنصر در سمت راست خود هستند را نیز می‌تواند پنهان کند.

در مرجع ، برای غلبه بر محدودیت‌های الگوریتم DSRRC، الگوریتم MDSRRC را معرفی کردند. این الگوریتم قوانین حساس با چندین عنصر در سمت چپ و راست را مخفی می‌سازد. این روش با حذف عنصر حساس با بیشترین تکرار از بین عناصر سمت راست و کاهش پشتیبان قوانین حساس عمل پنهان‌سازی را انجام می‌دهد. در مرجع [۱۵]، یک رویکرد حریصانه برای پنهان‌سازی مجموعه اقلام حساس، توسط درج تراکنش‌های جدید به پایگاه داده ارائه‌شده است. قوانین تجربی در توزیع نرمال استاندارد برای تعیین تعداد مناسب تراکنش‌ها اعمال می‌شود؛ که شامل سه مرحله برای درج تراکنش‌های جدید داخل دیتاست اصلی به‌منظور پنهان‌سازی عناصر حساس است. در مرحله اول، محدوده امنی برای عناصر حساس محاسبه می‌شود تا تعداد تراکنش‌هایی که باید در دیتاست درج شوند، مشخص گردد. در مرحله دوم، طول تراکنش‌هایی که باید درج شوند از طریق توزیع نرمال استاندارد ارزیابی می‌شوند. در مرحله سوم، تعداد اختلاف بین مجموعه عناصر حساس پرتکرار و عناصر غیر حساس پرتکرار برای عناصر سطح Kام محاسبه می‌شود. سپس مجموعه عناصر غیر حساس پرتکرار به ترتیب بر اساس تعداد اختلافشان در تراکنش درج می‌شوند.

در مرجع[۱۶] ، الگوریتم اکتشافی با نام MSI- LNSI جهت پنهان‌سازی مجموعه عناصر فراوان حساس بر روی پایگاه داده Mushroom ارائه‌شده است که با در نظر گرفتن ارتباط بین مجموعه عناصر حساس و غیر حساس ارائه‌شده است که سعی می‌شود بهترین تراکنش برای حذف انتخاب شود. که این عمل درنهایت تعداد مجموعه عناصر غیر حساسی که پنهان می‌شوند را کاهش می‌دهد.

در مرجع [۱۷]، برای پنهان‌سازی قواعد انجمنی حساس از تکنیک خوشه‌بندی بر روی مجموعه عناصر سمت راست قواعد و سپس آشفته سازی، مبتنی بر کاهش میزان اطمینان قواعد، استفاده‌شده است. کاهش میزان اطمینان قواعد حساس از طریق کاهش میزان پشتیبانی مجموعه عناصر سمت راست قواعد، و کار بر روی مجموعه تراکنش‌هایی انجام می‌شود که به‌صورت کامل قواعد حساس را پشتیبانی می‌کنند و ابتدا تراکنشی برای تغییر انتخاب می‌شود که دارای کمترین تعداد عناصر باشد.

در مرجع[۱۸]، حفظ حریم خصوصی قواعد انجمنی موردبررسی قرارگرفته است. یک روش جدید بر اساس اعوجاج ارائه‌شده است که پنهان‌سازی قواعد حساس را با حذف برخی از عناصر در پایگاه داده به‌منظور کاهش پشتیبانی و اطمینان قواعد حساس به کمتر از آستانه از پیش تعیین‌شده ممکن می‌سازد. الگوریتم Relevance-Sorting به‌منظور به حداقل رساندن عوارض جانبی بر روی اطلاعات، اطلاعات در مجموعه عناصر غیر حساس موجود در تراکنش برای مرتب کردن پشتیبانی تراکنش‌ها استفاده می‌شوند. تراکنش‌هایی که شامل مجموعه عناصر غیر حساس کمتری هستند ترجیحاً برای اصلاح انتخاب می‌شوند. به‌منظور کاهش درجه اعوجاج، حداقل تعداد تراکنش‌هایی که نیاز به تغییر دارند برای پنهان کردن یک قاعده حساس در نظر گرفته می‌شوند.

در مرجع[۱۹]، یک روش ترکیبی برای پنهان‌سازی قوانین انجمن حساس ارائه‌شده است. کاهش پشتیبانی و کاهش اطمینان قوانین حساس برای انتخاب و اصلاح عناصر موجود در تراکنش‌ها استفاده می‌شود. روش پیشنهادی، ترکیبی از مزایای استفاده از هر دو الگوریتم برای حفظ حریم خصوصی و حفظ کیفیت پایگاه داده است.

### ۲-۲- رویکرد مبتنی بر مرز

در این رویکرد، از مفهوم عناصر مرزی استفاده می‌شود. عناصر مرزی، مجموعه عناصری هستند که به‌عنوان جداکننده مجموعه عناصر فراوان از غیر فراوان هستند و قوانین انجمنی حساس بر اساس تغییر مرزها در گراف عناصر فراوان و غیر فراوان در پایگاه داده اصلی، پنهان خواهند شد. در مرجع [۲۰]، برای اولین بار روش پنهان‌سازی مجموعه عناصر فراوان، با استفاده از مفهوم مرز را پیشنهاد دادند. این الگوریتم با نام BBA به هرکدام از تراکنش‌هایی که حاوی عنصر حساس هستند، وزنی اختصاص می‌دهد و سپس به روش حریصانه با حذف عنصر به دنبال انتخاب عنصری است که کمترین عنصر جانبی را روی مرزهای مجموعه عناصر داشته باشد. ابتدا مرزهای مثبت و منفی را محاسبه می‌کند. سپس عنصری که کمترین اثر جانبی را در پنهان‌سازی دارد را انتخاب می‌کند و مرزهای مثبت را اصلاح می‌کند.

در مرجع[۲۱] ، الگوریتمی به نام MaxMin پیشنهاد دادند که از مفهوم مرزهای مثبت و منفی مجموعه عناصر فراوان استفاده می‌کند. به این صورت که از بین مجموعه عناصر فراوان، عناصری که شامل عنصر حساس هستند در مرز مثبت قرار می‌گیرند و از بین آن‌ها، عنصری که

بیشترین اندازه پشتیبانی را نسبت به سایر عناصر دارد انتخاب می‌شود که مجموعه عناصر MaxMin نامیده می‌شوند و عنصر قربانی نیز از این مجموعه انتخاب می‌شود تا کمترین اثر جانبی را بر روی پایگاه داده بر جای گذارد.

### ۲-۳- رویکرد دقیق

در الگوریتم دقیق، یک فرمول کلی بیان می‌شود و الگوریتم بر اساس آن عمل می‌کند و با استفاده از قوانین ریاضی سعی می‌شود تا حد ممکن دقیق عمل کرده و اثرات جانبی بر روی پایگاه داده به حداقل برسد. حسن این روش‌ها در دقت بالای آن‌ها و عیب آن‌ها در هزینه بالا و زمان اجرای طولانی است.

در مرجع [۲۲]، اولین الگوریتم ترکیبی را ارائه دادند که از دو بخش اکتشافی و دقیق برای پنهان‌سازی قوانین انجمنی حساس استفاده می‌کند. این الگوریتم در بخش رویکرد دقیق، یک برنامه صحیح برای شناسایی حداقل تعداد تراکنش‌هایی که برای پنهان‌سازی تمامی مجموعه عناصر حساس نیاز است، اجرا می‌کند. سپس با استفاده از رویکرد اکتشافی به ازای هر مجموعه عناصر حساس در تراکنش‌های انتخاب‌شده برای تغییر، عنصر پرتکرار را انتخاب می‌کند.

در مرجع [۲۳]، اولین روش دقیق برای تولید پایگاه داده‌های ترکیبی ارائه دادند. در این روش یک فرآیند استاندارد جهت ایمن‌سازی inline داده، ارائه‌شده است که در این فرآیند، جهت انجام عملیات ایمن‌سازی، پایگاه داده اصلی به‌گونه‌ای گسترش داده می‌شود تا بتوان تغییرات لازم جهت پنهان‌سازی عناصر حساس را در آن درج نمود.

### ۲-۴- سایر رویکردها

در مسیر تحقیقات در این حوزه روش‌های ترکیبی نیز ارائه‌شده‌اند که از الگوریتم‌های تکاملی برای حل مسئله پنهان‌سازی استفاده می‌کنند. در مرجع [۲۴]، برای پنهان‌سازی قوانین انجمنی حساس، از الگوریتم ژنتیک استفاده کرده‌اند. در این الگوریتم، هر تراکنش یک کروموزوم است. در الگوریتم‌های ژنتیک برای حل هر مسئله از تابع برازندگی استفاده می‌شود. تابع برازندگی بر اساس فاکتورهای مختلف محاسبه می‌شود. بعد از به دست آوردن مقادیر مربوط به کروموزوم‌ها، تعداد کروموزوم‌هایی که باید تغییر کنند مشخص می‌شوند. سپس با عمل تلفیق بخشی از کروموزوم مقادیر عناصر حساس تغییر می‌کند و درنهایت تراکنش‌های ایمن‌سازی شده به وجود می‌آیند.

در مرجع [۲۵]، از الگوریتم‌های ژنتیک برای ارائه راه‌حل‌های بهینه، جهت پنهان‌سازی عناصر حساس استفاده کرده‌اند. دو الگوریتم مبتنی بر ژنتیک به نام sGA2DT[۹] و pGA2DT[۱۰] پیشنهاد دادند که با حذف تراکنش‌های از پیش تعیین‌شده باعث پنهان‌سازی عناصر حساس می‌شوند. الگوریتم sGA2DT از روش الگوریتم ژنتیک ساده با استفاده از تابع ارزیابی، مناسب‌ترین تراکنش‌ها را جهت حذف شدن انتخاب می کند برای این منظور باید هر بار کل پایگاه داده اسکن شود اما الگوریتم pGA2DT علاوه بر مفهوم GA ساده از مفهوم عناصر Large و Pre-Large برای محاسبه آسان تابع ارزیابی، بدون اسکن مجدد پایگاه داده اصلی استفاده می‌کند که سبب کاهش زمان اجرا و کاهش هزینه‌ی اسکن مجدد پایگاه داده خواهد شد. الگوریتم استخراج عناصر Pre-Large با توجه به اینکه در الگوریتم پیشنهادی نیز مورد استفاده قرار می‌گیرد، در پیوست مقاله آورده شده است.

در مرجع [۲۶]، برای حل محدودیت‌های الگوریتم‌های سنتی مبتنی بر GA با الزام حافظه بالا و محاسبات پیچیده در هر فرآیند تکاملی، مکانیسم الگوریتم ژنتیک فشرده(cGA[۱۱]) و مفهوم عناصر Pre-Large را در الگوریتم cpGA2DT[۱۲] پیشنهاد دادند. این الگوریتم از بردار احتمال برای هر تراکنش استفاده می‌کند. بر اساس مکانیسم cGA در هر تکرار، تنها دو کروموزوم با یکدیگر به رقابت می‌پردازند در این صورت احتمال انتخاب تراکنش‌های موجود در کروموزوم برنده، افزایش می‌یابد و از طرفی احتمال انتخاب تراکنش‌های کروموزوم بازنده، کاهش خواهد یافت.

در مرجع [۲۷]، مسئله پنهان‌سازی قوانین انجمنی حساس از طریق فرایند بهینه‌سازی چندهدفه تکاملی[۱۳] ارائه‌شده است. روش پنهان‌سازی پیشنهادی به‌عنوان EMO-RH-DT شناخته‌شده است که در این روش،کروموزوم‌ها شامل شناسه مربوط به مجموعه‌ای از تراکنش‌های انتخابی برای حذف شدن هستند و الگوریتم EMO زیرمجموعه بهینه از تراکنش‌ها را جهت حذف شدن انتخاب می‌کند به‌طوری‌که عوارض جانبی را به حداقل برساند.

در مرجع [۲۸]، الگوریتمی بر پایه ژنتیک معرفی‌شده است. این روش دارای یک متد برای ارزیابی برازندگی می‌باشد که در آن از دو عامل، قوانین گم‌شده و مجموعه عناصر گم‌شده در مجموعه داده، برای ارزیابی راه‌حل‌های ارائه‌شده استفاده می‌شود. به‌این‌ترتیب می‌تواند به یک‌راه حل بهینه یا نزدیک به راه‌حل بهینه دست یابد. در این الگوریتم قبل از اقدام برای مخفی کردن مجموعه عناصر حساس می‌داند برای چند بار باید عنصر قربانی را در تراکنش‌های موردنظر حذف کند. این الگوریتم همچنین از همپوشانی مجموعه عناصر حساس برای کاهش زمان و اثرات پنهان‌سازی استفاده می‌کند.

در مرجع [۲۹]، یک روش جدید و کارآمد با نام الگوریتم COA4ARH[۱۴] معرفی‌شده است که از الگوریتم بهینه‌سازی فاخته برای پنهان‌سازی قوانین انجمنی حساس بهره می‌برد. در این روش عمل پنهان‌سازی با استفاده از روش اعوجاج انجام می‌شود. علاوه بر این در این مطالعه سه تابع برازندگی تعریف‌شده است که امکان رسیدن به یک‌راه حل با کمترین عوارض جانبی را ممکن می‌سازد. معرفی یک تابع مهاجرتی کارآمد در این روش قابلیت فرار از بهینه‌های محلی را فراهم ساخته است.

در مرجع [۳۰]، الگوریتم PSO2DT[۱۵]، مبتنی بر بهینه‌سازی ازدحام ذرات برای پنهان کردن مجموعه عناصر حساس با حداقل عوارض جانبی فرایند ایمن‌سازی توسعه داده‌شده است. هر ذره‌ای در الگوریتم

پیشنهادی نشان‌دهنده مجموعه‌ای از تراکنش‌ها جهت حذف می‌باشد. ذرات با استفاده از یک تابع برازندگی برای به حداقل رساندن عوارض جانبی ایمن‌سازی ارزیابی می‌شوند. الگوریتم پیشنهادی همچنین می‌تواند حداکثر تعداد تراکنش‌ها جهت حذف شدن را به‌منظور پنهان‌سازی مؤثر مجموعه عناصر حساس را تعیین کند که برخلاف روش رویکرد مبتنی بر GA است. مزیت دیگر روش پیشنهادی در مقابل GA داشتن پارامترهای کمتر برای تنظیم کردن می‌باشد. علاوه بر این مفهوم pre-large جهت افزایش سرعت تکامل بکار برده شده است.

در مرجع[۳۱]، روی حفظ حریم خصوصی در کاوش قواعد انجمنی تمرکز شده است. ازآنجایی‌که تعادل در عوارض جانبی مختلف وجود دارد، تلاش می‌شود تا آن‌ها را با بهینه‌سازی چندهدفه به حداقل رساند. در الگوریتم EMO-RH[۱۶] پنهان‌سازی قواعد حساس از طریق حذف عناصر شناخته‌شده انجام می‌گیرد. عوارض جانبی روی قوانین غیر حساس گم‌شده، قوانین مصنوعی و قوانین ازدست‌رفته به‌عنوان اهداف بهینه‌سازی فرموله می‌شوند. بهینه‌سازی چندهدفه برای پیدا کردن یک زیرمجموعه مناسب از تراکنش‌ها به‌طوری‌که عوارض جانبی را کاهش دهد، استفاده‌شده است. به دلیل زیاد بودن الگوریتم‌ها در بخش کارهای مرتبط با پنهان‌سازی قوانین انجمنی، آن‌ها بر اساس فاکتورهای مختلف در جدول ۱ مقایسه و طبقه‌بندی‌شده‌اند.

**جدول ۱: مقایسه و طبقه‌بندی الگوریتم‌های پنهان‌سازی قوانین انجمنی**

| رویکرد پنهان‌سازی | نام الگوریتم | پنهان‌سازی عناصر پرتکرار | پنهان‌سازی قواعد انجمنی | حذف عناصر | درج عناصر | حذف تراکنش |
|---|---|---|---|---|---|---|
| اکتشافی | Algo1.a [11] | | √ | | √ | |
| | Algo1.b, 2.a [11] | | √ | √ | | |
| | DSRRC [13] | | √ | √ | | |
| | ADSRRC [14] | | √ | √ | | |
| | RRLR [14] | | √ | √ | √ | |
| | MDSRRC [15] | | √ | √ | | |
| | Algo2b, 2c [11] | √ | | √ | | |
| | Aggregate [12] | √ | | | | √ |
| | Disaggregate [12] | √ | | √ | | |
| | Greedy [32] | √ | | | √ | |
| | Hybrid [19] | | √ | √ | | |
| | Relevance-Sorting [18] | | √ | √ | | |
| مبتنی بر مرز | BBA [20] | √ | | √ | | |
| | Max-Min [21] | √ | | √ | | |
| سایر رویکردها (تکاملی) | Genetic [24] | | √ | √ | | |
| | sGA2DT [25] | √ | | | | √ |
| | pGA2DT [25] | √ | | | | √ |
| | cpGA2DT [26] | √ | | | | √ |
| | EMO-RH-DT [27] | √ | | | | √ |
| | COA4ARH [29] | | √ | √ | | |
| | PSO2DT [30] | √ | | | | √ |
| | EMO-RH [31] | | √ | √ | | |

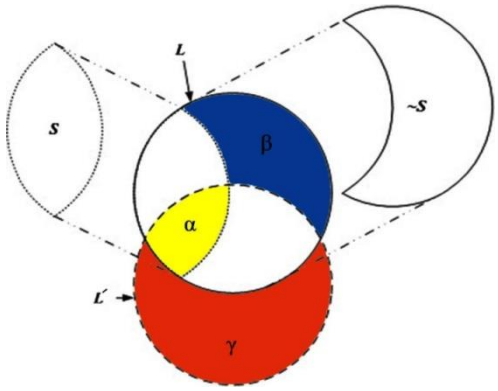
## ۳- الگوریتم پیشنهادی پنهان‌سازی مجموعه عناصر حساس با حذف تراکنش‌های حساس مرتب‌سازی شده

قبل از شروع فرایند ایمن‌سازی برای مخفی کردن مجموعه عناصر حساس، مجموعه عناصر پرتکرار را می‌توان از طریق تکنیک‌های مختلف داده‌کاوی به دست آورد[۳۳، ۳۴]. مجموعه $I=\{i_1,i_2,i_3,\ldots,i_n\}$ مجموعه عناصر در پایگاه داده اصلی D را نشان می‌دهد که $i_j$ عنصر j ام می‌باشد و پایگاه داده اصلی D، $D=\{t_1,t_2,t_3,\ldots,t_d\}$ شامل چندین تراکنش می‌باشد و $t_j$ تراکنش j ام را نشان می‌دهد. هر تراکنش از زیرمجموعه‌ای از عناصر I تشکیل‌شده است. تعداد عناصر را n و تعداد تراکنش‌ها d در نظر گرفته می‌شود.

اگر $\sup(i_j)$ به معنای پشتیبانی از عنصر $i_j$ ام (تعداد تکرار عنصر $i_j$ام در پایگاه داده‌D) و ε حداقل آستانه پشتیبانی در پایگاه داده باشد. آنگاه یک عنصر به‌عنوان عنصر پرتکرار[۱۷] یا $freq(i_j)$ تعریف می‌شود اگر شرایط رابطه (۱) برقرار باشد[۲۵]:

$$freq(i_j)=\frac{\sup(i_j)}{|D|}\geq\varepsilon \qquad (۱)$$

هدف PPDM، مخفی کردن مجموعه عناصر حساس با کمترین عوارض جانبی است. ارتباط عناصر حساس بعد و قبل از فرایند ایمن‌سازی را می‌توان در شکل ۱ مشاهده کرد که L نشان‌دهنده مجموعه عناصر بزرگ[۱۸] از D است وS نشان‌دهنده مجموعه عناصر حساس تعریف‌شده توسط کاربر است که بزرگ هم هستند؛ و ~S نشان‌دهنده مجموعه عناصر غیر حساس که بزرگ هم هستند؛ و ′L مجموعه عناصر بزرگ بعد از حذف تراکنش است.




**شکل ۱: ارتباط بین عوارض جانبی و عناصر حساس قبل و بعد از ایمن‌سازی[۲۵]**

تعاریف زیر را در نظر بگیرید:

تعریف ۱) فرض کنید مجموعه HS شامل تعدادی از عناصر حساس جهت پنهان شدن است و $HS=\{Si_1,Si_2,Si_3,\ldots,Si_k\}$ است. در الگوریتم پیشنهادی برای پنهان‌سازی عناصر حساس از طریق حذف تراکنش‌ها، تعداد پشتیبانی از عناصر حساس باید کمتر از حداقل آستانه پشتیبانی باشد. به‌طوری‌که هر تراکنش حذف‌شده باید شامل حداقل یک عنصر حساس باشد.

تعریف ۲) فرض کنید پایگاه داده اصلی $D=\{t_1,t_2,\ldots,t_d\}$ است و $D'$ پایگاه داده فرعی است که از D ایجادشده است. به‌طوری‌که هر تراکنش $t_j$ در $D'$ باید شامل هر عنصر حساس در HS باشد.

تعریف ۳) m تعداد تراکنش‌های حذف‌شده است که می‌تواند با جمع تعداد تفاوت بین مجموعه عناصر حساس و تعداد حداقل پشتیبانی محاسبه شود.

$$m = \sum_{j=1}^{k} freq(si_j) - (\varepsilon^* \mid D \mid) + 1 \qquad (۲)$$

تعریف ۴) در الگوریتم ژنتیک به‌کاررفته درروش پیشنهادی کروموزوم یک آرایه به طول $m$ از جنس صحیح می‌باشد. که کروموزوم $i$ام، $C_i$، مجموعه‌ای از $m$ ژن است به‌طوری‌که در آن هر ژن به‌عنوان یک تراکنش $t_j$ نشان داده‌شده است و $t_j \in D' \text{ or Null}$. درواقع مقدار هر ژن معادل تراکنشی است که بایستی حذف گردد. لازم به ذکر است که مقادیر ژن‌ها با یکدیگر باید متفاوت باشد. اندازه کروموزوم در فرایند الگوریتم ژنتیک ثابت می‌باشد. از رابطه (۲) نیز برای تعیین مقدار m مناسب در ابتدای الگوریتم استفاده می‌شود. در ضمن نحوه مقداردهی اولیه به کروموزوم دربند ۵ شبه کد توضیح داده خواهد شد.

درروش مطرح‌شده توسط هانگ و همکاران [۲۵]، برای پنهان‌سازی مجموعه عناصر حساس از الگوریتم ژنتیک معمولی استفاده‌شده است؛ که در این روش برای محاسبه تابع برازندگی، چند تابع را با یکدیگر جمع نموده و حاصل جمع را کمینه می‌کند درصورتی‌که لزوماً تک‌تک توابع کمینه نخواهند شد. ازاین‌رو الگوریتم پیشنهادی در این پژوهش، با به‌کارگیری الگوریتم‌های تکاملی چندهدفه، این ضعف را برطرف نموده و هم‌زمان تمامی توابع برازندگی کمینه خواهند شد. همچنین برای کاهش خطای مربوط به پنهان‌سازی و اثرات ناشی از آن، تراکنش‌هایی با کمترین طول و یا تراکنش‌هایی با بیشترین پشتیبانی از عناصر حساس و کمترین عناصر غیرحساس، به‌عنوان درصدی از جمعیت اولیه انتخاب می‌شوند که این امر موجب تسریع بخشی در اجرای الگوریتم می‌شود و عملکرد روش را بهبود می‌بخشد.

الگوریتم پیشنهادی با استفاده از رویکرد مبتنی بر پشتیبان، سعی در کاهش پشتیبانی مجموعه عناصر حساس موجود در پایگاه داده تراکنشی دارد. برای کاهش پشتیبانی مجموعه عناصر حساس،تراکنش‌هایی را که شامل عناصر حساس هستند را حذف می‌کند تا زمانی که پشتیبانی عنصر حساس به کمتر از حداقل آستانه پشتیبانی برسد. برای این منظور ابتدا یک عمل پیش‌پردازش بر روی تراکنش‌های موجود بر روی پایگاه داده اصلی صورت می‌گیرد. فرایند پیش ایمن‌سازی پایگاه داده اصلی[۱۹] (DPSP) اولین گام از روش پیشنهادی است که شامل پیش‌پردازش پایگاه داده اصلی است. به دلیل اینکه عناصر حساس محدود به برخی از تراکنش‌ها می‌باشند نیازی به تغییر همه تراکنش‌ها در الگوریتم وجود ندارد؛ بنابراین الگوریتم در مرحله پیش‌پردازش همه تراکنش‌هایی که از عنصر حساس پشتیبانی می‌کنند را انتخاب می‌کند. با استفاده از این مرحله‌ی الگوریتم، می‌توان به بهترین کارایی ازنظر سرعت ایمن‌سازی و کمترین تعداد تغییرات موردنیاز در فرایند پنهان‌سازی دست‌یافت.

سپس با تعیین تعداد تراکنش‌های مناسب جهت حذف شدن هر عنصر حساس و انتخاب شناسه تراکنش‌های مناسب برای هر عنصر حساس به‌صورت جداگانه، ژن‌های موجود در هر کروموزوم را تعیین می‌کنیم. الگوریتم در هر تکرار، تنها با یک‌بار پویش تراکنش‌های حساس به‌جای پویش کل تراکنش‌های پایگاه داده، باعث افزایش سرعت و کاهش هزینه‌های اجرا خواهد شد. از طرفی برای اینکه عوارض ناشی از حذف تراکنش‌های حساس به حداقل ممکن برسد تعدادی از کروموزوم‌های جمعیت اولیه بر اساس دو معیار مرتب‌سازی می‌شوند. به‌این‌ترتیب به‌منظور بهبود انتخاب اعضای جمعیت، دو راه‌حل اضافه‌شده است: اولاً هر عنصر حساس، شناسه تراکنش‌هایی را به‌عنوان ژن خود انتخاب می‌کند که دارای کمترین طول پشتیبانی باشند؛ یعنی برای هر عنصر حساس، تراکنش‌هایی که آن عنصر را پشتیبانی می‌کنند را بر اساس کمترین طول مرتب می‌کند تا عوارض جانبی کمتری به وجود آورد. ثانیاً هر عنصر حساس، شناسه تراکنشی را به‌عنوان ژن قبول می‌کند که دارای بیشترین تعداد پشتیبانی از عناصر حساس است و درعین‌حال کمترین

عنصر غیر حساس را شامل می‌شود. بعد از انتخاب جمعیت اولیه، تابع برازندگی به‌جای استفاده از بهینه‌سازی تک هدفه از بهینه‌سازی چندهدفه استفاده می‌کند و با استفاده از الگوریتم ژنتیک مبتنی بر مرتب‌سازی نخبه‌گرای نامغلوب (NSGA-II) جمعیت نسل بعد تولید می‌شود تا زمانی که مقدار هر سه فاکتور تابع برازندگی برابر صفر شود یا تعداد دفعات تکرار الگوریتم به مقدار مشخص برسد.

در الگوریتم پیشنهادی که حذف تراکنش‌های حساس با الگوریتم ژنتیک چندهدفه صورت می‌پذیرد، هدف مینیمم‌سازی چندین تابع (معیار) می‌باشد. ازاین‌رو این توابع که در اصل توابع برازندگی الگوریتم ژنتیک چندهدفه می‌باشند به فرم زیر در نظر گرفته‌شده‌اند، این توابع به‌طور هم‌زمان در فرایند الگوریتم ژنتیک چندهدفه بهبود می‌یابند:

تابع برازندگی شکست پنهان‌سازی[۲۰] مجموعه عناصر حساس، در PPDM با عنوان $\alpha$ تعریف شده است. $\alpha$ مشخص‌کننده عناصر حساسی هستند که بعد از عمل ایمن‌سازی شکست‌خورده‌اند و پنهان نشده‌اند و باید در حالت ایده آل صفر باشند که به‌صورت رابطه (۳) تعریف می‌شود:

$$\alpha = S \cap L' \qquad (۳)$$

تابع برازندگی مجموعه عناصر گم‌شده[۲۱] در PPDM با علامت $\beta$ نمایش داده می‌شود. مجموعه عناصر گم‌شده، عناصر بزرگ غیر حساس در پایگاه داده اصلی است؛ اما از پایگاه داده ایمن‌سازی شده *D قابل‌استخراج نیستند. به‌طوری‌که $\beta$ به‌صورت رابطه (۴) تعریف می‌شود:

$$\beta = \sim S - L' = (L - S) - L' \qquad (۴)$$

تابع برازندگی عناصر مصنوعی[۲۲] در PPDM به‌عنوان $\gamma$ تعریف می‌شوند. L نشان‌دهنده مجموعه عناصر بزرگ در پایگاه داده ایمن‌سازی شده *Dاست اما در پایگاه داده اصلی متعلق به عناصر بزرگ نیست. به‌طوری‌که $\gamma$ به‌صورت رابطه (۵) تعریف می‌شود:

$$\gamma = L' - L \qquad (۵)$$

بنابراین $\alpha$، $\beta$ و $\gamma$ به عنوان سه تابع برازندگی خواهند بود. شبه کد الگوریتم پیشنهادی آورده شده است:

۱. مشخص کردن عناصر حساس و حساسیت آن‌ها
۲. اعمال تکنیک داده‌کاوی برای استخراج مجموعه عناصر پرتکرار با حداقل آستانه پشتیبانی توسط الگوریتم apriori.
۳. انجام عمل پیش‌پردازش بر روی کل پایگاه داده اصلی جهت انتخاب تراکنش‌های حساس و درج آن‌ها در پایگاه داده فرعی D´.
۴. محاسبه تعداد تراکنش‌های مناسب جهت حذف شدن (m) در هر بار عمل پویش از پایگاه داده اصلی.
۵. ابتدا جمعیت اولیه P0 به تعداد N کروموزوم به فرم آرایه‌ای صحیح به طول m ایجاد می‌شود. (از شمارنده t برای تمایز بین نسل‌ها استفاده می‌شود، در مرحله اول t=۰ است.) مرحله انتخاب تراکنش‌های حساس از پایگاه داده فرعی D´ و ایجاد کروموزوم‌های نسل اولیه شامل سه قسمت است:

- انتخاب تراکنش‌های حساس بر اساس کمترین طول پشتیبانی به‌عنوان تعدادی از کروموزوم‌ها.
- انتخاب تراکنش‌های حساس بر اساس بیشترین تعداد پشتیبانی از عناصر حساس و کمترین تعداد عناصر غیر حساس به‌عنوان تعدادی از کروموزوم‌ها.
- انتخاب تراکنش‌های حساس به‌صورت تصادفی از بین تراکنش‌های پایگاه داده فرعی D´ برای هر عنصر حساس به‌صورت مجزا، برای مابقی کروموزوم‌ها.

۶. سپس هر کروموزوم، توسط تابع برازندگی چندهدفه که از قبل طراحی‌شده است برای محاسبه عوارض جانبی ناشی از پنهان‌سازی، موردبررسی و ارزیابی قرار می‌گیرد.
۷. اعمال عملگر ترکیب، عملگری که در این روش به‌عنوان عملگر ترکیب بکار رفته است عملگر یک نقطه‌ای است. در این روش دو کروموزوم به عنوان والد انتخاب‌شده و با توجه به احتمال اعمال عملگر ترکیب، pc[۲۳]، ممکن است روی این دو کروموزوم عملگر جهش اعمال گردد. سپس یک موقعیت تصادفی بین دو کروموزوم در نظر گرفته می‌شود و تمامی ژن‌های طرف راست یا طرف چپ این موقعیت در کروموزوم‌های والد با یکدیگر جابجا می‌شوند تا کروموزوم‌های جدید به دست آیند. سپس بایستی یکسان نبودن ژن‌ها در هر یک از کروموزوم‌های فرزند بررسی گردد. در صورت مشاهده شدن ژن یکسان در هر دو کروموزوم فرزند جای دو ژن در دو کروموزوم فرزند جابجا می‌شود. ممکن است حالتی وجود داشته باشد که فقط در یکی از کروموزوم‌ها ژن‌های یکسان وجود دارد. در این حالت ژن تکرار شده با اولین ژنی از کروموزوم فرزند دوم که در کروموزوم فرزند اول نیست جابجا می‌گردد. این کار تا زمانی ادامه می‌یابد که تمامی ژن‌های هر دو کروموزوم فرزند متفاوت شود.

اگر P1 و P2 دو کروموزوم والد باشند و روی آن‌ها عملگر ترکیب یک نقطه‌ای اعمال گردد، آنگاه دو کروموزوم فرزند C1 و C2 ایجاد می‌گردد. اگر نقطه شکست k باشد ساختار کروموزوم‌های فرزند به‌صورت زیر خواهد بود [۳۵].

C1 : P1[1], P1[2], …,P1[k],P2[k+1],…,P2[m]
C2 : P2[1], P2[2], …,P2[k],P1[k+1],…,P1[m]

که Pi[j] معادل ژن jام از کروموزوم iام می‌باشد. همان‌طور که مشاهده می‌گردد طول کروموزوم‌های فرزند برابر m خواهد بود. اگر $P_1[i_1]$ با $P_1[i_2]$ از کروموزوم اول یکسان باشند و $P_2[j_2]$ با $P_2[j_1]$ از کروموزوم دوم یکسان

باشند آنگاه جای دو کروموزوم $P_1[i_1]$ و $P_2[j_1]$ عوض می‌گردد. تا این قسمت شبیه به روش عملگر ترکیب PMX[۲۴] [۳۶] می‌باشد. این کار تا زمانی که حداقل یکی از کروموزم‌ها ساختار درستی (ژن‌ها یکسان نباشند) بگیرد ادامه می‌یابد. در ادامه بایستی کروموزومی که ساختار درستی ندارد اصلاح شود. اگر $P_1[i_1]$ با $P_1[i_2]$ از کروموزوم اول یکسان باشند آنگاه با $P_2[j_1]$ از کروموزوم دوم به شرط $\forall i; P_2[j_1] \neq P_1[i]$ با $P_1[i_1]$ جابجا می‌گردد. این مرحله نیز تا زمانی که کروموزم باقیمانده ساختار درستی بگیرد ادامه می‌یابد.

۸. اعمال عملگر جهش، عملگر جهش نقش بسیار مهمی در فرار از بهینه‌های محلی و افزایش سرعت همگرایی الگوریتم ژنتیک ایفا می‌کند. در الگوریتم پیشنهادی، عملگرهای جهش برای ایجاد گوناگونی ارائه‌شده‌اند. بعد از اعمال عملگر ترکیب با درنظر گرفتن احتمال اعمال عملگر جهش[۲۵]، بر روی برخی از کروموزوم‌ها عملگر جهش اعمال می‌گردد. در روش پیشنهادی از عملگر جهش استفاده شده است: الف) عملگر جهش جابجایی[۲۶] که دو ژن تصادفی از یک کروموزوم را جابجا می‌کند [۳۶]. ب) عملگر جهش مقداردهی مجدد یک ژن[۲۷] که در آن یک ژن به تصادف انتخاب شده و به آن مقداردهی مجدد می‌شود به قسمی که مقدار آن در کروموزوم نباشد. ج) مقداردهی مجدد تمام ژن‌های یک کروموزوم با شرط یکسان نبودن هیچ دو ژنی. هر یک از این عملگرها مزایایی دارند که در عمل در صورت اعمال عملگر جهش یکی از این سه عملگر به کار گرفته می‌شود.

با اعمال عملگرهای ادغام و جهش، جمعیت فرزند Q با اندازه N تولید می‌شود.

۹. در مسائل بهینه‌ساز چندهدفه بحث بهینه‌سازی همزمان چند تابع و غلبه[۲۸] حائز اهمیت است. برای این منظور روش‌های متعددی از قبیل روش وزن‌دهی، روش قیود محدود[۲۹]، روش میل به مقصود[۳۰]، روش تقاطع مرزی متعامد[۳۱] و مجموعه پارتو مطرح شده است [۳۷]. با توجه به اینکه در روش پیشنهادی از الگوریتم NSGA-II بهره گرفته شده است، در این الگوریتم از رتبه‌بندی نامغلوب[۳۲] استفاده می‌کند و سپس فاصله ازدحام[۳۳] جمعیت محاسبه محاسبه می‌گردد. از این دو پارامتر در عملگر انتخاب[۳۴] که مبتنی بر انتخاب تورنمنت[۳۵] می‌باشد استفاده می‌گردد [۳۸]. اگر شرایط اتمام الگوریتم برقرار باشد یعنی مقدار هر سه فاکتور تابع برازندگی، برابر صفر شود یا تعداد دفعات تکرار الگوریتم، به مقدار مشخص برسد، بهترین تراکنش‌ها جهت حذف از پایگاه داده اصلی مشخص‌شده و الگوریتم پایان می‌یابد، در غیر این صورت مرحله اعمال عملگر انتخاب برای ایجاد جمعیت بعدی با استفاده از الگوریتم NSGA-II [۳۸] انجام می‌شود.

مراحل اجرای الگوریتم NSGA-II به‌صورت زیر است:

۱۰. جمعیت Rt با اندازه 2N از اجتماع دو جمیعت Qt , Pt ایجاد می‌شود. با استفاده از روش مرتب‌سازی غلبه نشده، تمامی اعضای جمعیت، به ترتیب بر اساس سه فاکتور شکست پنهان‌سازی، عناصر گم‌شده و عناصر مصنوعی به‌صورت نزولی مرتب‌شده و رتبه‌بندی می‌شوند و در جبهه‌های F1,F2,…,Fk قرار می‌گیرند.

۱۱. برای اعضای همه جبهه‌ها مقادیر فواصل ازدحام محاسبه می‌شود.

۱۲. جمعیت والدین Pt+۱ بدین‌صورت ایجاد می‌شود که با شروع از اعضای جبهه F1 درصورتی‌که $|Pt+1|+|Fi|<=N$، اعضای جبهه Fi به جمعیت Pt+1 اضافه می‌شوند. درصورتی‌که $|Pt+1|+|Fi|>N$، اعضایی از Fi که فاصله ازدحام بیشتری داشته باشند به Pt+۱ اضافه می‌شوند تا اندازه جمعیت Pt+۱ برابر N شود.

۱۳. پس از تشکیل جمعیت Pt+۱ با استفاده از روش انتخاب مسابقه‌ای، مقایسه رتبه بین اعضای هر مسابقه انجام می‌شود. عضوی که رتبه بهتری (پایین‌تر) داشته باشد، انتخاب می‌شود. درحالی‌که رتبه اعضای مسابقه یکسان باشد، عضوی که فاصله ازدحام آن بیشتر باشد، انتخاب می‌شود. سپس عملگرهای ادغام و جهش برای تولید جمعیت فرزندان Qt+۱ با اندازه N به کار می‌رود.

۱۴. t=t+۱ و به مرحله ۶ می‌رود، این چرخه تا رسیدن به شرایط توقف الگوریتم ادامه می‌یابد.

در شکل (۲) شبه کد روش پیشنهادی با توجه به ۱۴ مرحله بالا نشان داده‌شده است.

**Input**: Original Database D, a set of sensitive itemsets HS, a number of m //transaction to be deleted, a population size N. Pt population of Parent and Qt population of children , and a upper Support Threshold $S_u$ and a lower support threshold $S_l$.
**Output**: A Sanitized Database $D^*$
Termination Condition: The fitness=0 or number of Generation=N.
1. Set $S_l=S_u*(1-m/|D|)$
2. Scan D to get L and PL respectively by $S_u$ and $S_l$.
3. For j=1 :n ; a=1:k do
4. If $Si_a \in T_j$ then
5. Project $T_j$ from D forming D´.
6. End if
7. End for
// Generate initial population
// 10 chromosome with m genes Have been Generated from the shortest Supporting transaction and from the transaction that support the most sensitive rules but support the least non-sensitive rules from D´ into p.

8. For i=1:10 do
9. Ci[a]={$T_j$ or 0 , $T_j \in D'$ , $1 \le a \le m$}
10.End for
// p-10 chromosome with m genes Have been Generated randomly from D′.
11.For i=p-10:p do
12. For t=1:numel(HS)
13. Ci[a][t]={$T_j$ or 0,$T_j \in D'$,$1 \le a \le m$}
14. End for
15.End for
// Fitness assignment and sorting descending order.
16. Qt=Evaluate Fitness($C_i$)
17. Crossover
18. Mutation
19. If terminated condition is not satisfied then
20. Rt=($Pt \cup Qt$)
21. Fast-nondominated sort(Rt)
22. density estimation(Rt)
23. Selection Pt+1:Reduce Pt+1 from (Rt/2)
24. Go to line 16
25. Else
26. Terminate
27. End if

**شکل ۲: شبه کد الگوریتم پیشنهادی**

با توجه به مطالب گفته‌شده روش پیشنهادی نسبت به روش‌های مشابه و متداول با در نظر گرفتن معیارهای چندهدفه نسبت به یک هدفه عملکرد بهتری خواهد داشت. همچنین با یک‌بار پیمایش عناصر حساس در ابتدای الگوریتم باعث افزایش سرعت الگوریتم خواهد شد که در ادامه در آزمایش‌های این مسئله نشان داده خواهد شد.

## ۴- مثالی از الگوریتم ارائه‌شده

در ادامه با ارائه یک مثال الگوریتم پیشنهادی تشریح می‌شود. در جدول۲، پایگاه داده تراکنشی نشان داده‌شده است.

**جدول ۲: پایگاه داده تراکنشی**

| TID | ITEMS |
|---|---|
| ۱ | A,B,C |
| ۲ | B,C,E |
| ۳ | A,B,C,E |
| ۴ | A,B,E |
| ۵ | A,B,E,G |
| ۶ | A,C,D |
| ۷ | B,C,D,E,G |
| ۸ | B,C,E,F |
| ۹ | C,F,G |
| ۱۰ | A,B,G |

پایگاه داده تراکنشی شامل ۱۰ تراکنش است. همچنین تعدادی از عناصر پرتکرار به‌عنوان عناصر حساس {BE,BCE} برای پنهان‌سازی تعریف‌شده‌اند و آستانه پشتیبانی بالا ۴۰٪ و آستانه پشتیبانی پایین ۲۰٪ (Su=0.4, Sl=0.2) است. مجموعه عناصر پرتکرار Large با پشتیبانی بزرگ‌تر و مساوی با ۴۰٪ در جدول ۳ نشان داده‌شده است. همچنین مجموعه عناصر پرتکرار pre-large با پشتیبانی بزرگ‌تر و مساوی با ۲۰٪ در جدول ۴ نشان داده‌شده است.

**جدول ۳: مجموعه عناصر پرتکرارLarge**

| 1-itemset | count | 2-itemset | count | 3-itemset | Count |
|---|---|---|---|---|---|
| A | 6 | AB | 5 | BCE | 4 |
| B | 8 | BC | 5 | | |
| C | 7 | BE | 6 | | |
| E | 6 | CE | 4 | | |
| G | 4 | | | | |

**جدول ۴: مجموعه عناصر پرتکرار Pre-large**

| 1-itemset | count | 2-itemset | count | 3-itemset | Count |
|---|---|---|---|---|---|
| D | 2 | AC | 3 | ABC | 2 |
| F | 2 | AE | 3 | ABE | 3 |
| | | AG | 2 | ABG | 2 |
| | | BG | 3 | BEG | 2 |
| | | CD | 2 | | |
| | | CG | 2 | | |
| | | EG | 2 | | |
| | | CF | 2 | | |

با توجه به عناصر حساس که توسط کاربر تعیین‌شده است، از بین ۱۰ تراکنش پایگاه داده اصلی، ۶ تراکنش با شناسه ۲ و ۳ و ۴ و ۵ و ۷ و ۸ به‌عنوان تراکنش‌های حساس در پایگاه داده فرعی D′ ذخیره می‌شوند. میزان حساسیت عنصر حساس BE مساوی ۶ و حساسیت عنصر حساس BCE مساوی ۴ است.

حال برای اینکه عوارض جانبی ناشی از پنهان‌سازی کمتر شود دو راه‌حل برای انتخاب تراکنش‌ها در نظر گرفته‌شده است. در راه‌حل اول ابتدا تراکنش‌های حساس بر اساس کمترین طول مرتب می‌شوند. در جدول ۵ تراکنش‌های حساس که بر اساس کمترین طول مرتب‌شده‌اند نمایش داده می‌شوند.

**جدول ۵: مرتب‌سازی تراکنش‌های حساس بر اساس کمترین طول**

| TID | ITEMS | Length |
|---|---|---|
| ۲ | B,C,E | 3 |
| ۴ | A,B,E | 3 |
| ۳ | A,B,C,E | 4 |
| ۵ | A,B,E,G | 4 |
| ۸ | B,C,E,F | 4 |
| ۷ | B,C,D,E,G | 5 |

و در راه‌حل دوم تراکنش‌های حساس بر اساس کمترین عناصر غیر حساس به‌صورت صعودی و بیشترین تعداد عناصر حساس به‌صورت نزولی مرتب می‌شوند. تراکنش‌های مرتب‌شده در جدول ۶ قابل مشاهده هستند.

**جدول ۶: مرتب‌سازی تراکنش‌های حساس بر اساس کمترین عنصر غیرحساس و بیشترین عنصر حساس**

| TID | ITEMS | Non-Sensitivity | Sensitivity |
|---|---|---|---|
| ۲ | B,C,E | 0 | 2 |
| ۳ | A,B,C,E | 1 | 2 |

| | | | |
|---|---|---|---|
| ۸ | B,C,E,F | 1 | 2 |
| ۷ | B,C,D,E,G | 2 | 2 |
| ۴ | A,B,E | 1 | 1 |
| ۵ | A,B,E,G | 2 | 1 |

در این مرحله باید تعداد تراکنش‌های مناسب برای هر عنصر حساس (m)، جهت حذف شدن را مشخص کنیم.

با استفاده از رابطه ۲ مقدار m را تعیین می‌کنیم. با توجه به فرمول m تعداد تراکنش مناسب برای پنهان‌سازی عنصر be، $m_1=3$ است و تعداد تراکنش مناسب برای پنهان‌سازی عنصر bce، $m_2=1$ است که درمجموع $m=m_1+m_2$، برابر ۴ می‌باشد.

اکنون باید به تعداد جمعیت اولیه یعنی p کروموزوم (p=40) با m ژن ایجاد کنیم که هر ژن در کروموزوم حاوی شناسه مربوط به تراکنش حساس است. با توجه به مقدار m باید از بین تراکنش‌های حساس تعدادی از تراکنش‌ها را جهت حذف شدن انتخاب کنیم. مرحله انتخاب تراکنش‌ها برای جمعیت اولیه، شامل سه مرحله است. مرحله اول ایجاد ۵ کروموزوم با انتخاب تراکنش‌هایی که دارای کمترین طول می‌باشند. مرحله دوم ایجاد ۵ کروموزوم با انتخاب تراکنش‌هایی که دارای کمترین عنصر غیر حساس و بیشترین عنصر حساس می‌باشند و در مرحله آخر ایجاد ۳۰ کروموزوم با انتخاب تراکنش‌ها به‌صورت تصادفی می‌باشد.

پس از ایجاد کروموزوم‌های جمعیت اولیه، اکنون با استفاده از یک تابع برازندگی چندهدفه باید هر یک از کروموزوم‌ها را موردبررسی و ارزیابی قرار دهیم که توابع برازندگی شامل سه فاکتور مجزا در روابط (۳)، (۴) و (۵) تعریف شده‌اند.

درصورتی‌که مقدار هر سه فاکتور توابع برازندگی برابر صفر شود یا تعداد دفعات تکرار الگوریتم به مقدار مشخص برسد الگوریتم پایان می‌یابد در غیر این صورت عمل ترکیب و جهش بر روی کروموزوم‌ها صورت می‌گیرد و با استفاده از الگوریتم NSGA-II جمعیت نسل بعدی تشکیل می‌شود و دوباره تابع برازندگی برای جمعیت نسل جدید بررسی می‌شود تا اینکه شرایط خاتمه الگوریتم برآورده شود. نتیجه حاصل از اجرای این الگوریتم به‌صورت زیراست:

شناسه‌ی تراکنش‌های مناسب جهت حذف شدن برابر ۲، ۴، ۵ و ۸ است؛ بنابراین پایگاه داده ایمن‌سازی شده بعد از فرایند پنهان‌سازی به‌صورت جدول ۷ می‌شود؛ و عوارض جانبی حاصل از پنهان‌سازی به‌صورت جدول ۸ است.

**جدول ۷: پایگاه داده ایمن‌سازی شده D***

| TID | ITEMS |
|---|---|
| ۱ | A,B,C |
| ۳ | A,B,C,E |
| ۶ | A,C,D |
| ۷ | B,C,D,E,G |
| ۹ | C,F,G |
| ۱۰ | A,B,G |

**جدول ۸: عوارض ناشی از فرایند پنهان‌سازی**

| Hiding failure | Missing cost | Artificial cost |
|---|---|---|
| ۰ | ۰/۳ | ۰/۰۱۴۲۹ |

## ۵- نتایج آزمایشات

در این بخش به بررسی چارچوب تحقیق و ارزیابی نتایج می‌پردازیم. ابتدا معیارهای ارزیابی پنهان‌سازی قواعد انجمنی را بیان می‌کنیم. سپس با فاکتورهای مختلف، ازجمله مقادیر متفاوت از تعداد تراکنش‌های حذف شونده و حداقل آستانه پشتیبانی به بررسی و ارزیابی الگوریتم پنهان‌سازی می‌پردازیم. تمامی آزمایشات در محیط matlab انجام‌شده و عناصر پرتکرار توسط الگوریتم apriori استخراج‌شده است. معیارهای ارزیابی و دادگان مورداستفاده در ادامه شرح داده خواهند شد.

### ۵-۱- معیارهای ارزیابی

معیارهای ارزیابی موردبررسی در این پژوهش، شامل تعداد شکست‌های پنهان‌سازی، هزینه عناصر گم‌شده، عناصر مصنوعی و عدم تشابه است که از طریق فرمول‌های ارائه‌شده در زیر قابل‌محاسبه هستند[۲۵].

شکست پنهان‌سازی (HF): این مقیاس، مقدار عناصر حساس که بعد از اعمال فرایند ایمن‌سازی در پایگاه داده باقی‌مانده‌اند را تعیین می‌کند. مقدار این پارامتر توسط رابطه (۶) محاسبه می‌شود. که این نسبت برابر است با تعداد عناصر حساس در پایگاه داده ایمن شده HS(D*) نسبت به تعداد عناصر حساس در پایگاه داده‌های اصلی HS(D).

$$HF = \frac{HS(D^*)}{HS(D)} \qquad (۶)$$

هزینه عناصر گم‌شده (MC): این مقیاس، مقدار عناصر غیر حساس که بعد از اعمال فرآیند ایمن‌سازی روی پایگاه داده‌ها،پنهان‌شده‌اند را تعیین می‌کند. مقدار این پارامتر توسط رابطه ۷ محاسبه می‌شود که برابر است با تفاوت بین تعداد عناصر پرتکرار در پایگاه داده اصلی FS(D) و عناصر پرتکرار در پایگاه داده ایمن شده FS(D*) نسبت به‌کل عناصر پرتکرار در پایگاه داده اصلی FS(D).

$$MC = \frac{|FS(D) - FS(D^*)|}{|FS(D)|} \qquad (۷)$$

هزینه عناصر مصنوعی (AC): این مقیاس، مقدار عناصر پرتکرار کشف‌شده که مصنوعی و غیرواقعی هستند را بعد از فرآیند ایمن‌سازی بر روی پایگاه داده تغییریافته، تعیین می‌کند. مقدار این پارامتر توسط رابطه ۸ محاسبه می‌شود. در اینجا، FS(D) عناصر پرتکرار در پایگاه داده اصلی و FS(D*) عناصر پرتکرار در پایگاه داده ایمن‌سازی شده را نشان می‌دهند.

$$AC = \frac{|FS(D^*)| - |FS(D^*) \cap FS(D)|}{|FS(D^*)|} \qquad (۸)$$

عدم تشابه[۳۶]: تفاوت بین پایگاه داده اصلی با پایگاه داده ایمن‌سازی شده یک معیار مهم برای ارزیابی عملکرد الگوریتم است که به میزان از دست

رفتن اطلاعات اشاره دارد. این معیار با رابطه ۹ محاسبه می‌شود. که در آن $D_{ij}$ عنصر mام از تراکنش nام در پایگاه داده اصلی است و $D^*_{ij}$ عنصر mام از تراکنش nام در پایگاه داده ایمن‌سازی شده است.

$$Dis = \frac{|\sum_{i=1}^{n}\sum_{j=1}^{m}(D_{ij} - D_{ij}^{*})|}{|\sum_{i=1}^{n}\sum_{j=1}^{m}D_{ij}|} \quad (۹)$$

## ۵-۲- دادگان

در این پژوهش از پایگاه داده واقعی BMSWebView-1 جهت پیاده‌سازی و تست الگوریتم پیشنهادی استفاده‌شده است. مشخصات مربوط به این پایگاه داده مورد استفاده در ارزیابی، در جدول ۹ نشان داده‌شده است.

**جدول ۹: مشخصات مربوط به پایگاه داده**

| نام پایگاه داده | تعداد تراکنش‌ها | تعداد عناصر | میانگین طول تراکنش |
|---|---|---|---|
| BMSWebView-1 | ۵۹۶۰۲ | ۴۹۷ | ۲،۵ |

پارامترهای پیاده‌سازی الگوریتم‌های مرجع و الگوریتم پیشنهادی در جدول ۱۰ بیان‌شده است.

**جدول ۱۰: پارامترهای پیاده‌سازی**

| تعداد دفعات تکرار الگوریتم | اندازه جمعیت | نرخ جهش | نرخ آمیزش |
|---|---|---|---|
| ۱۰۰ | ۴۰ | ۰،۱ | ۰،۹ |

## ۵-۳- طراحی آزمایش‌ها

الگوریتم پیشنهادی که opMOGA2DT[۳۷] نامیده می‌شود با حذف تراکنش‌های مناسب با رویکرد مبتنی بر پشتیبان و پنهان‌سازی عناصر حساس عمل ایمن‌سازی را انجام می‌دهد که الگوریتم پیشنهادی با الگوریتم‌های sGA2DT و pGA2DT [۲۵] در ادامه مورد مقایسه و بررسی قرارگرفته است.

برای مقایسه و ارزیابی الگوریتم پیشنهادی با الگوریتم‌های sGA2DT و pGA2DT، آزمایش‌های مختلفی با مقادیر متفاوت حداقل آستانه پشتیبانی، درصد عناصر پرتکرار و تعداد تراکنش‌های حذف‌شده، در هر بار پویش پایگاه داده بر روی پایگاه داده واقعی ذکرشده انجام شده است. بر روی پایگاه داده BMSWebView-1 آزمایشات زیر صورت گرفته است:

دسته اول آزمایشات Su=1.5% و FIsهای مختلف

۱- با FIs=5%, m=300 (تعداد عناصر حساس برابر 2 است.)

۲- با FIs=10%, m=700 (تعداد عناصر حساس برابر ۴ است.)

۳- با FIs=15%, m=900 (تعداد عناصر حساس برابر ۶ است.)

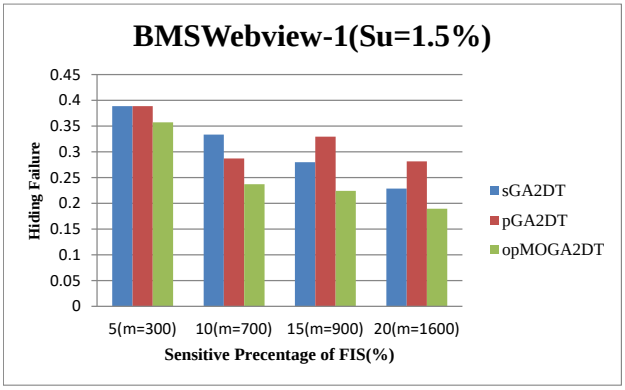
۴- با FIs=20%, m=1600 (تعداد عناصر حساس برابر ۸ است.)

در دسته اول آزمایشات مقدار حداقل آستانه پشتیبانی برابر با مقدار ثابت ۱/۵ درصد در نظر گرفته‌شده است و هر بار درصد عناصر پرتکرار و تعداد تراکنش جهت حذف شدن متغیر خواهد بود. نمودار مربوط به این دسته از آزمایشات برای نرخ شکست پنهان‌سازی در شکل ۳، نرخ عناصر ازدست‌رفته در شکل۴، نرخ عناصر جدید در شکل ۵ و نرخ عدم تشابه در شکل ۶ قابل‌ملاحظه می‌باشد.

برای محاسبه نرخ شکست پنهان‌سازی الگوریتم‌های مختلف، پس از اعمال هر الگوریتم تعداد عناصر حساس در پایگاه داده ایمن شده را شمارش کرده بر تعداد عناصر حساس در پایگاه داده اصلی تقسیم می‌نماییم.

شکل (۳) نرخ شکست پنهان‌سازی برای روش‌های مختلف را روی دادگان BMSWebView-1 با آستانه پشتیبان بالا ۱/۵ درصد نشان می‌دهد. محور افقی درصد عناصر حساسی که قرار است از m تراکنش حذف شوند را نشان می‌دهد. همان‌طور که مشاهده می‌شود روش پیشنهادی، opMOGA2DT، نسبت به دو روش دیگر دارای نرخ شکست پنهان‌سازی کمتری می‌باشد. این رفتار در تمامی حالات درصد عناصر حساس حذف‌شده مشاهده می‌گردد. دلیل عملکرد بهتر روش پیشنهادی نسبت به دیگر روش‌ها به دلیل به‌کارگیری بهینه‌ساز چندهدفه و نیز نوع توابع برازندگی تعریف‌شده مرتبط با ماهیت مسئله می‌باشد.

روش pGA2DT با استفاده از عناصر pre-large درروند الگوریتم خود باعث کاهش اسکن مجدد پایگاه داده اصلی شده که در مجموعه تراکنش‌های کم عملکرد قابل قبولی داشته و با افزایش اندازه تراکنش‌های کارایی آن کاهش می‌یابد که در شکل نیز قابل‌مشاهده می‌باشد.

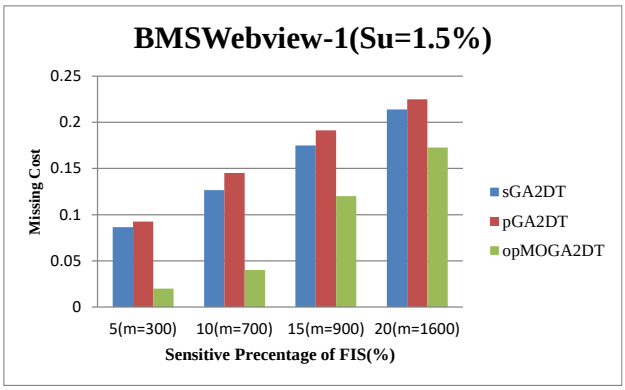



**شکل ۳: نرخ شکست پنهان‌سازی**

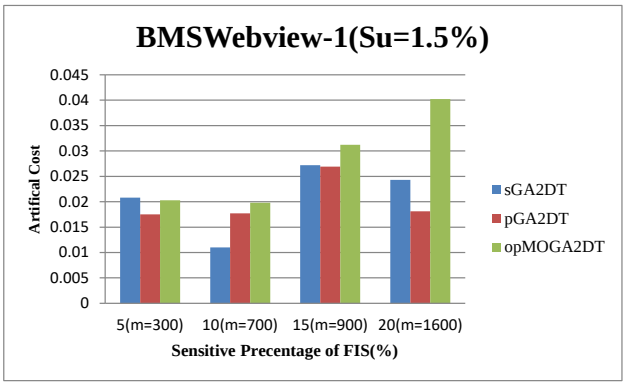
برای محاسبه نرخ عناصر گم‌شده‌ی الگوریتم‌های مختلف، پس از اعمال هر الگوریتم تعداد عناصر پرتکرار پایگاه داده ایمن شده و تعداد عناصر پرتکرار پایگاه داده اصلی را شمارش‌شده با توجه به رابطه (۷) نرخ عناصر گم‌شده به دست می‌آید.

برای محاسبه نرخ عناصر مصنوعی جدید الگوریتم‌های مختلف، پس از اعمال هر الگوریتم تعداد عناصر پرتکرار پایگاه داده ایمن شده و تعداد عناصر پرتکرار پایگاه داده اصلی را شمارش‌شده با توجه به رابطه (۸) نرخ عناصر مصنوعی جدید به دست می‌آید.

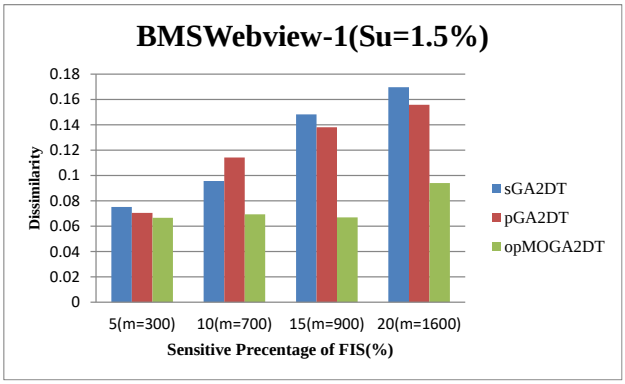
برای محاسبه نرخ عدم تشابه الگوریتم‌های مختلف، پس از اعمال هر الگوریتم تعداد کل عناصر پایگاه داده ایمن شده و تعداد کل عناصر پایگاه داده اصلی را شمارش‌شده با توجه به رابطه (۹) نرخ عدم تشابه به دست می‌آید.




شکل ۴: نرخ عناصر گم‌شده




شکل ۵: نرخ عناصر مصنوعی جدید




شکل ۶: نرخ عدم تشابه

دسته دوم آزمایشات FIs=٪۵ و m=۳۰۰ و Suهای مختلف

۱- با Su=٪۱/۵ (تعداد عناصر حساس برابر ۲ است.)

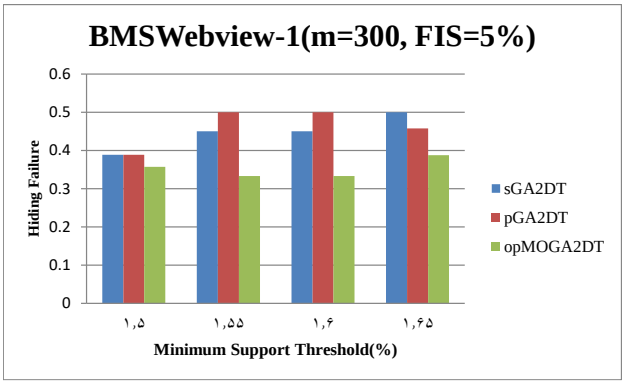
۲- با Su=٪۱/۵۵ (تعداد عناصر حساس برابر ۲ است.)

۳- با Su=٪۱/۶ (تعداد عناصر حساس برابر ۲ است.)

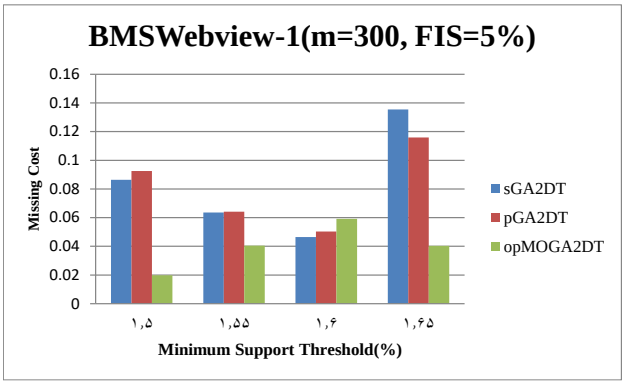
۴- با Su=٪۱/۶۵ (تعداد عناصر حساس برابر ۲ است.)

در دسته دوم آزمایشات درصد عناصر پرتکرار برابر با مقدار ثابت ۵ درصد و تعداد تراکنش‌ها جهت حذف شدن برابر با مقدار ثابت ۳۰۰ خواهد بود ولی مقدار حداقل آستانه پشتیبانی هر بار تغییر می‌کند. محور افقی درصد حداقل آستانه پشتیبان را نشان می‌دهد.

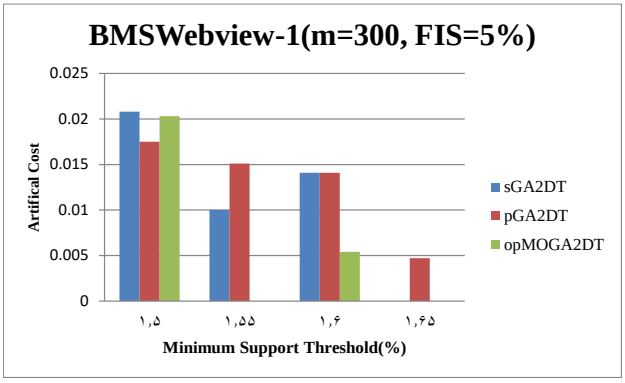
نمودار مربوط به این دسته از آزمایشات برای نرخ شکست پنهان‌سازی در شکل ۷، نرخ عناصر ازدست‌رفته در شکل ۸، نرخ عناصر مصنوعی در شکل ۹ و نرخ عدم تشابه در شکل ۱۰ قابل‌ملاحظه می‌باشد.




شکل ۷: نرخ شکست پنهان‌سازی



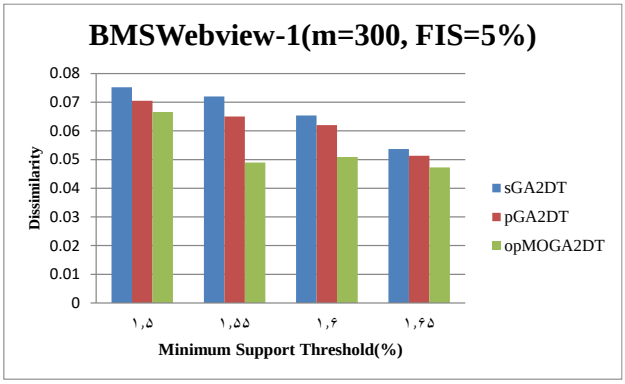

شکل ۸: نرخ عناصر گم‌شده




شکل ۹: نرخ عناصر مصنوعی جدید

در دسته دوم آزمایشات با توجه به این‌که مقدار Su برای هر آزمایش متفاوت است الگوریتم پیشنهادی توانسته است کاهش معیار شکست پنهان‌سازی و معیار عدم تشابه را برآورده سازد؛ زیرا الگوریتم پیشنهادی با استفاده از الگوریتم ژنتیک چندهدفه و انتخاب تراکنش‌هایی با طول کمتر و یا تراکنش‌هایی با بیشترین عنصر حساس و کمترین عنصر غیر حساس، موجب کاهش شکست پنهان‌سازی و کاهش عناصر ازدست‌رفته

شده است؛ اما ازنظر کاهش عناصر ازدست‌رفته و ایجاد عناصر مصنوعی نتوانسته است در همه حالت‌ها از الگوریتم‌های مرجع کاهش یابد.




**شکل ۱۰: نرخ عدم تشابه**

## ۶- نتیجه‌گیری

قواعد انجمنی، برای استخراج ارتباط پنهان و وابستگی‌های میان مجموعه عناصر مختلف در پایگاه داده به‌کاررفته که باعث افشای اطلاعات حساس و تهدید محرمانگی اطلاعات می‌شوند. درروش پیشنهادی این مقاله با به‌کارگیری الگوریتم ژنتیک چندهدفه و حذف تراکنش‌هایی که شامل عناصر حساس هستند، باعث کاهش پشتیبانی عناصر حساس به کمتر از حداقل آستانه پشتیبانی می‌شود که این امر موجب ایمن‌سازی پایگاه داده می‌گردد. روش پیشنهادی در هر تکرار، تنها با یک‌بار پویش تراکنش‌های حساس به‌جای پویش کل تراکنش‌های پایگاه داده، باعث افزایش سرعت و کاهش هزینه‌های اجرا می‌گردد. همچنین برای کاهش عوارض ناشی از پنهان‌سازی، تراکنش‌ها بر اساس کمترین طول یا بیشترین عنصر حساس و کمترین عنصر غیر حساس مرتب‌سازی می‌شوند. تفاوت کار حاضر با مرجع [۲۸] در این است که در [۲۸] پنهان‌سازی با الگوریتم ژنتیکی که جمع چند تابع را کمینه می‌کند انجام می‌شود در صورتی‌که در پژوهش حاضر با استفاده از الگوریتم‌های تکاملی چندهدفه و حذف تراکنشی که دارای بیشترین عنصر حساس و کمترین عنصر غیر حساس است عمل پنهان‌سازی عناصر حساس انجام می‌گیرد.

الگوریتم پیشنهادی با الگوریتم‌های مرجع بر روی پایگاه داده BMSWebView-1 مورد ارزیابی قرار گرفت. برای پنهان‌سازی مجموعه عناصر حساس در الگوریتم‌های مرجع، از الگوریتم ژنتیک معمولی استفاده‌شده است؛ که برای محاسبه تابع برازندگی، چند تابع را با یکدیگر جمع نموده و حاصل جمع را کمینه می‌کند درصورتی‌که لزوماً تک‌تک توابع کمینه نخواهند شد. ازاین‌رو الگوریتم پیشنهادی در این پژوهش، با به‌کارگیری الگوریتم‌های تکاملی چندهدفه، این ضعف را برطرف نموده و هم‌زمان تمامی توابع برازندگی کمینه شده‌اند. همچنین برای کاهش خطای مربوط به پنهان‌سازی و اثرات ناشی از آن، تراکنش‌هایی با کمترین طول و یا تراکنش‌هایی با بیشترین پشتیبانی از عناصر حساس و کمترین عناصر غیر حساس، به‌عنوان درصدی از جمعیت اولیه انتخاب می‌شوند که این امر موجب تسریع بخشی در اجرای الگوریتم می‌شود و عملکرد روش را بهبود می‌بخشد. الگوریتم پیشنهادی در پایگاه داده BMSWebView-1 هدف اول و دوم پنهان‌سازی یعنی کاهش شکست پنهان‌سازی با ۸/۲۲٪ بهبود و کاهش عناصر ازدست‌رفته با ۵/۵۳٪ بهبود را برآورده ساخته است؛ اما ازنظر ایجاد عناصر مصنوعی نتوانسته است در همه حالت‌ها از الگوریتم‌های مرجع کاهش یابد و باعث افزایش عناصر مصنوعی در بعضی حالت‌ها شده است. همچنین ازلحاظ معیار عدم تشابه توانسته است با ۲/۸۸٪ بهبود این معیار را نسبت به الگوریتم‌های مرجع کاهش دهد. درصورتی‌که انتخاب تراکنش‌های حساس به ازای هر عنصر حساس به‌صورت تصادفی صورت نگیرد و تعداد انتخاب تراکنش‌های حساس برای هر عنصر حساس به‌طور جداگانه کنترل شود احتمال کاهش عوارض جانبی پنهان‌سازی وجود خواهد داشت.

## پیوست الگوریتم استخراج عناصر Pre-Large

• **INPUT:** A lower support threshold Sl, an upper support threshold Su, a set of pre-stored large and pre-large itemsets in the original database D consisting of (d − c) records, and a set of deleted records T consisting of t records.

• **OUTPUT:** A set of final association rules for the updated database U.

**Step 1:** Calculate the safety bound of deleted records, f, according to
Theorem 1 as follows:

$$f \le \left\lfloor \frac{(Su - Sl) \times d}{Su} \right\rfloor$$

**Step 2:** Set k =1, where k records the number of items currently being processed.

**Step 3:** Find all the k-itemsets $R_K^T$ with their counts from T that exists in the pre-stored large k-itemsets $L_K^D$ or in the pre-large k-itemsets $P_K^D$ of D.

Step 4: For each itemset I existing in $L_K^D$ , do the following substeps (for managing Cases 1 to 3):

Substep 4-1: If I exists in $R_K^T$ , hen set the new count SU(I) = SD(I)− ST(I); otherwise, set SU(I) = SD(I).

Substep 4-2: If SU(I)/(d − c − t) ≥ Su, assign I as a large itemset, set SD(I) = SU(I) and keep I with SD(I); otherwise, if SU(I)/(d − c − t) ≥ Sl, assign I as a pre-large itemset, set SD(I)=SU(I) and keep I with SD(I); otherwise, neglect I.

**Step 5:** For I existing in $P_K^D$ , do the following substeps (for managing Cases 4 to 6):

Substep 5-1: If I exists in $R_K^T$ , then set the new count SU(I) = SD(I)− ST(I); otherwise, set SU(I) = SD(I).

Substep 5-2: If SU(I)/(d − c − t) ≥ Su, assign I as a large itemset, set SD(I) = SU(I) and keep I with SD(I); otherwise, if SU(I)/(d − c − t) ≥Sl, assign I as a pre-large itemset, set SD(I) = SU(I) and keep I with SD(I); otherwise, neglect I.

**Step 6:** If (t + c) ≤ f, then do nothing; otherwise, rescan D to determine large or pre-large itemsets (for managing Cases 7 to 9).

**Step 7:** Generate candidate (k + 1)-itemsets Ck+1 from ($L_K^U$ U $P_K^U$) in a way similar to that in the apriori algorithm (Agrawal & Srikant, 1994).

**Step 8:** Set k = k + 1.

**Step 9:** Set $L_K^D$=$L_K^D \cap C_K$ and $P_K^D$= $P_K^D \cap C_K$ .

**Step 10:** Repeat STEPs 3 to 9 until no new large or pre-large itemsets are found.

**Step 11:** Derive the association rules from the final large itemsets for U.
**Step 12:** If (t + c) > f, then set d = d − c − t and set c = 0; otherwise, set c = c + t.
After Step 12, the final association rules for the updated database can then be found.

## مراجع


[1] C. C. Aggarwal and S. Y. Philip, "A general survey of privacy-preserving data mining models and algorithms," Privacy-preserving data mining, pp. 11-52, 2008.

[2] R. Agrawal and R. Srikant, "Privacy-preserving data mining," in ACM Sigmod Record, vol. 29, no. 2, pp. 439-450, 2000.

[3] Y. Lindell and B. Pinkas, "Privacy preserving data mining," in Annual International Cryptology Conference, pp. 36-54, 2000.

[۴] محمدعلی زارع چاهوکی، سید حمیدرضا محمدی، «بهینه‌سازی هسته‌های چندگانه در ماشین‌بردارپشتیبان جفتی برای کاهش شکاف معنایی تشخیص صفحات فریب‌آمیز»، مجله مهندسی برق دانشگاه تبریز، دوره ۴۶، شماره ۴، ص. ۱۳۵-۱۴۵، ۱۳۹۵.

[۵] مجید محمدپور، حمید پروین، «الگوریتم ژنتیک آشوب گونه مبتنی بر حافظه و خوشه بندی برای حل مسائل بهینه سازی پویا»، مجله مهندسی برق دانشگاه تبریز، دوره ۴۶، شماره ۳، ص. ۲۹۹-۳۱۸، ۱۳۹۵.

[۶] مرتضی به‌نام، حسین پورقاسم، «شناسایی صرع بر اساس بهینه‌سازی ویژگی‌های ادغامی تبدیل هارتلی با مدل ترکیبی MLP و GA همراه با استراتژی یادگیری ممتیک»، مجله مهندسی برق دانشگاه تبریز، دوره ۴۵، شماره ۴، ص. ۵۱-۶۷، ۱۳۹۴.

[7] M. Atallah, E. Bertino, A. Elmagarmid, M. Ibrahim, and V. Verykios, "Disclosure limitation of sensitive rules," In: Proceedings of the 1999 Workshop on Knowledge and Data Engineering Exchange (KDEX'99), pp. 45-52, 1999.

[8] E. Dasseni, V. S. Verykios, A. K. Elmagarmid, and E. Bertino, "Hiding association rules by using confidence and support," in International Workshop on Information Hiding, pp. 369-383, 2001.

[9] Y. Saygin, V. S. Verykios, and C. Clifton, "Using unknowns to prevent discovery of association rules," Acm Sigmod Record, vol. 30, no. 4, pp. 45-54, 2001.

[10] Y. Saygin, V. S. Verykios, and A. K. Elmagarmid, "Privacy preserving association rule mining," In: Proceedings of 12th International Workshop on Research Issues in Data Engineering: Engineering E-Commerce/E-Business Systems, pp. 151-158, 2002.

[11] V. S. Verykios, A. K. Elmagarmid, E. Bertino, Y. Saygin, and E. Dasseni, "Association rule hiding," IEEE Transactions on knowledge and data engineering, vol. 16, no. 4, pp. 434-447, 2004.

[12] A. Amiri, "Dare to share: Protecting sensitive knowledge with data sanitization," Decision Support Systems, vol. 43, no. 1, pp. 181-191, 2007.

[13] C. N. Modi, U. P. Rao, and D. R. Patel, "Maintaining privacy and data quality in privacy preserving association rule mining," In: Proceedings of International Conference on Computing Communication and Networking Technologies (ICCCNT), pp. 1-6, 2010.

[14] K. Shah, A. Thakkar, and A. Ganatra, "Association rule hiding by heuristic approach to reduce side effects & hide multiple RHS items," International Journal of Computer Applications, vol. 45, no. 1, pp. 1-7, 2012.

[15] N. H. Domadiya and U. P. Rao, "Hiding sensitive association rules to maintain privacy and data quality in database," In: Proceedings of 3rd International Advance Computing Conference (IACC), pp. 1306-1310, 2013.

[۱۶] زهرا کیانی ابری، محمد نادری دهکردی، «الگوریتمی اکتشافی برای پنهان سازی مجموعه عناصر حساس»، دومین همایش ملی علوم و مهندسی کامپیوتر، کامپیوتر، دانشکده مهندسی کامپیوتر، دانشگاه آزاد اسلامی نجف آباد، ۱۳۹۳.

[۱۷] [17] حامد ارشادی‌پور، سید مجید مزینانی، «ارائه‌ی یک روش بهبود یافته جهت مخفی سازی قوانین انجمنی حساس در داده کاوی»، اولین کنفرانس سراسری توسعه محوری مهندسی عمران، معماری، برق و مکانیک ایران، گرگان، دانشگاه گلستان،، ۱۳۹۳.

[18] P. Cheng, J. F. Roddick, S.-C. Chu, and C.-W. Lin, "Privacy preservation through a greedy, distortion-based rule-hiding method," Applied Intelligence, vol. 44, no. 2, pp. 295-306, 2016.

[19] N. H. Domadiya and U. P. Rao, "A Hybrid Technique for Hiding Sensitive Association Rules and Maintaining Database Quality," In: Proceedings of First International Conference on Information and Communication Technology for Intelligent Systems: Vol. 2, pp. 359-367, 2016.

[20] X. Sun and P. S. Yu, "A border-based approach for hiding sensitive frequent itemsets," In: Proceedings of Fifth IEEE International Conference on Data Mining (ICDM), p. 8, 2005.

[21] G. V. Moustakides and V. S. Verykios, "A MaxMin approach for hiding frequent itemsets," Data & Knowledge Engineering, vol. 65, no. 1, pp. 75-89, 2008.

[22] S. Menon, S. Sarkar, and S. Mukherjee, "Maximizing accuracy of shared databases when concealing sensitive patterns," Information Systems Research, vol. 16, no. 3, pp. 256-270, 2005.

[23] A. Gkoulalas-Divanis and V. S. Verykios, "Exact knowledge hiding through database extension," IEEE Transactions on Knowledge and Data Engineering, vol. 21, no. 5, pp. 699-713, 2009.

[24] M. N. Dehkordi, K. Badie, and A. K. Zadeh, "A novel method for privacy preserving in association rule mining based on genetic algorithms," Journal of software, vol. 4, no. 6, pp. 555-562, 2009.

[25] C.-W. Lin, T.-P. Hong, K.-T. Yang, and S.-L. Wang, "The GA-based algorithms for optimizing hiding sensitive itemsets through transaction deletion," Applied Intelligence, vol. 42, no. 2, pp. 210-230, 2015.

[26] C.-W. Lin, B. Zhang, K.-T. Yang, and T.-P. Hong, "Efficiently hiding sensitive itemsets with transaction deletion based on genetic algorithms," The Scientific World Journal, vol. 2014, 2014.

[27] P. Cheng, J.-S. Pan, and C.-W. L. Harbin, "Use EMO to protect sensitive knowledge in association rule mining by removing items," In: Proceedings of IEEE Congress on Evolutionary Computation (CEC), pp. 1108-1115, 2014.

[۲۸] فرشاد شهسواری، محمد نادری دهکردی، ارائه روشی بهینه برای مخفی کردن مجموعه عناصر حساس مبتنی بر الگوریتم ژنتیک، پایان‌نامه کارشناسی ارشد، دانشگاه آزاد اسلامی نجف‌آباد، ۱۳۹۴.

[29] M. H. Afshari, M. N. Dehkordi, and M. Akbari, "Association rule hiding using cuckoo optimization algorithm," Expert Systems with Applications, vol. 64, pp. 340-351, 2016.

[30] J. C.-W. Lin, Q. Liu, P. Fournier-Viger, T.-P. Hong, M. Voznak, and J. Zhan, "A sanitization approach for hiding sensitive itemsets based on particle swarm optimization," Engineering Applications of Artificial Intelligence, vol. 53, pp. 1-18, 2016.

[31] P. Cheng, I. Lee, C.-W. Lin, and J.-S. Pan, "Association rule hiding based on evolutionary multi-objective optimization," Intelligent Data Analysis, vol. 20, no. 3, pp. 495-514, 2016.

[32] C.-W. Lin, T.-P. Hong, C.-C. Chang, and S.-L. Wang, "A greedy-based approach for hiding sensitive itemsets by transaction insertion," Journal of Information Hiding and Multimedia Signal Processing, vol. 4, no. 4, pp. 201-227, 2013.

[33] T.-P. Hong and C.-Y. Wang, "Maintenance of association rules using pre-large itemsets," Intelligent databases: technologies and applications, pp. 44-60, 2007.

[34] T.-P. Hong, C.-Y. Wang, and Y.-H. Tao, "A new incremental data mining algorithm using pre-large itemsets," Intelligent Data Analysis, vol. 5, no. 2, pp. 111-129, 2001.

[35] J. E. Beasley and P. C. Chu, "A genetic algorithm for the set covering problem," European journal of operational research, vol. 94, no. 2, pp. 392-404, 1996.

[36] E. Cantú-Paz, "A survey of parallel genetic algorithms," in Calculateurs paralleles, 1998.

[37] N. N. Lakhan, "on multi-objective linear and non-linear programming," University of the South Pacific, 2015.

[38] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," IEEE transactions on evolutionary computation, vol. 6, no. 2, pp. 182-197, 2002.

## زیرنویس‌ها

[1] Privacy preserving in data mining

[2] Heuristic approach

[3] Exact approach

[4] Border-Base Approach

[5] Lattice-like graph

[6] Decrease support of right hand side item of rule clusters

[7] Advanced decrease support of right hand side item of rule clusters

[8] Remove and reinsert L.H.S of rule

[9] Simple genetic algorithm to delete transaction

[10] Pre-large genetic algorithm to delete transaction

[11] Compact genetic algorithm

[12] Compact pre-large genetic algorithm to delete transaction

[13] Evolutionary Multi Objective

[14] Cuckoo optimization algorithm for the sensitive association rules hiding

[15] particle swarm optimization to delete transaction

[16] evolutionary multi-objective optimization- rule hiding

[17] Frequent item

[18] Large itemsets

[19] Dataset pre-sanitization process

[20] Hiding failure

[21] Missing itemsets

[22] Artificial Items

[23] Probability of Crossover

[24] Partially matched crossover (PMX)

[۲۵] Probability of Mutation (pm)

[۲۶] Swap Mutation

[۲۷] Random Resetting Mutation

[۲۸] Dominant

[۲۹] ε - Constraint)

[۳۰] Goal Attainment

[۳۱] Normal Boundary Intersection

[۳۲] Non-dominated Sorting)

[۳۳] Crowding Distance

[۳۴] Selection or Reproductive

[۳۵] tournament

[36] Dissimilarity

[37] Ordered pre-large multi objective genetic algorithm to delete transaction