Journal of Information Technology Management
Vol. 9, No. 4
2017
DOI: 10.22059/jitm.2017.237094.2088

University of Tehran
Faculty of Management

A New Mechanism to Improve the Detection Rate of Shilling Attacks in the Recommender Systems

Javad Nehriri [1], Sasan Hosseinalizadeh [2]

Abstract: Recommender systems are widely used, in social networks and online stores, to overcome the problems caused by the large amount of information. Most of these systems use a collaborative filtering method to generate recommendations to the users. But, as in this method users' feedback is considered for recommendations, it can be significantly erroneous by the malicious people. In other words, there may be some users who open fake profiles and vote one-sided or biased in the system that may cause disturbance in providing proper recommendations to other users. This kind of damage is said to be shiling attacks. If the attackers succeed, the user's trust in the recommender systems will reduce. In recent years, efficient attack detection algorithms have been proposed, but each has its own limitations. In this paper, we use profile-based and item-based algorithms to provide a new mechanism to significantly reduce the detection error for shilling attacks.

Key words: ***Collaborative filtering, HHT algorithm, Recommender systems, SDF algorithm, Shilling attacks.***

1. MSc. Student, Faculty of Computer and Information Technology Engineering, Qazvin Branch, Islamic Azad University, Qazvin, Iran
2. Assistant Prof., Faculty of Computer and Information Technology Engineering, Qazvin Branch, Islamic Azad University, Qazvin, Iran

Submitted: 04 / July / 2017
Accepted: 02 / December / 2017
Corresponding Author: Sasan Hosseinalizadeh
Email: sasan.h.alizadeh@qiau.ac.ir

مدیریت فناوری اطلاعات
دانشکدهٔ مدیریت دانشگاه تهران
دورهٔ ۹، شمارهٔ ۴
زمستان ۱۳۹۶
صص. ۸۹۲- ۸۷۱

Journal of Information Technology Management
DOI: 10.22059/jitm.2017.237094.2088

سازوکار جدیدی برای کاهش خطای تشخیص حملات شیلینگ در سیستم‌های توصیه‌گر

جواد نحریری[۱]، ساسان حسینعلی‌زاده[۲]

چکیده: در شبکه‌های اجتماعی و فروشگاه‌های اینترنتی، برای مواجهه با مشکلات برآمده از حجم فراوان اطلاعات، به‌طور گسترده‌ای از سیستم‌های توصیه‌گر استفاده می‌شود. پالایش مشارکتی روشی است که اغلب این سیستم‌ها برای تولید توصیه به کاربر استفاده می‌کنند. از آنجا که این روش، برای توصیه، رأی کاربران را در نظر می‌گیرد، رأی افراد مخرب می‌تواند آسیب‌های شایان توجهی به آن وارد کند. به بیان دیگر، ممکن است کاربرانی وجود داشته باشند که با ایجاد پروفایل‌های جعلی، آرای خود را به‌صورت مغرضانه، وارد سیستم کنند و باعث اختلال در تولید توصیه‌ای مناسب به سایر کاربران شوند. به این نوع آسیب، حملات شیلینگ گفته می‌شود. اگر مهاجمان در این امر موفق شوند، اعتماد کاربران به سیستم‌های توصیه‌گر کاهش خواهد یافت. در سال‌های اخیر، الگوریتم‌های تشخیص حملهٔ خوبی ارائه شده است، اما هر یک محدودیت‌هایی دارند. در این مقاله با استفاده از الگوریتم‌های مبتنی بر پروفایل و آیتم، سازوکاری ارائه شده است که خطای تشخیص حملات شیلینگ را به‌صورت چشمگیری کاهش می‌دهد.

واژه‌های کلیدی: پالایش مشارکتی، حملات شیلینگ، سیستم‌های توصیه‌گر، الگوریتم SDF، الگوریتم HHT.

۱. دانشجوی کارشناسی ارشد نرم‌افزار، دانشکدهٔ مهندسی کامپیوتر و فناوری اطلاعات، واحد قزوین، دانشگاه آزاد اسلامی، قزوین، ایران
۲. استادیار دانشکدهٔ مهندسی کامپیوتر و فناوری اطلاعات، دانشکدهٔ مهندسی کامپیوتر و فناوری اطلاعات، واحد قزوین، دانشگاه آزاد اسلامی، قزوین، ایران

تاریخ دریافت مقاله: ۱۳۹۶/۰۴/۱۳
تاریخ پذیرش نهایی مقاله: ۱۳۹۶/۰۹/۱۱
نویسندهٔ مسئول مقاله: ساسان حسینعلی زاده
E-mail: sasan.h.alizadeh@qiau.ac.ir

مقدمه

رشد انفجاری مقدار داده‌های دیجیتالی، تعداد بازدیدکنندگان اینترنتی و نیز گسترش سریع و مداوم وب گستردهٔ جهانی[1] و تجارت الکترونیک[2]، به مشکلی جدی در دنیای وب منجر شده بود. سیستم‌های بازیابی اطلاعات[3]، مانند گوگل[4] تا حدی این مشکل را حل کردند، اما اولویت‌بندی و شخصی‌سازی[5] اطلاعات کاربر، رشد محسوسی نکرد تا اینکه، سیستم‌های توصیه‌گر برای بهبود و ایجاد پیشرفت در این مشکل پدیدار شدند. سیستم‌های توصیه‌گر[6]، نوعی سیستم پالایش اطلاعاتی[7] هستند که با مشکل اضافه بار اطلاعات سروکار دارند و از طریق فیلترکردن و قطعه‌قطعه کردن اطلاعات حیاتی، از مقدار زیاد اطلاعات پویای تولیدشده با توجه به علاقه یا رفتار مشاهده‌شدهٔ کاربر در مورد آیتم‌ها[8]، می‌کاهند. به بیان دیگر این سیستم‌ها با توجه به واکنش‌های[9] کاربر، می‌توانند پیش‌بینی کنند که گزینهٔ پیشنهادی را می‌پذیرد یا آن را رد می‌کند (سان و همکاران، ۲۰۱۵). بر این اساس، این سیستم‌ها با روند رو به افزایشی در وب‌سایت‌های تجاری به کار گرفته شدند (کریمی، عسکری و پرسته، ۱۳۹۴).

سیستم‌های توصیه‌کننده هم برای خدمتگزارها[10] و هم برای کاربران[11] مفیدند. آنها در محیط خرید آنلاین، هزینه‌های معامله و انتخاب گزینه‌ها را کاهش می‌دهند. هدف اصلی این سیستم‌ها فراهم آوردن ابزاری است که به‌وسیلهٔ آن بتوانند کاربران را در یافتن سریع و مناسب اطلاعات و رفع نیازها یاری کنند (مطهری‌نژاد، ذوالفقارزاده، خدنگی و سعدآبادی، ۱۳۹۵). در حال حاضر بسیاری از سایت‌های تجارت الکترونیک از مزیت‌های این سیستم اطلاعاتی، بهره‌مند شده‌اند. برای مثال، آمازون[12] که یکی از بزرگ‌ترین سایت‌های فروشگاه اینترنتی است و افزون بر ۸۰ میلیون عضو داد، در سال ۲۰۰۶ اعلام کرد که ۳۵ درصد از محصولات خود را به‌وسیلهٔ سیستم‌های توصیه‌گر فروخته است. یا نتفلیکس[13] در همان سال مدعی شد که توانسته ۶۰ درصد از کسب‌وکار اجاره‌نامه‌های خود را با استفاده از سیستم‌های توصیه‌گر معامله کند (ژانگ و فوگو،

---

1. World Wide Web
2. Electronic Business
3. Information Retrieval Systems
4. Google
5. Personalization
6. Recommender Systems
7. Information Filtering
8. Items
9. Actions
10. Servers
11. Users
12. Amazon
13. Netflix

۲۰۱۵). باتوجه به اهمیت این موضوع، تاکنون الگوریتم‌های فراوانی برای ارتقـای سیسـتم‌هـای توصیه‌گر ارائه شده است. یکی از پرکاربرترین و مهم‌ترین تکنیک‌هایی که در این زمینه اسـتفاده می‌شود، پالایش مشارکتی[۱] است و همان‌طور که می‌دانیم ایـن تکنیـک از الگـوی رأی کـاربران برای توصیه‌های خـود اسـتفاده مـی‌کنـد. در سـال ۲۰۰۲ سـایت آمـازون از مشـتریانش دربـارهٔ پیشنهادهای نامناسب سیستم توصیه‌گر خود شکایت‌هـای فراوانـی دریافـت کـرد کـه بعـدها در تحقیقاتی که توسط این سایت انجام شد، نتیجه به فروشندگان بی‌پروا ختم شد (ژانـگ و فوگـو، ۲۰۱۵). پس این ظرفیت وجود دارد که افراد سودجو به این سیسـتم‌هـا توجـه کننـد و بـا ایجـاد پروفایل‌های جعلی[۲] و آرای مغرضانه، درصدد پایین آوردن محصولات رقیبان خـود و بـالا بـردن محصول خود به‌عنوان محصول برتر سیستم‌های توصیه‌گر باشند. بـه ایـن نـوع حملـه، حمـلات شیلینگ[۳] می‌گویند (ایزینکای، فولاجیمی و اوجوکوه، ۲۰۱۵).

مهم‌ترین مسئله‌ای که باعث می‌شود کاربران به پیشنهاد توصیه‌گرها توجه داشته باشند و آن را بپذیرند، اعتماد آنها به محصولات و نزدیک بودن توصیه به علایقشان است. از این رو باید بـه افرادی که با ساختن پروفایل‌های جعلی و دادن آرای مغرضانه درصدد پـایین آوردن محصـولات دیگران و بالا بردن محصولات خود هستند، توجه ویژه‌ای کرد. تاکنون الگـوریتم‌هـای مختلفـی برای شناسایی پروفایل‌های مخرب و پیدا کردن آیتم‌هایی که مورد حمله واقـع‌شـده، ارائـه شـده است که هر یک مشکلاتی دارند. آیا می‌توان الگوریتمی ارائـه داد کـه تشـخیص بهتـر و درصـد خطای کمتری داشته باشد؟ ما در این پژوهش برای پیدا کردن پروفایل‌های حمله و در بر داشتن نتایج بهتر، الگوریتمی ترکیبی[۴] ارائه می‌دهیم.

پیشینهٔ پژوهش

به‌دلیل طبیعت باز سیستم‌های توصیه‌گر پالایش مشارکتی، طراحی سیستمی که مورد حمله واقع نشود، دشوار است، از این رو باید دربارهٔ حملات شیلینگ شناخت داشت. در این زمینه، تحقیقـات به دو گروه دسته‌بندی می‌شوند؛ تکنیک‌های که برای بهبود استحکام[۵] الگوریتم‌هـای توصـیه‌گـر ایجاد شده‌اند و تکنیک‌هایی کـه بـرای تشـخیص حملـه[۶]، بـه کـار گرفتـه مـی‌شـوند. از جملـه راهکارهایی که در زمینهٔ استحکام الگوریتم ارائه شده است، می‌توان به راهکار افزایش تجزیـه و

1. Collaborative filtering
2. Fake profiles
3. Shilling Attacks
4. Hybrid
5. Robustness
6. Detection attack

تحلیل معنایی[1] (محتا و هافمن، ۲۰۰۸)، سازوکار اعتماد[2] (ژانگ، لی و پیتسیلیس، ۲۰۱۳) و کنترل تصدیق[3] (نوح، کانگ، اوه و کیم، ۲۰۱۴) اشاره کرد، اما این تکنیک‌ها نیز روی تمام حملات استحکام ندارند. تکنیک‌های تشخیص حمله خود به دو دستهٔ تشخیص پروفایل حمله[4] و تشخیص آیتم‌های مورد حمله واقع‌شده، طبقه‌بندی می‌شوند.

تشخیص حمله مبتنی بر پروفایل

از لحاظ یادگیری ماشین، تشخیص را می‌توان به سه گروه دسته‌بندی کرد. طبقه‌بندی با نظارت[5] که در این روش از تکنیک‌هایی مانند KNN[6]، C4.5 و SVM برای تفکیک کاربر مهاجم از کاربر عادی استفاده می‌شود و پرکاربردترین آنها SVM است. طبقه‌بندی نیمه‌نظارتی[7] دسته‌ای از روش‌های یادگیری ماشین هستند که در آن برای بهبود دقت یادگیری، همزمان از داده‌های بدون برچسب[8] و داده‌های برچسب‌دار استفاده می‌شود. طبقه‌بندی بدون نظارت[9] که یادگیری روی داده‌های بدون برچسب و برای یافتن الگوهای پنهان در این داده‌ها انجام شود؛ روش خوشه‌بندی[10] (بهامیک، مباشر و بورک، ۲۰۱۱)، روش تجزیه و تحلیل مؤلفهٔ اصلی (PCA)[11] مبتنی بر انتخاب متغیر خوشه (چنگ و هارلی، ۲۰۰۹) و روش توزیع احتمالی بتا[12] (چانگ، اچسو و هانگ، ۲۰۱۳) از جملهٔ این روش‌ها هستند. اما هر یک از این سه دسته، از ویژگی‌هایی برای طبقه‌بندی استفاده می‌کنند که آنها نیز به سه دسته تقسیم می‌شوند. ویژگی‌های عمومی[13]، خاص[14] و درون ـ پروفایلی[15].

۱. ویژگی‌های عمومی: این ویژگی‌ها با استفاده از آمار توصیفی، مشخصه‌هایی از پروفایل کاربران حمله را پیدا می‌کنند که از مشخصه‌های پروفایل کاربران عادی متمایز است. برای مثال،

1. Semantic Analysis
2. Trust Mechanisms
3. Admission Control
4. Attack Profile
5. Supervised
6. K-Nearest Neighbors
7. Semi-Supervised
8. Label
9. UnSupervised
10. Clustering
11. Principal Component Analysis based on selection cluster
12. Beta-Protection
13. Generic
14. Specific
15. Intra-Profile

ویژگی‌های عمومی RDMA[1] و WDMA[2] انحراف از امتیاز را برای هر پروفایل کاربر اندازه‌گیری می‌کند.

انحراف امتیاز از میانگین شرط (RDMA) و درجهٔ تشابه با همسایه‌های بالایی (DegSim)[3] در سال ۲۰۰۵ توسط چریتا ارائه شد.(چیریتا، نجدل و زمفیر، ۲۰۰۵) RDMA، انحراف میانگین پروفایل از هر آیتم را بر اساس وزن معکوس تعداد امتیازهای هر آیتم بررسی می‌کند و Degsim، میانگین همبستگی پیرسون از k همسایهٔ نزدیک است. و در در سال ۲۰۰۶ ویژگی‌های درجهٔ وزن از شرط (DMA)[4] و انحراف وزن از میانگین شرط (WDMA) توسط بورک معرفی شد که از مشتقات RDMA محسوب می‌شوند. LengthVar[5] نوعی ویژگی است که بر اساس تعداد رأی‌های داده‌شده توسط کاربر محاسبه می‌شود. این ایده توسط واریانس اندازه‌گیری طول (LengthVar) اندازه گرفته می‌شود و در اصل میزان طول پروفایل از طول متوسط در پایگاه داده را اندازه‌گیری می‌کند (بورک، مباشر، ویلیامز و بهامیک، ۲۰۰۶).

۲. ویژگی‌های مدل ـ خاص: روش‌های مبتنی بر مدلی هستند که با توجه به دانش قبلی در مورد مدل‌های حمله به اندازه‌گیری صحت پروفایل کاربران می‌پردازند. برای مثال ویژگی واریانس پرکنندهٔ میانگین (MeanVar)[6] با استفاده از الگوی رأی مدل حملهٔ میانگین به این نتیجه رسید که واریانس امتیاز یک آیتم پرکننده (آیتمی است که فرد مخرب برای تشابه بیشتر با کاربر نرمال به آن رأی می‌دهد) بوده و میانگین امتیازهای آن آیتم به هم بسیار نزدیک هستند. این ویژگی فقط برای تشخیص حملهٔ میانگین کاربرد دارد و ویژگی تفاوت پرکنندهٔ متوسط هدف (FMTD)[7] ویژگی‌ای است که از آن می‌توان فقط برای تشخیص پروفایل حملهٔ بندوگن[8] و سگمنت[9] استفاده کرد. این ویژگی برای هر کاربر از طریق تفاوت در بالاترین یا پایین‌ترین امتیاز به آیتم‌ها توسط کاربر نسبت به همهٔ آیتم‌هایی که توسط وی امتیاز داده شده است (به‌جز آیتم‌هایی با بالاترین یا پایین‌ترین امتیاز) محاسبه می‌شود (بورک و همکاران، ۲۰۰۶).

۳. ویژگی‌های درون ـ پروفایلی: بر خلاف ویژگی‌های مدل خاص، این ویژگی، مشخصه‌های یک پروفایل را به‌تنهایی بررسی می‌کند. برای مثال، اندازهٔ پرکننده با مجموعهٔ آیتم‌ها (FSTI)[10]

---

1. Rating Deviation from Mean Agreement
2. Weighted Deviation from Mean Agreement
3. Degree of Similarity with Top Neighbors
4. Deviation from Mean Agreement
5. Length Variance
6. Mean Variance
7. Filler Mean Target Difference
8. Bandwagon
9. Segment
10. Filler Size with Total Items

نسبت تعداد آیتم‌های رأی داده‌شده توسط کاربر به تعداد آیتم‌های کل سیستم توصیه‌گر یک ویژگی است؛ یا اندازهٔ پرکننده با آیتم‌های مشهور (FSPI)[1] نسبت تعداد آیتم‌های مشهور رأی داده شده توسط کاربر به تعداد کل آیتم‌های مشهور در سیستم توصیه‌گر است (ژائو و شانگ، ۲۰۱۰). مشکل این ویژگی‌ها این است که در اندازهٔ حملهٔ پایین کارآمد نیستند و معمولاً هر یک برای یک یا دو مدل حمله جوابگو هستند. البته الگوریتم‌هایی نیز معرفی شده‌اند که از ترکیب این ویژگی‌ها استفاده می‌کنند، مانند الگوریتمی که ژائو و همکارانش ارائه دادند و با ترکیب ویژگی‌های RDMA و Degsim و آنالیز آیتم هدف[2] عمل می‌کند.(ژائو و همکاران، ۲۰۱۵) البته این مدل ضمن زمان زیادی که صرف می‌کند، روی همهٔ مدل‌های حمله کارایی ندارد و باید با دانش از نوع حملهٔ رخداده، ویژگی مناسب را انتخاب کرد یا الگوریتم HHT-SVM[3] را به کار برد که از ویژگی‌های درون ـ پروفایلی برای تشخیص کاربران حمله استفاده می‌کند. اگرچه دقت این الگوریتم از سایر الگوریتم‌ها بیشتر است، برای مدل جدید حمله، نیازمند به‌روزرسانی پایگاه دادهٔ خود است (ژانگ و ژوو، ۲۰۱۴).

تشخیص مبتنی بر آیتم

در این روش با پیدا کردن آیتم‌هایی که به آنها حمله شده و بازه‌های زمانی حمله، کار به پایان می‌رسد. مشکل بسیاری از روش‌های تشخیص ناهنجاری آیتم، تعیین اندازهٔ بازهٔ زمانی است. برخی محققان، بازه‌های زمانی را به‌صورت آزمایشی تعیین کردند و برخی نیز با طراحی رویکرد اکتشافی به حل این مشکل پرداختند. تشخیص آماری ناهنجاری، روشی متکی بر دو تکنیک کنترل حد بازهٔ اطمینان[4] و کنترل حد است (بهامیک، ویلیامز، مباشر و بورک، ۲۰۰۶). در این روش یک آیتم زمانی مشکوک در نظر گرفته می‌شود که مقدار متوسط آن در خارج از سطح اعتماد تنزل کند. نوع دیگر این الگوریتم‌ها، الگوریتمی است که با استفاده از توزیع مربع کای (χ^2)[5] برای مقایسهٔ توزیع رأی‌ها در فواصل زمانی مختلف و تعیین فواصل زمانی غیرعادی، معرفی شده است (گائو و همکاران، ۲۰۱۵). همچنین می‌توان در این زمینه، الگوریتم چارچوب دسته‌بندی پویا (SDF)[6] را که توسط ژیا ارائه شده، نام برد؛ در این چارچوب از تکنیک دسته‌بندی بازهٔ زمانی به‌صورت پویا استفاده شده است (ژیا و همکاران، ۲۰۱۵).

---

1. Filler Size with Popular Items
2. Target Item Analysis
3. Hilbert–Huang Transform and Support Vector Machine
4. Confidence Interval Control Limit
5. Chi Square Distribution
6. Segmented Dynamic Framework

در جدول ۱ ضعف‌ها و قوت‌های چندین الگوریتم مبتنی بر کاربر و آیتم مهم را که در سال‌های اخیر ارائه شده‌اند، بررسی می‌کنیم. با توجه به ضعف‌ها و قوت‌های هر الگوریتم، بهترین الگوریتم در زمینۀ پیدا کردن پروفایل‌های حمله را الگوریتم HHT-SVM می‌دانیم و بهترین الگوریتمی که تاکنون برای پیدا کردن آیتم‌های مورد حمله واقع‌شده، ارائه شده است را الگوریتم SDF در نظر می‌گیریم.

جدول ۱. ضعف‌ها و قوت‌های الگوریتم‌ها

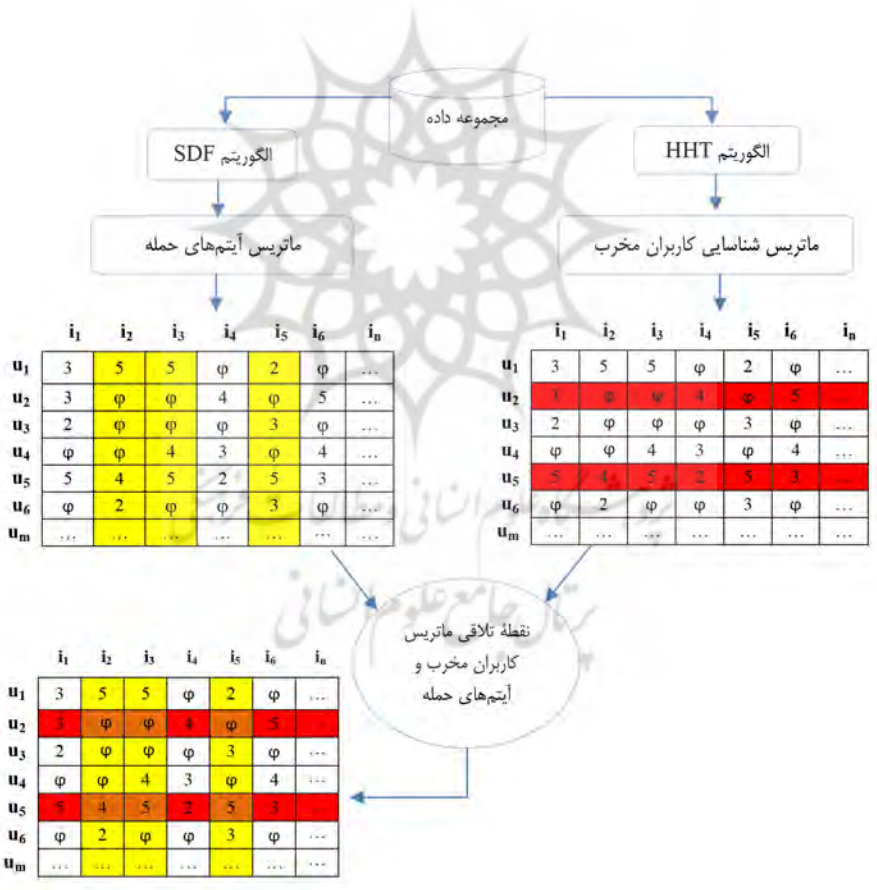
تکنیک	نوع تشخیص/ سال انتشار	تجربی/ نظری	قوت‌ها	ضعف‌ها
PCA-Based	کاربر / ۲۰۰۷	تجربی	• ساده‌ترین روش	• عملکرد ضعیف در ماتریس خلوت
βP	کاربر / ۲۰۱۳	تجربی	• عملکرد خوب در ماتریس خلوت	• عدم پویایی به‌علت استفاده از توزیع بتا • عملکرد ضعیف در برابر اندازۀ حملات بزرگ
RD-TIA	کاربر / ۲۰۱۵	تجربی	• دقت زیاد	• نیاز به دانش نسبت به نوع حمله • زمان زیاد برای تشخیص
HHT-SVM	کاربر / ۲۰۱۴	نظری	• به بررسی کل پایگاه داده نیاز ندارد.	• نیازمند به‌روزرسانی پایگاه داده برای به‌روزرسانی مدل‌های جدید حمله
x-bar	آیتم / ۲۰۰۶	تجربی	• پیچیدگی زمانی ندارد	• دقت کم در بازه‌های با تراکم امتیازی
χ^2	آیتم / ۲۰۱۵	تجربی	• پویایی	• هشدار غلط زیاد که ناشی از مقدار آستانۀ توقف یا تقسیم بازۀ زمانی نامناسب است.
SDF	آیتم / ۲۰۱۵	تجربی	• به دانش در مورد حملات نیاز ندارد • پیچیدگی زمانی خطی • مقیاس‌پذیر، مؤثر در حملۀ وسیع	• عملکرد ضعیف در مقابله با حملات تغییر هدف (اگر کاربران حمله امتیاز max-1 به آیتم‌های هدف بدهند).

روش‌شناسی پژوهش

با توجه به مشکلات بیان‌شده برای پیدا کردن پروفایل‌های حمله، در صدد بهبود خطای تشخیص[1] این پروفایل‌ها برآمدیم و الگوریتم جدیدی متشکل از دو الگوریتم مبتنی بر پروفایل و آیتم که به‌صورت موازی روی مجموعۀ دادۀ کاربر- آیتم - امتیاز اعمال می‌شوند، ارائه دادیم.

1. False Alarm Rate

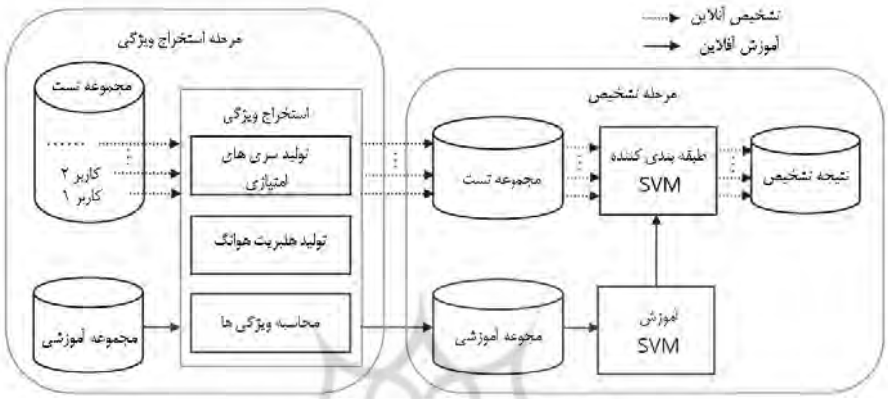
نوآوری ما در این است که توانستیم در یک روش ترکیبی برای پیدا کردن پروفایل‌های حمله، از الگوریتمی که صرفاً برای پیدا کردن آیتم‌های مورد حمله واقع‌شده استفاده می‌شود، بهره بریم. تکنیک نخست یا اصلی HHT به شناسایی پروفایل‌های حمله می‌پردازد و ماتریسی از آنها ارائه می‌دهد. تکنیک دوم یا کمکی SDF، آیتم‌هایی را که در بازه‌های خاص به آنها حمله شده است، شناسایی می‌کند و با پیدا شدن این آیتم‌ها، می‌توان پروفایل‌هایی را که به آنها رأی داده‌اند، پیدا کرد و آنها را در ماتریسی جداگانه، ذخیره نمود. نقطهٔ تلاقی درایه‌های غیرتهی‌ای که در این دو تکنیک تشخیص داده شوند را به‌عنوان پروفایل مهاجم در نظر می‌گیریم. به این نکته باید توجه کرد که تکنیک دوم، تکنیکی مبتنی بر آیتم است و به همین علت در تشخیص پروفایل‌های حمله خطای بیشتری خواهد داشت. شکل ۱ معماری سیستم پیشنهادی الگوریتم ترکیبی این پژوهش را نشان می‌دهد.



شکل ۱. معماری الگوریتم پیشنهادی

الگوریتم HHT

همان‌طور که در شکل ۲ دیده می‌شود، این الگوریتم از دو فاز اصلی به‌وجود آمده است. در فاز نخست، ویژگی‌ها استخراج[1] می‌شوند و در فاز دوم ماشین بردار پشتیبان به تشخیص می‌پردازد.



شکل ۲. معماری الگوریتم HHT

ژانگ و همکاران (۲۰۱۴)

در فاز نخست، رأی‌های هر پروفایل طی فرایندی باید به سیگنال تبدیل شوند، این کار ابتدا با مرتب‌سازی سری‌های امتیازی[2] یک کاربر بر حسب مشهوریت[3] یا تازگی[4] آیتم انجام می‌شود (البته به‌دلیل زمان اجرای بسیار زیاد مرتب‌سازی بر حسب تازگی، از آن صرف‌نظر کردیم). آنگاه سری امتیازی مرتب‌شده را با شرط زیر به سیگنال با دامنهٔ ۱ تا ۱- تبدیل می‌کنیم.

رابطهٔ ۱)
$$PBRS_u(i) = \begin{cases} 1 & , \ r_{u,i} \neq \perp \ \wedge (i = 1 \ \vee \ PBRS_u(i-1) \neq 1 \\ -1, & r_{u,i} = \perp \ \wedge (i = 1 \ \vee \ PBRS_u(i-1) \neq -1 \\ 0 & , \ otherwise \end{cases}$$

اگر کاربر u به آیتم i رأی داده باشد، با شرط اینکه آیتمِ نخست باشد یا آیتم قبل از آن ۱ نباشد، عدد ۱ را جایگزین آن می‌کنیم و اگر کاربر u به آیتم i رأی نداده باشد با شرط اینکه آیتمِ نخست باشد یا آیتم قبل از آن ۱- نباشد، جای آن ۱- می‌گذاریم. در غیر این صورت، هر شرط دیگری رخ دهد، جای آن ۰ قرار می‌دهیم. حال باید از این سیگنال اطلاعات به‌دست آوریم.

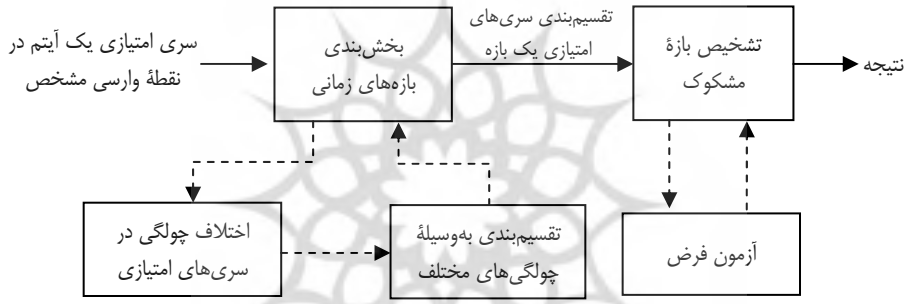
---

1. Feature Extraction
2. Rating Series
3. Popularity
4. Novelty

یکی از قدرتمندترین ابزاری که برای این کار وجود دارد، تبدیل هلبریت هوانگ است که شامل دو بخش تجزیهٔ حالت تجربی (EMD)[1] برای تجزیهٔ هر سیگنال و تابع‌های حالت ذاتی (IMF)[2] برای استخراج ویژگی‌هاست. ویژگی‌های میانگین فاز[3] نسبت به تعداد آیتم‌های مشهور، میانگین فاز نسبت به تمام آیتم‌ها، میانگین دامنه[4] نسبت به تعداد آیتم‌های مشهور و میانگین دامنه نسبت به تمام آیتم‌ها، تفکیک‌کنندهٔ مناسبی برای پروفایل‌های عادی از مخرب هستند که در کار خود از آنها برای آموزش SVM استفاده کردیم. در نهایت ماتریسی از پروفایل‌های حمله و عادی به‌عنوان پروفایل حمله به‌دست آوردیم.

الگوریتم SDF

این الگوریتم شامل دو فاز اصلی است، تقسیم‌بندی بازهٔ زمانی[5] و تشخیص بازهٔ مشکوک[6].



شکل ۳. معماری الگوریتم SDF

ژیا و همکاران (۲۰۱۵)

در فاز نخست، ابتدا سری‌های امتیازی آیتم‌ها را که بر اساس زمان مرتب شده‌اند، به‌وسیلهٔ یک نقطهٔ وارسی[7] به چندین بازهٔ مساوی تقسیم می‌کنیم که با توجه به یافته‌های پژوهشگر این نقطهٔ وارسی را سه روز در نظر گرفتیم. سپس برای بازدهی بهتر، با استفاده از تعیین تراکم[8] هر آیتم (تعداد رأی‌های داده‌شده به آیتم) و چولگی، سری امتیازی هر آیتم را به چندین گروه تقسیم

1. Empirical Mode Decomposition
2. Intrinsic Mode Functions
3. Phase
4. Amplitude
5. Time Interval Segmentation
6. Abnormal Interval Detection
7. Checkpoint
8. Density

می‌کنیم و در فاز دوم با استفاده از آزمون فرض میانگین، مشخص می‌کنیم که آیا در بازه‌های زمانی خاص روی یک آیتم حمله صورت گرفته است یا خیر. برای تعیین تراکم، آیتم‌ها را به دو مجموعه تقسیم می‌کنیم که با آزمون‌های متعدد روی دو مجموعه داده، تراکم زیر ۲۵ رأی و بیشتر از ۱۰۰۰ رأی را انتخاب کردیم (در مقالهٔ ژیا، تراکم بین ۴۰ تا ۸۰ و ۸۰ به بالا در نظر گرفته شده است، اما به‌دلیل مقدار تشخیص کم در اندازه‌های حملهٔ پایین، این دسته‌بندی را عوض کردیم). پس از تقسیم‌بندی آیتم‌ها به‌وسیلهٔ تراکم امتیاز آنها، باید اختلاف چولگی را روی تک‌تک امتیاز آیتم‌ها، بررسی کنیم. اما ابتدا میانگین نمونهٔ[1] ($\bar{X}_k$) و انحراف مجذور میانگین[2] (S_k^+) کل امتیازهای یک آیتم را محاسبه می‌کنیم، سپس برای هر یک از امتیازها نسبت به توزیع امتیازهای قبلی، چولگی[3] را به‌دست می‌آوریم. نخستین تفاوت نظم از مقدار چولگی در هر امتیاز را می‌توانیم از طریق تغییر در مقدار چولگی با همسایهٔ آن محاسبه کنیم، سپس باید امتیازهایی را که هم‌نوع هستند، خوشه‌بندی کنیم که بهترین راه، ضرب اختلاف چولگی دو همسایه پشت سر هم است. به‌ازای هر اختلاف چولگی در سری امتیازی، امتیازها به گروه‌های جدیدی تقسیم می‌شوند.

در آخر چندین گروه داریم که هر گروه امتیاز میانگین $\bar{X}_{ki'}$ و تعداد امتیازهای $n_{ig'}$ خود را دارد و این گروه‌ها به‌عنوان گروه‌های مشکوک در نظر گرفته می‌شوند. پس باید آنها را با آزمون فرض میانگین که در رابطه‌های ۲ و ۳ آورده شده‌اند، بررسی کنیم.

رابطهٔ ۲) $$H_0: \bar{X}_{ki'} \approx \bar{X}_k \qquad H_1: \bar{X}_{ki'} \neq \bar{X}_k$$

رابطهٔ ۳) $$\left\{ \frac{|\bar{X}_{ki'} - \bar{X}_k|}{\delta_k} \sqrt{n_g^{i'}} > u_{1-\partial/2} \right\}$$

بنا بر بهترین نتیجهٔ پژوهشگر، مقدار شرط $u_{1-\partial/2}$ را برابر با ۲/۱۷ در نظر می‌گیریم. اگر رابطهٔ ۳ درست باشد، باید فرض H. را رد کرده، بازهٔ مربوطه را شامل امتیاز مشکوک تلقی کنیم و پروفایل‌هایی که در این بازه هستند را به‌عنوان پروفایل‌های حمله ذخیره کنیم و اگر رابطهٔ ۳ درست نباشد، باید فرض H. را بپذیریم و امتیازهای بازهٔ مربوطه را به‌عنوان امتیازهای سالم در نظر بگیریم. در نهایت ماتریسی از پروفایل‌های حمله و عادی در اختیار داریم که به‌عنوان پروفایل‌های حمله در نظر گرفته شده‌اند.

---

1. Sample Mean
2. Average Squared Deviation
3. Skewness

الگوریتم ترکیبی

با توجه به خروجی دو الگوریتم، دو ماتریس در اختیار داریم که شامل پروفایل‌های حمله و عادی هستند. نقطهٔ تلاقی این دو ماتریس، بیشتر شامل پروفایل‌های حمله بوده و پروفایل‌های عادی کمتری دارد. پس در نتیجه خطای تشخیص را کاهش خواهیم داد که این موضوع در نتیجهٔ آزمایش‌ها به‌طور محسوسی مشاهده می‌شود.

انتخاب و پردازش داده‌ها

برای آزمایش الگوریتم خود، از مجموعه دادهٔ مووی لنز[1] ۱۰۰ هزار و ۱ میلیونی رکوردی[2] استفاده کردیم. این مجموعه داده‌ها سالم هستند و باید پروفایل‌های مخرب را با الگوهایی که در دسترس داریم، ایجاد کرده و داخل این مجموعه داده‌ها تزریق کنیم. در جدول ۲ تعداد کاربران و فیلم‌های موجود در هر مجموعه داده، مشاهده می‌شود.

جدول ۲. مجموعه دادهٔ سالم ۱۰۰ هزار و ۱ میلیون رکوردی

تعداد امتیازها	تعداد کاربران	تعداد فیلم
۱۰۰/۰۰۰	۹۴۳	۱۶۸۲
۱/۰۰۰/۰۰۰	۶۰۴۰	۳۹۰۰

در جدول ۳ نحوهٔ امتیازدهی افراد مخرب به انواع آیتم‌ها دیده می‌شود. این چهار نوع حمله، از مشهورترین نوع حملات هستند که بیشتر پژوهشگران روی آنها کار می‌کنند و ما نیز با ایجاد این نوع حملات به پیاده‌سازی الگوریتم خود پرداختیم.

جدول ۳. الگوی رأی دادن انواع حملات

نوع حمله	آیتم انتخابی / آیتم رأی	آیتم پرکننده / آیتم رأی	آیتم خالی	آیتم هدف
تصادفی	استفاده نمی‌شود	تصادفی میانگین سیستم	I – If	حداکثر/حداقل
میانگین	استفاده نمی‌شود	تصادفی میانگین آیتم	I – If	حداکثر/حداقل
بندوگن	آیتم‌های مشهور	تصادفی میانگین سیستم	I- { If + Is }	حداکثر
سگمنت	آیتم‌های هم‌بخش	تصادفی امتیاز حداکثر	I - { If + Is }	حداکثر

منبع: گونز ، کاللی، بیلج و پولات (۲۰۱۴)

1. Movielens
2. Record

I_S یا آیتم‌های انتخابی[1]: مجموعه‌ای از آیتم‌های انتخاب شده است که کم و بیش به آیتم‌های هدف مربوط است. برای مثال در مجموعه دادهٔ سایت مووی لنز، اگر آیتم هدف در دستهٔ انیمیشن باشد، آیتم‌های انتخاب‌شده نیز در همان دسته خواهند بود.

I_F یا آیتم‌های پرکننده[2]: مجموعه‌ای از آیتم‌های پرکنندهٔ انتخاب‌شده به‌طور تصادفی که کاربر حمله برای پنهان کردن خود استفاده می‌کند.

I_Θ یا آیتم‌های رأی داده‌نشده: مجموعه‌ای از آیتم‌های رأی داده نشده است.

I_t یا آیتم‌های هدف[3]: مجموعه‌ای از آیتم‌های هدف است که کاربر می‌خواهد آنها را بالا برده (پوش)[4] یا پایین بیاورد (ناک)[5] (گونز و همکاران، ۲۰۱۴).

حملهٔ تصادفی[6]: در این نوع حمله، پروفایل‌های حمله به آیتم‌های پرکننده به‌صورت تصادفی و با امتیاز میانگین کل سیستم رأی می‌دهند و برای آیتم‌های هدف نیز امتیاز حداکثری[7] یا حداقلی[8] (حملالت پوش و ناک) در نظر می‌گیرند (گونز و همکاران، ۲۰۱۴).

حملهٔ میانگین[9]: در این نوع حمله، پروفایل‌های حمله به آیتم‌های پرکننده به‌صورت تصادفی ولی با امتیاز میانگین هر آیتم رأی می‌دهند و برای آیتم‌های هدف نیز رأی حداکثری یا حداقلی (حملات پوش یا ناک) در نظر می‌گیرند (گونز و همکاران، ۲۰۱۴).

حملهٔ بندوگن[10] یا حملهٔ محبوب: در این نوع حمله یک مهاجم به آیتم‌های پرکننده امتیاز متوسط سیستم، به آیتم‌های انتخابی امتیاز حداکثری و به آیتم یا آیتم‌های هدف امتیاز حداکثری می‌دهد. به این ترتیب، پروفایل‌های تزریق‌شده می‌توانند به‌راحتی از نظر شباهت با کاربران دیگر همراه شوند. آیتم‌های انتخابی در اینجا، آیتم‌های مشهور هستند و این نوع از حمله فقط حالت پوش دارد (گونز و همکاران، ۲۰۱۴).

حملهٔ سگمنت[11]: این حمله برای هدف قرار دادن گروه خاصی از کاربران که به احتمال زیاد محصول خاصی را خریداری می‌کنند، طراحی شده است. در پروفایل‌های حمله، مهاجم، به

---

1. Selectetd Items
2. Filler Items
3. Target Items
4. Push
5. Nuke
6. Random
7. Maximum
8. Minimum
9. Avarage
10. Bandwagon
11. Segment

آیتم‌های پرکننده که به‌صورت تصادفی انتخاب می‌شوند، امتیاز حداقل و به آیتم‌های انتخابی و آیتم‌های هدف، امتیاز حداکثر می‌دهد (گونز و همکاران، ۲۰۱۴).

توانایی رویداد حمله به‌طور کلی توسط اندازهٔ حمله[1] و اندازهٔ پرکننده[2] سنجیده می‌شود. اندازهٔ حمله، درصد تعداد پروفایل کاربر حمله در یک سیستم توصیه‌گر است. اندازهٔ پرکننده، نسبت تعداد اقلام در یک پروفایل کاربر مخرب به کل اقلام در سیستم توصیه دهنده است که درجهٔ پراکندگی اقلام ـ امتیاز را توصیف می‌کند (گونز و پولات، ۲۰۱۶).

برای هر چهار نوع حمله و هر دو نوع مجموعه داده، ۵۰ پروفایل با اندازهٔ پرکنندهٔ ۱ درصد، ۳ درصد، ۵ درصد، ۱۰ درصد، ۲۵ درصد و ۵۰ درصد ایجاد کردیم و برای پیچیده‌تر شدن حملات نیز، حملات را تک‌هدفه و سه هدفه در نظر گرفتیم.

یافته‌های پژوهش

محیط پیاده‌سازی

به‌منظور پیاده‌سازی الگوریتم‌ها و تولید پروفایل‌های حمله، از نسخهٔ ۲۰۱۴ نرم‌افزار متلب[3] استفاده کردیم و الگوریتم پیشنهادی خود را در رایانه‌ای با مشخصات پردازندهٔ ۵ هسته‌ای، حافظهٔ داخلی ۶ گیگ و سیستم عامل ویندوز ۸.۱ به اجرا درآوردیم.

برآورد پارامترها

هستهٔ تابع SVM را چندجمله‌ای در نظر گرفتیم و هزینهٔ خطا را با توجه به مقالهٔ اصلی ۳۲ قرار دادیم. در هر دو مجموعه داده، ۸۰ درصد داده‌ها را برای آموزش و ۲۰ درصد را برای آزمایش انتخاب کردیم. از مجموعه دادهٔ آزمایش برای آزمون این الگوریتم و الگوریتم SDF استفاده کردیم. درصد خطای دادهٔ آموزشی تقریباً ۴ درصد است.

نتایج شبیه‌سازی

برای دقت بیشتر در نتایج، آزمایش ۱۰ بار تکرار شد که میانگین نتایج به‌عنوان خروجی ارائه می‌شود. معیار ارزیابی در این کار، مقدار تشخیص[4] و مقدار خطای تشخیص[5] حملات است که با به‌صورت زیر تعریف می‌شوند.

---

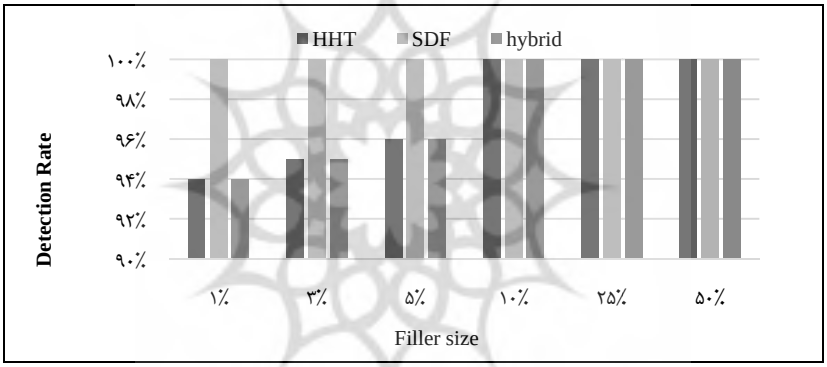
1. Attack Size
2. Filler Size
3. Matlab
4. Detection Rate
5. False Alarm Rate

رابطهٔ ۴)
$$\text{detection rate} = \frac{\text{detection attack profiles}}{\text{all attacks}}$$

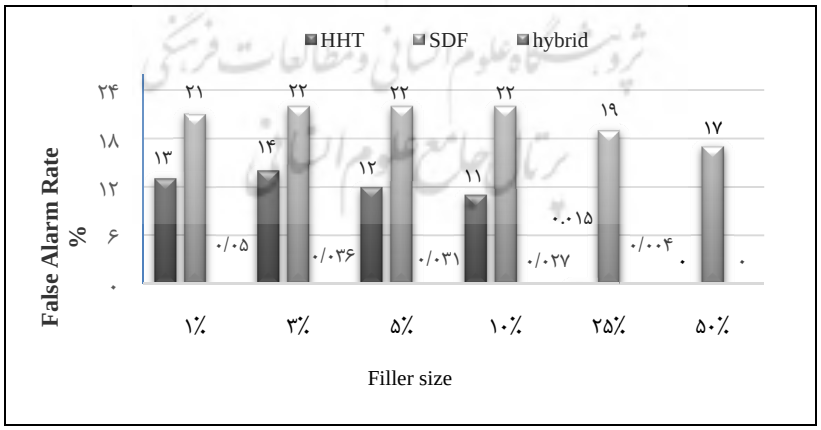
رابطهٔ ۵)
$$\text{false alarm rate} = \frac{\text{false positive}}{\text{genuine profile}}$$

حملات شیلینگ، به دو قصد انجام می‌شوند؛ حملاتی که قصد افزایش اعتبار برخی از آیتم‌های هدف را دارند حملات پوش هستند و حملاتی که قصد کاهش محبوبیت آیتم‌های هدف را دارند، حملات ناک شناخته می‌شوند. در پیاده‌سازی نیز مانند سایر پژوهشگرها فقط حالت پوش را بررسی کردیم، زیرا روش تشخیص را می‌توان به‌راحتی به حالت ناک تغییر داد.

نتیجهٔ حملات تک‌هدفه بر مجموعه دادهٔ ۱۰۰ هزار رکوردی در شکل ۴ ترسیم شده است که قسمت الف، مقدار تشخیص و قسمت ب، مقدار خطای تشخیص را نشان می‌دهد.

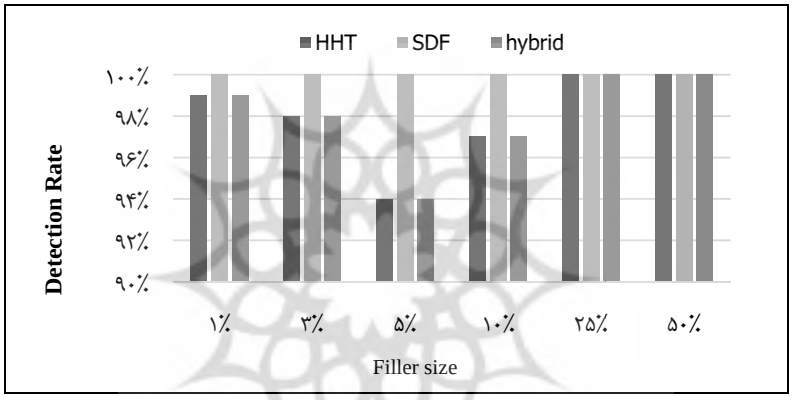


شکل ٤- الف. حملات تک‌هدفه بر مجموعه دادهٔ ۱۰۰ هزار رکوردی (مقدار تشخیص)

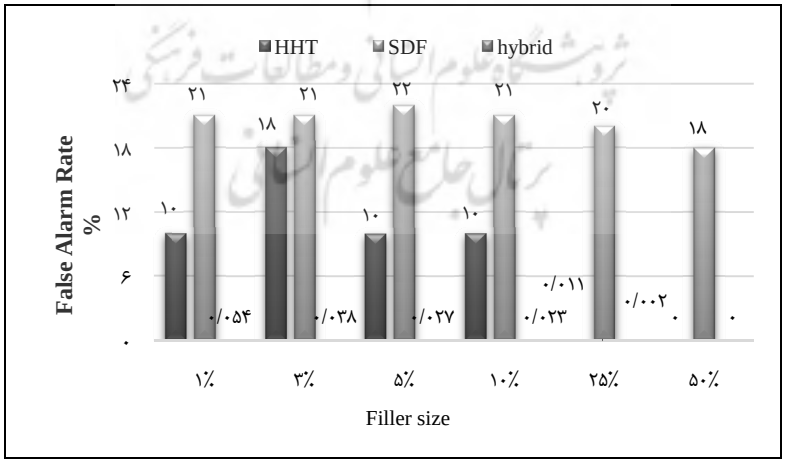


شکل ٤- ب. حملات تک‌هدفه بر مجموعه دادهٔ ۱۰۰ هزار رکوردی (مقدار خطای تشخیص)

نتایج شکل ۴ نشان می‌دهد که الگوریتم SDF همواره مقدار تشخیص حداکثری را ارائه می‌کند و از آنجا که الگوریتم ما از دو مولفهٔ HHT و SDF تشکیل شده است، مقدار تشخیص الگوریتم به مقدار تشخیص الگوریتم HHT وابسته است و با آن نتایج یکسانی دارد. اما مقدار خطای تشخیص ما، به‌خصوص در اندازهٔ حملات پایین بسیار کمتر از دو الگوریتم دیگر است. برای مثال، در اندازهٔ حملات ۱، ۳، ۵ و ۱۰ درصد، مقدار خطا به‌ترتیب تقریباً ۵، ۴ ،۳ و ۳ درصد است، در حالی که در الگوریتم HHT این خطا به‌ترتیب تقریباً ۱۳، ۱۴، ۱۲و ۱۱ درصد و در الگوریتم SDF این خطا بسیار بیشتر است. در شکل ۵ نتیجهٔ حملات سه هدفه بر مجموعه دادهٔ ۱۰۰ هزار رکوردی مشاهده می‌شود.

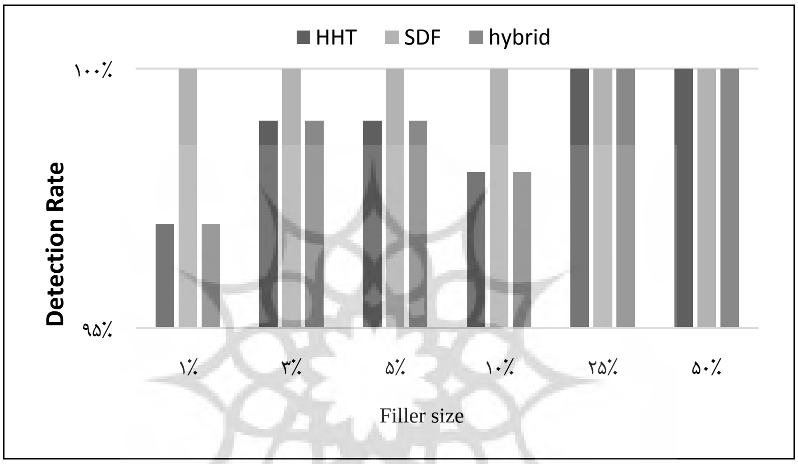


شکل ۵- الف. حملات سه هدفه بر مجموعه دادهٔ ۱۰۰ هزار رکوردی (مقدار تشخیص)

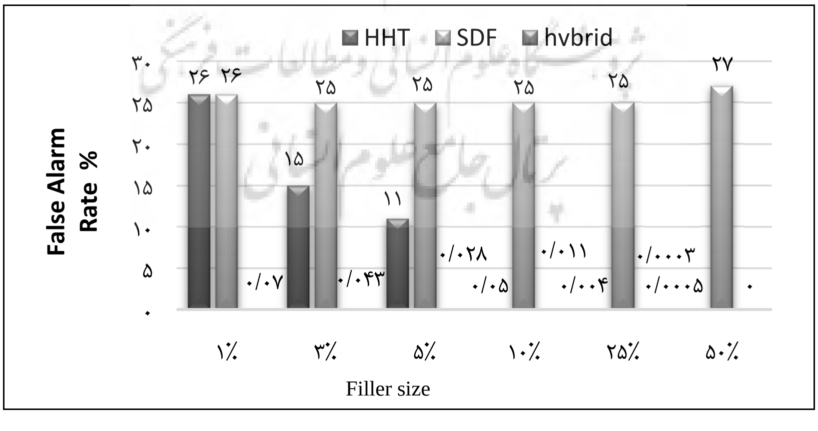


شکل ۵- ب. حملات سه هدفه بر مجموعه دادهٔ ۱۰۰ هزار رکوردی (مقدار خطای تشخیص)

همان‌طور که در شکل ۵ مشاهده می‌شود، حملات سه هدفه، روی مقدار تشخیص الگـوریتم SDF تأثیری ندارد، اما مقدار تشخیص الگوریتم HHT را نسبت بـه حمـلات تـک‌هدفـه کمـی دگرگون کرده است، در نتیجه مقدار تشخیص الگوریتم ما بـاز هـم بـه الگـوریتم HHT وابسـته است. اما در خطای تشخیص حملات سه هدفه، روی دو الگوریتم مؤلف تأثیر نسبتاً زیـادی را در حملات با اندازهٔ پایین گذاشته، در حالی که روی مقدار خطای الگوریتم ما تـأثیر بسـیار نـاچیزی دارد. حملات تک‌هدفه روی دادهٔ ۱ میلیون رکوردی در شکل ۶ دیده می‌شود.

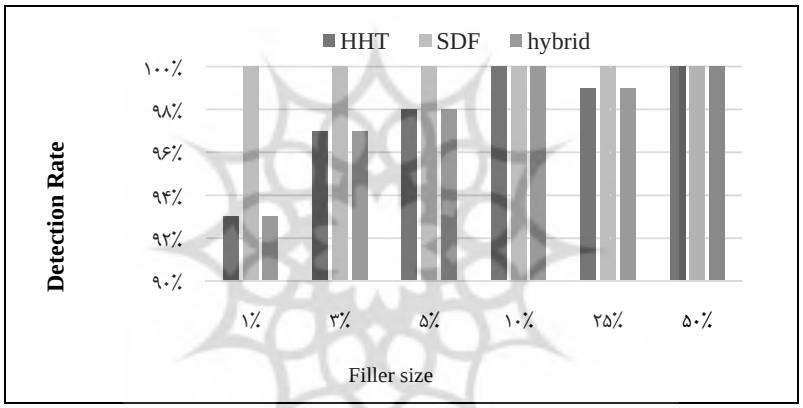


شکل ٦ – الف. حملات تک‌هدفه بر مجموعه دادهٔ ۱ میلیون رکوردی (مقدار تشخیص)

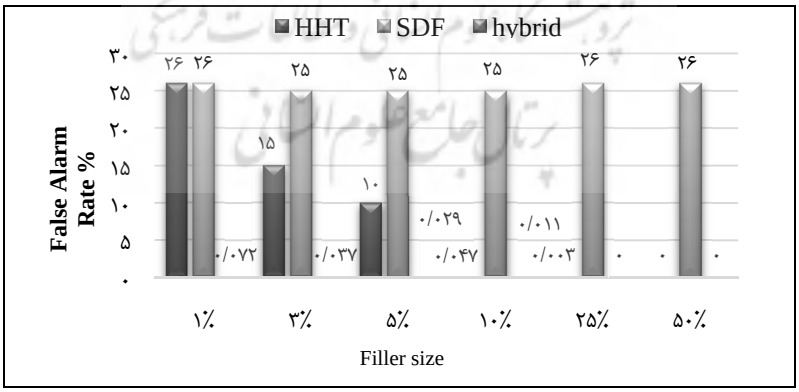


شکل ٦– ب. حملات تک‌هدفه بر مجموعه دادهٔ ۱ میلیون رکوردی (مقدار خطای تشخیص)

نتایج مجموعه دادهٔ یک میلیون رکوردیِ تک‌هدفه نشان می‌دهد مقدار تشخیص الگوریتم ما نسبت به مجموعه دادهٔ قبلی به‌نسبت بهتر شده، زیرا مقدار تشخیص الگوریتم HHT بهبود یافته است. این بهبود به‌دلیل زیاد شدن تعداد نمونه‌های آزمایشی است و مانند گذشته مقدار خطای تشخیص ما در تمام حالت‌ها، به‌خصوص در حملات با اندازهٔ پایین، بسیار کمتر ثبت شده است. اما در الگوریتم‌های دیگر، مقدار خطای تشخیص این داده‌ها نسبت به مقدار خطای تشخیص مجموعه دادهٔ ۱۰۰ هزار رکوردی افزایش یافته است. دلیل نتیجهٔ به‌دست آمده چندین برابر شدن تعداد کاربران در بازه‌های زمانی است. در پیاده‌سازی آخر، حملات سه هدفه روی دادهٔ ۱ میلیون رکوردی در شکل ۷ ترسیم شده است که نتایج مقدار تشخیص کمی تغییر کرده، ولی مقدار خطای تشخیص آن مشابه حملات تک‌هدفه است.



شکل ۷ – الف. حملات سه هدفه بر مجموعه دادهٔ ۱ میلیون رکوردی (مقدار تشخیص)



شکل ۷– ب. حملات سه هدفه بر مجموعه دادهٔ ۱ میلیون رکوردی (مقدار خطای تشخیص)

با توجه به نتایج، درست است که الگوریتم SDF همواره مقدار تشخیص ۱۰۰ درصدی را ارائه می‌دهد، اما همیشه بیشترین خطا را نیز در تشخیص دارد. الگوریتم پیشنهادی ما در مقایسه با الگوریتم HHT، هم در حملات تک‌هدفه و هم در حملات سه هدفه، مقدار تشخیص یکسان دارد، ولی الگوریتم پیشنهادی ما، مقدار خطای تشخیص محسوسی به‌خصوص در اندازهٔ حملات پایین‌تر نسبت به HHT ارائه داده است. این نتیجه از آنجا ناشی می‌شود که الگوریتم کمکی SDF، پروفایل‌های حمله را به‌خوبی استخراج می‌کند و پروفایل‌های سالمی را که به‌عنوان پروفایل حمله شناخته است با الگوریتم HHT، اشتراک کمی دارند.

نتیجه‌گیری و پیشنهادها

سیستم‌های توصیه‌گر یکی از ارکان مهم در سایت‌های تجاری به‌شمار می‌روند، ولی حملات شیلینگ برای استحکام و از بین بردن اعتماد به این سیستم، تهدیدی جدی محسوب می‌شوند. بدین ترتیب برای پیداکردن پروفایل‌های مخرب، الگوریتم‌هایی پدید آمدند و نتایج خوبی به‌دست آورند، اما در حملات با اندازهٔ پایین هنوز به مقدار تشخیص ۱۰۰ درصد و مقدار خطای تشخیص صفر درصد دست نیافتند. برای بهبود این مسئله، در این پژوهش سازوکار جدیدی ارائه دادیم که مقدار تشخیص را در بهترین سطح نگه می‌دارد و مقدار خطای تشخیص حملات را به‌صورت محسوسی کاهش می‌دهد. سازوکار ما شامل دو الگوریتم HHT و SDF است. این دو الگوریتم به‌صورت موازی به پیدا کردن پروفایل‌های حمله اقدام کرده و در ماتریس‌های جداگانه ذخیره می‌کنند. نقطهٔ تلاقی این دو ماتریس، همانند الگوریتم HHT مقدار تشخیص بالایی خواهد داشت و به‌دلیل اینکه پروفایل‌های سالم در این دو ماتریس نسبت به هم تشابه کمی دارند، الگوریتم ما شامل مقدار خطای کمتری خواهد بود که این کار روی دو مجموعه دادهٔ مووی لنز پیاده‌سازی شد و نتایج خوبی دربرداشت.

برای مطالعات بعدی، با توجه به ساختاری که ایجاد کردیم، پیشنهاد می‌شود که این برای بررسی کارایی الگوریتم، ابتدا از لحاظ بی‌درنگ بودن آزمایش شود. سپس الگوریتم را روی مدل‌های حملهٔ دیگر و مجموعه داده‌های مختلف آزمایش کرد تا جامع بودن الگوریتم بررسی شود. بررسی‌ها نشان می‌دهد اخیراً تعداد اندکی از فروشگاه‌های اینترنتی داخل کشور مانند دیجی‌کالا، بامیلو، شیپور و دیجی استیل از سیستم‌های توصیه‌گر برای پیشنهاد کالاهای خود به کاربران بهره گرفته‌اند. نتایج این تحقیق نشان می‌دهد با افزایش آرای کاربران و فراهم شدن شرایط آزمون در سایت‌های داخل کشور، می‌توان از الگوی پیشنهادی این تحقیق برای آنها استفاده کرد.

فهرست منابع

کریمی علویجه، م.، عسکری، ش. و پرسته, س. (۱۳۹۴). فروشگاه اینترنتی هوشمند: سیستم پیشنهاددهندهٔ مبتنی بر تحلیل رفتار کاربران. *فصلنامهٔ علمی ـ پژوهشی مدیریت فناوری اطلاعات*، ۷(۲)، ۴۰۶-۳۸۵.

مطهری نژاد, م س.، ذوالفقارزاده, م. م.، خدنگی, ا. و سعدآبادی، ع. ا. (۱۳۹۵). طراحی مدلی برای بهبود سیستم‌های پیشنهاددهندهٔ بانکی بر اساس پیش‌بینی علایق مشتریان: کاربرد روش‌های داده‌کاوی. *فصلنامهٔ علمی ـ پژوهشی مدیریت فناوری اطلاعات*، ۸(۲)، ۳۹۳-۳۱۴.

Bhaumik, R., Mobasher, B. & Burke, R. (2011). *A clustering approach to unsupervised attack detection in collaborative recommender systems. In Proceedings of the 7th IEEE international conference on data mining.* Las Vegas, NV, USA. 181-187.

Bhaumik, R., Williams, C., Mobasher, B. & Burke, R. (2006). *Securing collaborative filtering against malicious attacks through anomaly detection.* In Proceedings of the 4th Workshop on Intelligent Techniques for Web Personalization., Boston.

Burke, R., Mobasher, B., Williams, C., & Bhaumik, R. (2006). Classification features for attack detection in collaborative recommender systems. *August, In Proceedings of the 12th ACM international conference on Knowledge discovery and data mining.*

Cheng Z, Hurley N. (2009). Effective diverse and obfuscated attacks on model-based recommender systems. *3rd ACM Conf. Recommender system*.

Chirita, P. A., Nejdl, W. & Zamfir, C. (2005). Preventing shilling attacks in online recommender systems. *November, 7th annual ACM international workshop on web information and data management*.

Chung, C. Y., Hsu, P. Y. & Huang, S. H. (2013). βP: A novel approach to filter out malicious rating profiles from recommender systems. *Decision Support Systems*, 55(1), 314-325.

Noh, G. Kang, Y. Oh, H. Kim,C. (2014). Robust Sybil attack defense with information level in online Recommender Systems. *Expert Systems with Applications*, 41(4), 1781-1791.

Gao, M., Tian, R., Wen, J., Xiong, Q., Ling, B. & Yang, L. (2015). Item anomaly detection based on dynamic partition for time series in recommender systems. *PloS one*, 10(8).

Gunes, I., & Polat, H. (2016). Detecting shilling attacks in private environments. *Information Retrieval Journal*, 19(6), 547-572.

Gunes, I., Kaleli, C. Bilge, A. & Polat, H. (2014). Shilling attacks against recommender systems: a comprehensive survey. *Artificial Intelligence Review*, 42 (4), 1-33.

Isinkaye, F. O., Y. O. Folajimi, and B. A. Ojokoh. (2015). Recommendation systems: Principles, methods and evaluation. *Egyptian Informatics Journal*, 16(3), 261-273.

Karimi, M. R. & Askari, SH. & Paraste, S. (2015). Intelligent Online Store: User Behavior Analysis based Recommender System. *Journal of Information Technology Management*, 7(2), 385-406. (*in Persian*)

Mehta, B, Hofmann, T. (2008). A survey of attack-resistant collaborative filtering algorithms. *IEEE Data Eng*, *31*(2), 14-22.

Motaharinejad, M. S. & Zolfagharzadeh, M. M. & Khadangi, E. & Sadabadi, A.A. (2016). Designing a Model for Improving Banking Recommender Systems Based on Predicting Customers' Interests: Application of Data Mining Techniques. *Journal of Information Technology Management*, 8(2), 393-314. (*in Persian*)

Sun, Z., Han, L., Huang, W., Wang, X., Zeng, X., Wang, M. & Yan, H. (2015). Recommender systems based on social networks. *Journal of Systems and Software*, 99, 109-119.

Xia, H., Fang, B., Gao, M., Ma, H., Tang, Y. & Wen, J. (2015). A novel item anomaly detection approach against shilling attacks in collaborative recommendation systems using the dynamic time interval segmentation technique. *Information Sciences*, 306, 150-165.

Zhang, F. & Zhou, Q. (2014). HHT–SVM: An online method for detecting profile injection attacks in collaborative recommender systems. *Knowledge-Based Systems*, 65, 96-105.

Zhang, F. (2015). Robust Analysis of Network based Recommendation Algorithms against Shilling Attacks. *International Journal of Security & Its Applications*, 9(3), 13-24.

Zhang, X.-L., Lee, T., Pitsilis, G. (2013). Securing recommender systems against shilling attacks using social-based clustering. *Journal of Computer Science and Technology*, 28(4), 616–624.

Zhao, Z. D. & Shang, M. S. (2010). User-based collaborative-filtering recommendation algorithms on hadoop. *In Knowledge Discovery and Data Mining. Third International Conference on IEEE*. 478-481.

Zhou, W., Wen, J., Koh, Y. S., Xiong, Q., Gao, M., Dobbie, G., & Alam, S. (2015). Shilling attacks detection in recommender systems based on target item analysis. *PloS one*, 10(7).