شناسایی خوشه‌های حساب‌های جعلی در شبکه‌های اجتماعی اینترنتی

Nima Akhlaghi

نیما اخلاقی - me@nimix3.com

چکیده

حساب‌های جعلی برای کاربران مخرب شبکه‌های اجتماعی آنلاین بهترین ابزار برای ارسال هرزنامه (اسپم)، تقلب کردن یا سوءاستفاده از سیستم هستند. یک عامل مخرب به تنهایی می‌تواند هزارها حساب جعلی بسازد تا عملیات خود را برای افزایش تعداد اعضای قانونی به حداکثر ممکن به انجام رساند. شناسایی این حساب‌های کاربری و اقدام در برابر آنها ، در سریع ترین زمان برای محافظت از اعضای قانونی و حفظ اعتبار شبکه، امری ضروری است. با این‌حال، هر یک از حساب‌های جعلی نیز ممکن است در نگاه نخست به دلیل داشتن نامی واقع نما یا پروفایلی پذیرفتنی قانونی به نظر برسند.

در این مقاله رویکردی سنجش پذیر برای یافتن گروه‌های حساب‌های جعلی که توسط یک عامل واحد ثبت شده‌اند، مطرح می شود. روش اصلی یک خط لوله‌ی نظارت شده‌ی یادگیری ماشین برای طبقه‌بندی یک خوشه‌ی کامل از حساب‌ها به عنوان حساب مخرب یا قانونی است. ویژگی‌های کلیدی استفاده شده در این مدل آمار مربوط به متن‌های تولید شده توسط کاربر مانند نام، آدرس ایمیل، شرکت یا دانشگاه است؛ این ویژگی‌ها هم شامل تعداد تکرار الگوها (مثلا تکرار یک الگوی حرفی یا رقمی در همه ایمیل ها) *در میان* خوشه است و هم شامل مقایسه تعداد تکرار متن‌ها (مثلا کمیاب بودن همه نام ها) در *کل* پایگاه کاربر است.

این مقاله بر اساس پژوهش شکل گرفته است که چهارچوب خود را برای تجزیه و تحلیل داده‌های حساب‌ها در لینکداین (LinkedIn) به کار گرفته است که بر اساس آدرس آی‌پی (IP) ثبت‌نام و تاریخ ثبت‌نام دسته‌بندی شده بودند. این پژوهش در یک مجموعه سنجش انجام شده به AUC 0.98 و در داده‌های آزمایشی خارج از نمونه به AUC 0.95 دست یافت. این مدل به صورت محصول درآمده است و از زمان استقرار، بیش از 250,000 حساب جعلی را شناسایی کرده است.

دسته‌بندی‌ها و توصیف‌کننده‌های موضوع

H.3.3 (ذخیره و بازیابی اطلاعات): جستجو و بازیابی اطلاعات – اسپم؛ 1.2.6 (هوش مصنوعی): یادگیری

کلید واژه ها: تشخیص اسپم، پروفایل‌های جعلی، یادگیری ماشین، داده کاوی، خوشه‌بندی، دسته‌بندی

معرفی

امروزه مردم سراسر دنیا برای به اشتراک گذاشتن دانش، نظرها و تجربه ها، جستجوی اطلاعات و منابع، و گسترش روابط شخصی بر شبکه‌های اجتماعی آنلاین (OSN) تکیه دارند. با این‌حال، همان ویژگی‌هایی که شبکه‌های اجتماعی آنلاین را برای مردم عادی ارزشمند می‌سازد، آن‌ها را اهدافی برای سوءاستفاده های گوناگون نیز قرار می‌دهد. برای مثال مخاطبان زیادی در یک پلتفرم واحد، هدف اصلی ارسال کنندگان اسپم (اسپمر) و کلاهبرداران هستند، و قابل اعتماد بودن پلتفرم ممکن است باعث شود این مخاطب‌ها بیشتر در معرض افتادن به دام

کلاهبرداران قرار گیرند [2]. جنبه‌های «بازی‌وار سازی[1]» یک سایت (برای مثال، شمارنده‌های «پسندیدن[2]» یا «دنبال کردن[3]») به ربات‌هایی که مشغول عملیات مصنوعی هستند داده می‌شوند تا به طور غیر قانونی محصولات یا خدمات را تبلیغ کنند [33 و 36]. و از جزئیات رابطه‌ها ممکن است برای استخراج اطلاعات ارزشمند تجاری استفاده شود [19]. و همین فراوانی اطلاعات اعضا توجه کسانی (اسکراپرهایی[4]) را که میخواهند پایگاه داده‌ی خود را با اطلاعات افراد واقعی راه‌اندازی کنند، جلب می‌کند [30]. بر اساس آماری که شرکت امنیتی کلودمارک (Cloudmark) ارائه کرد، بین 20 تا 40 درصد حساب‌های فیس‌بوک ممکن است پروفایل جعلی باشند [11]؛ توئیتر و لینکداین نیز با درجات متفاوتی از حساب‌های جعلی روبه‌رو هستند [13 و 21]. گذشته از انگیزه‌های خاص برای ایجاد حساب‌های جعلی، وجود تعداد زیاد حساب‌های جعلی می‌تواند اعتبار شبکه‌های اجتماعی اینترنتی را برای کاربران قانونی کاهش دهد. برای مثال، اگر کاربران در اعتبار اطلاعات پروفایل تردید کنند، این حساب‌های جعلی می‌توانند اعتبار شبکه را تضعیف کنند [20]. این حساب‌ها می‌توانند تأثیر منفی بر درآمد تبلیغات شبکه نیز داشته باشند، زیرا اگر بسیاری از کاربران افراد واقعی نباشند، تبلیغ‌کنندگان ممکن است در مورد نرخی که می‌پردازند بپرسند تا به تعداد مشخصی از کاربران دست یابند.

هنوز هم تشخیص و متوقف کردن حساب‌های جعلی دشوار است. یک شبکه‌ی اجتماعی اینترنتی در مقیاس بزرگ ممکن است میلیون‌ها کاربر فعال و میلیاردها فعالیت کاربری داشته باشد که حساب‌های جعلی تنها درصد کمی از آن است. با توجه به این عدم تعادل، نرخ‌های مثبت کاذب باید بسیار پایین نگه داشته شوند تا از مسدود شدن بسیاری از اعضای واقعی پیشگیری شود. در عین حال برخی از حساب‌های جعلی ممکن است الگوهای روشنی از اتوماتیک بودن را نشان دهند، بسیاری از آن‌ها طوری طراحی شده‌اند که از افراد واقعی تشخیص پذیر نیستند. اقدامات امنیتی مانند کدهای مچ‌گیر (Captcha) و تأیید تلفنی از طریق پیام کوتاه برای تحقیق درمورد حساب‌های مشکوک طراحی شده‌اند و از این رو مانع ایجاد حساب‌های جعلی می‌شوند. با این حال، شبکه‌های اجتماعی اینترنتی هنوز هم باید زیرمجموعه‌ای از حساب‌ها را برای بررسی انتخاب کنند (چرا که بررسی تمامی حساب‌ها ممکن است اصطکاک غیرضروری در استفاده‌ی کاربران واقعی ایجاد کند)، و اسپمر در رویارویی با این چالش ، می‌تواند آن را با استفاده از مزارع کد مچ‌گیر یا مزارع سیم کارت حل کند [7]، یا ممکن است با بهره گیری از این بازخورد یاد بگیرد چگونه از طبقه‌بندی حساب‌های جعلی دوری کند [16].

در حالی که تحقیقات زیادی در زمینه‌ی شناسایی حساب‌های جعلی انجام شده است (بخش 6 را ببینید)، که شامل برخی الگوریتم‌های یادگیری ماشین استفاده شده است، این موضوع هنوز نارسایی‌هایی دارد، مانند:

1. هیچ‌یک از رویکردهای موجود *خوشه‌های* حساب‌های جعلی را به سرعت تشخیص نمی‌دهند. اغلب الگوریتم‌های منتشر شده برای تشخیص حساب‌های جعلی در مورد هر حساب پیش‌بینی می‌کنند [1، 26، 31 و 36]. از آنجا که شبکه‌های اجتماعی اینترنتی در

1. Gamification
2. Like
3. Follow
4. Scraper

مقیاس بزرگ ممکن است صدها هزار حساب جدید در روز ثبت کند و عامل مخرب تلاش کند در این مقیاس حساب ایجاد کند، بسیار بهتر است یک الگوریتم تشخیص در سطح خوشه داشته باشیم که بتواند تشخیص سریع و سنجش‌پذیر انجام دهد و همه‌ی حساب‌های یک خوشه را یک‌باره بگیرد.

2. هیچ‌یک از رویکردهای موجود برای تشخیص حساب‌های جعلی و اقدام در برابر آن‌ها قبل از ارتباط آن‌ها با اعضای قانونی، خرابکاری و ارسال اسپم، طراحی نشده است. الگوریتم‌های موجود برای تشخیص حساب‌های جعلی به طور کلی بر مبنای تجزیه و تحلیل فعالیت‌های کاربر و یا ارتباطات شبکه‌ی اجتماعی است [10، 17، 27، 38 و 39]، این به این معناست که حساب‌های جعلی باید اجازه داشته باشند برای مدتی در شبکه بمانند تا ارتباطات آن‌ها توسعه یابد و اطلاعات کافی از فعالیت آن‌ها جمع‌آوری شود. ما عملاً می‌خواهیم حساب‌های جعلی را با بیشترین سرعت ممکن پس از ثبت نام آن‌ها بگیریم تا از تعامل آن‌ها با کاربران واقعی جلوگیری شود. این یک چالش ایجاد می‌کند، چرا که ما تنها برخی اطلاعات پایه را داریم که در طول جریان ثبت نام به‌دست آمده است. از این رو الگوریتمی که بتواند بر اساس اطلاعات بسیار محدود پروفایل بیشترین الگوهایی را که ممکن است بگیرد، یک نیاز فوری است.

1.1 مشارکت ما

در این مقاله، یک رویکرد یادگیری ماشین سنجش‌پذیر و حساس به زمان برای یافتن گروه‌های حساب‌های جعلی ثبت‌شده توسط یک عامل معرفی می‌شود. این رویکرد به چالش‌هایی که در زیر توضیح داده شده‌اند پاسخ می‌دهد:

1. اولین مرحله در خط لوله‌ی این پژوهش گروه‌بندی حساب‌ها با صورت خوشه است، و الگوریتم یادگیری ماشین آن ویژگی‌های در سطح خوشه را انحصاراً به عنوان ورودی می‌گیرد. تمام ویژگی‌ها طوری طراحی شده‌اند که به جای حساب های انفرادی کل خوشه را توصیف کنند، و طبقه‌بندی حاصل در کل خوشه‌هاست. این رویکرد در شبکه‌های اجتماعی اینترنتی که تعداد زیادی ثبت نام روزانه دارند، سنجش‌پذیر است.
2. این الگوریتم تنها از ویژگی‌هایی استفاده می‌کند که در زمان ثبت نام یا مدت کوتاهی پس از آن در دسترس هستند. به ویژه، به نمودار اطلاعات یا داده‌های فعالیت نیازی نیست. با این حال، از آن‌جا که داده‌های خام در زمان ثبت نام محدود است، باید به صورت هوشمندانه ویژگی‌هایی را ایجاد شود که تشخیص خوشه‌های خوب را از خوشه‌های بد امکان‌پذیر می‌سازد. در بخش 4 سه دسته از ویژگی‌ها شرح داده می‌شود که دستیابی به این هدف را امکان‌پذیر می‌سازد. همچنین *الگوریتم‌های کدگذاری الگوی* عمومی پیشنهاد شده که اجازه می‌دهد متن‌های تولید شده توسط کاربر در فضای کوچکی جمع شود تا محاسبه‌ی ویژگی‌های آماری ممکن شود.

این پژوهش چارچوب خود را به عنوان یک خط لوله‌ی یادگیری ماشین آفلاین در زبان هدوپ (Hadoop) پیاده‌سازی کرده است. این خط لوله از سه جزء تشکیل شده است: *سازنده‌ی خوشه* که خوشه‌ها را برای امتیازدهی تولید می‌کند؛ *ویژگی‌ساز پروفایل* که ویژگی‌هایی برای استفاده در مدل‌سازی استخراج می‌کند؛ و *امتیازدهنده حساب* که مدل‌های یادگیری ماشین را آموزش می‌دهد و مدل‌ها را از نظر داده‌های ورودی جدید ارزیابی می‌کند. جزئیات این خط لوله در بخش 3 آمده است.

1.2 نتایج تجربی

این پژوهش رویکرد خود را در داده‌های حساب لینکداین ارزیابی کرد. حدود 275,000 حساب ثبت‌شده را در یک دوره‌ی شش ماهه به عنوان نمونه برای داده‌های آموزشی بررسی کرد که 55٪ آن‌ها به عنوان حساب جعلی یا اسپم توسط تیم امنیتی لینکداین برچسب‌گذاری شده بود[5]. برچسب‌های در سطح حساب کاربری را، به برچسب‌های در سطح خوشه برای آموزش دادن دسته‌بندهای خود گروه‌بندی کرد.

مدل‌ها را با استفاده از روش جنگل تصادفی، رگرسیون منطقی و دسته‌بندهای ماشین بردار پشتیبانی آموزش داد. عملکرد دسته‌بندها را با آزمون شکاف 80–20 درون‌نمونه‌ای و برون‌نمونه‌ای با مجموعه داده‌های جدید ارزیابی کرد. آزمون دوم تخمین بهتری از عملکرد واقعی است، چرا که مدل‌ها با داده‌هایی از گذشته آموزش داده شده و بر روی داده‌های فعلی اجرا می‌شوند.

برای اندازه‌گیری عملکرد طبقه‌بندی‌ها، AUC (محدوده‌ی زیر منحنی ROC) محاسبه شده و با دقت 95٪ فراخوانی می‌شود. در عمل نرخ دقت مطلوب و آستانه‌ی طبقه‌بندی ممکن است بسته به نیازهای تجاری و هزینه‌ی نسبی مثبت کاذب و منفی کاذب، بالاتر یا پایین‌تر باشد. سپس دریافتند که الگوریتم جنگل تصادفی بهترین نتایج را برای همه‌ی معیارها ارائه کرده است. در مجموعه آزمون برگزار شده، مدل جنگل تصادفی AUC 0:98 را تولید کرده و 0:90 را با دقت 95٪ فراخوانی کرده است. هنگام اجرا روی داده‌های آزمایشی خارج از نمونه، مدل جنگل تصادفی با AUC 0:95 و فراخوانی 0:72 با دقت 95٪ باز هم بهتر عمل کرده است.

1.3 سازماندهی مقاله

در بخش 2 مروری بر روش‌های یادگیری نظارت‌شده همراه با معیارهایی برای ارزیابی مدل ارائه می‌شود که در مقاله از آن استفاده شده است. در بخش 3 خط لوله‌ی یادگیری ماشین توصیف می‌شود که برای پیاده‌سازی سیستم استفاده شده است، و در بخش 4 رویکرد پژوهش نسبت به مهندسی ویژگی‌ها توضیح داده می‌شود. سپس در بخش 5 نتایج آزمایش بر روی یکی از نمونه‌های رویکرد معرفی شده ارائه می‌شود که عملکرد داده‌های آزمایشی و همچنین نتایج بر روی داده‌های فعلی لینکداین را توصیف می‌کند. در مورد تحقیقات مرتبط در بخش 6 بحث می‌شود و تحقیقات آینده در بخش 7 مطرح میگردد.

5. به‌خاطر داشته باشید که این نمونه نمایش‌دهنده‌ی پایگاه اعضای لینکداین نیست، بلکه نمونه‌ای از حساب‌هایی است که به دلایلی به عنوان نمونه‌های مشکوک علامت‌گذاری شده‌اند.

1. روش‌های آموزش

1.2 روش‌های یادگیری نظارت‌شده

در طول آموزش مدل، هدف ساخت و انتخاب زیرمجموعه‌هایی از ویژگی‌هاست که برای ایجاد پیش‌بینی‌کننده‌ی خوب مفید است. در آزمایش-های انجام شده سه روش رگرسیون زیر در نظر گرفته شد: رگرسیون منطقی با تنظیم L1 [34]، ماشین بردار پشتیبانی با هسته‌ی تابع پایه شعاعی [15] و جنگل تصادفی [4]، یک مدل یادگیری گروهی غیرخطی مبتنی بر درخت.

رگرسیون منطقی. با توجه به مجموعه‌ی $S = \{(x^{(i)}, y^{(i)})\}$ از m نمونه‌ی آموزشی با $x^{(i)}$ به عنوان ویژگی ورودی و $y^i \in \{0, 1\}$ به عنوان برچسب‌ها است، رگرسیون منطقی می‌تواند به صورت زیر مدل‌سازی شود:

$$p(y = 1|x, \theta) = \frac{1}{1+\exp(-\theta^T x)} \quad (1)$$

که $\theta \in R^n$ پارامترهای مدل است.

بدون تنظیم، رگرسیون منطقی تلاش می‌کند پارامترها را با استفاده از معیار حداکثر احتمال پیدا کند، در حالی که با تنظیم کردن، هدف این است که اختلاف بین متغیرهای مناسب در مدل کنترل شود و متغیرهای کمتری در مدل انتخاب شوند. در این پژوهش از L_1 برای تنظیم مدل رگرسیون منطقی استفاده شده است. این روش توزیع احتمال برچسب دسته‌ی y را با توجه به بردار ویژگی x حداکثر می‌کند، و همچنین تعداد ویژگی‌های غیر مرتبط با استفاده از شرایط اجباری برای محدود کردن ضرایب θ در قانون L_1 کاهش می‌دهد. پارامترهای مدل $\theta \in R^n$ به صورت زیر محاسبه می‌شوند:

$$\arg\min_{\theta} \sum_{i=1}^{m} -\log p\left(y^{(i)}|x^{(i)}, \theta\right) + \beta|\theta|_1 \quad (2)$$

در این فرمول، β پارامتر تنظیم‌کردن است و با استفاده از روش اعتبارسنجی متقابل به طور بهینه انتخاب می‌شود.

ماشین بردار پشتیبانی. دومین الگوریتم یادگیری در نظر گرفته شده، ماشین بردار پشتیبانی (SVM) است [3، 9، 29 و 35]. الگوریتم ماشین بردار پشتیبانی یک سطح بالای بهینه را به عنوان تابع تصمیم‌گیری در فضایی با ابعاد بزرگ جستجو می‌کند.

مجموعه داده‌ی آموزشی باز هم از جفت‌های $\left(x^{(i)}, y^{(i)}\right) \in R^n \times \{0, 1\}$ تشکیل شده است. در مطالعات ما، از آن‌جا که در این مطالعات هدف استفاده از یک دسته‌بند غیر خطی است، از SVM با هسته‌ی تابع پایه‌ی شعاعی (RBF) در آموزش استفاده می‌شود. هسته‌ی RBF می‌تواند به صورت $k(x, x') = \exp(-r||x - x'||^2)$ فرموله شود. بیش‌پارامتر r *پهنای باند* هسته نامیده می‌شود و بر اساس نتایج اعتبارسنجی متقابل تنظیم می‌شود.

در اصل، الگوریتم SVM ابتدا x را از طریق تابع ψ به یک فضا با بعد بیشتر نگاشت می‌کند، سپس سطح بالاتر H را در فضایی با بعد بیشتر می‌یابد که فاصله‌ی بین نقاط مجموعه‌ی $\psi(x_i)$ و H را به حداکثر می‌رساند. اگر این سطح بالاتر $\langle w, X\rangle = b$ (که X فضای با بعد بیشتر است) باشد، بنابراین تابع تصمیم‌گیری به صورت $f(x) = \langle w, \psi(x)\rangle$ است. علامت f(x) برچسب کلاس x را می‌دهد. در عمل،

تابع ψ ضمنی است و تمامی محاسبات با هسته k انجام می‌شود. ما در آزمایشات انجام شده یک مدل احتمالی برای طبقه‌بندی بر اساس بسته‌ی R از "e1071" اتخاذ شده است [25]. مقادیر دسته‌بندهای دودویی با استفاده از حداکثر احتمال بر یک توزیع منطقی منطبق شده‌اند تا امتیازهای عددی نشان‌دهنده‌ی احتمال را تولید کنند. در حالیکه امکان داشت به آسانی از نمرات SVM خام برای دسته‌بندی استفاده شود، نگاشت این امتیازات احتمالا اجازه می‌دهد نتایج SVM را با سایر مدل‌هایی که تخمین احتمال را ایجاد می‌کنند، مقایسه شوند.

جنگل تصادفی: الگوریتم جنگل تصادفی [4] یک رویکرد گروهی است که بسیاری از دسته‌بندهای ضعیف را ترکیب می‌کند (درخت تصمیم‌گیری) تا یک دسته‌بند قوی تشکیل دهد (جنگل تصادفی). در هر درخت تصمیم‌گیری، ابتدا با جایگزینی از مجموعه‌ی آموزشی اصلی نمونه‌سازی می‌شود تا یک مجموعه‌ی آموزشی جدید با همان اندازه به دست آید. سپس در هر گره‌ی درخت تصمیم‌گیری، m ویژگی به صورت تصادفی انتخاب می‌شود و درخت تصمیم‌گیری بر اساس بهترین تجزیه‌ی ممکن از میان آن m ویژگی تقسیم می‌گردد. مقدار m باید طوری انتخاب شود که قدرت درخت‌های فردی (m بیشتر بهتر است) در برابر همبستگی بین درختان (m کمتر بهتر است) متعادل باشد. اکنون با توجه به نمونه‌ی جدید، مدل حاصل آن را با اجرای نمونه در همه‌ی درخت‌ها و سپس ترکیب نتایج، امتیازدهی می‌کند. در صورتی که مشکل دسته‌بندی دودویی وجود داشته باشد، امتیاز به سادگی درصد درختانی خواهد بود که نتیجه‌ی مثبتی در نمونه دارند.

1.4 معیارهای ارزیابی

هر سه دسته‌بندی امتیازاتی با اعداد واقعی تولید می‌کنند که می‌تواند برای مرتب کردن نمونه‌ها در مجموعه‌ی آزمایشی استفاده شود. برای اندازه‌گیری عملکرد دسته‌بندها، AUC (محدوده‌ی زیر منحنی ROC)، دقت و فراخوانی محاسبه می‌شود. می‌توان هر اندازه را هم در سطح خوشه و هم در سطح حساب کاربری محاسبه کرد، که به هر حساب امتیازی اختصاص داده شده که توسط دسته‌بند برای خوشه‌ی والد آن تولید شده است.

محدوده‌ی زیر منحنی ویژگی‌های عملکردی دریافت‌کننده (AUC) به طور مشترک در مقایسه‌ی مدل استفاده می‌شود و می‌تواند تفسیر شود به عنوان احتمال اینکه دسته‌بند به نمونه‌های مثبت تصادفی نسبت به نمونه‌های منفی تصادفی امتیاز بالاتری بدهد. مدلی با AUC بیشتر، مدل بهتری در نظر گرفته می‌شود. مزایای AUC به عنوان یک اندازه این است که به انتخاب آستانه‌ای برای تعیین برچسب‌ها برای امتیازدهی نیاز ندارد و اینکه مستقل از تعلق به کلاس در مجموعه‌ی آزمایشی است.

دقت و فراخوانی معیارهای شناخته شده‌ای برای دسته‌بندی دودویی هستند. در این کاربرد، دقت کسری از پیش‌بینی حساب‌های جعلی است که واقعاً جعلی هستند می‌باشد، در حالی‌که فراخوانی کسری از حساب‌های جعلی در جامعه است که توسط مدل گرفته شده‌اند. در دسته‌بندی که امتیاز یا احتمال تولید می‌کند، دقت و فراخوانی می‌تواند برای هر آستانه‌ی امتیاز با توجه به منحنی پارامتری محاسبه شود. از آنجا که مثبت کاذب در مدل حساب‌های جعلی بسیار پرهزینه است، معیار انتخاب برای ارزیابی مدل نرخ فراخوانی در آستانه‌ای است که دقت 95٪ دارد. (نرخ 95٪ صرفاً برای شروع است؛ در عمل هدف دقت بسیار بیشتری است.)

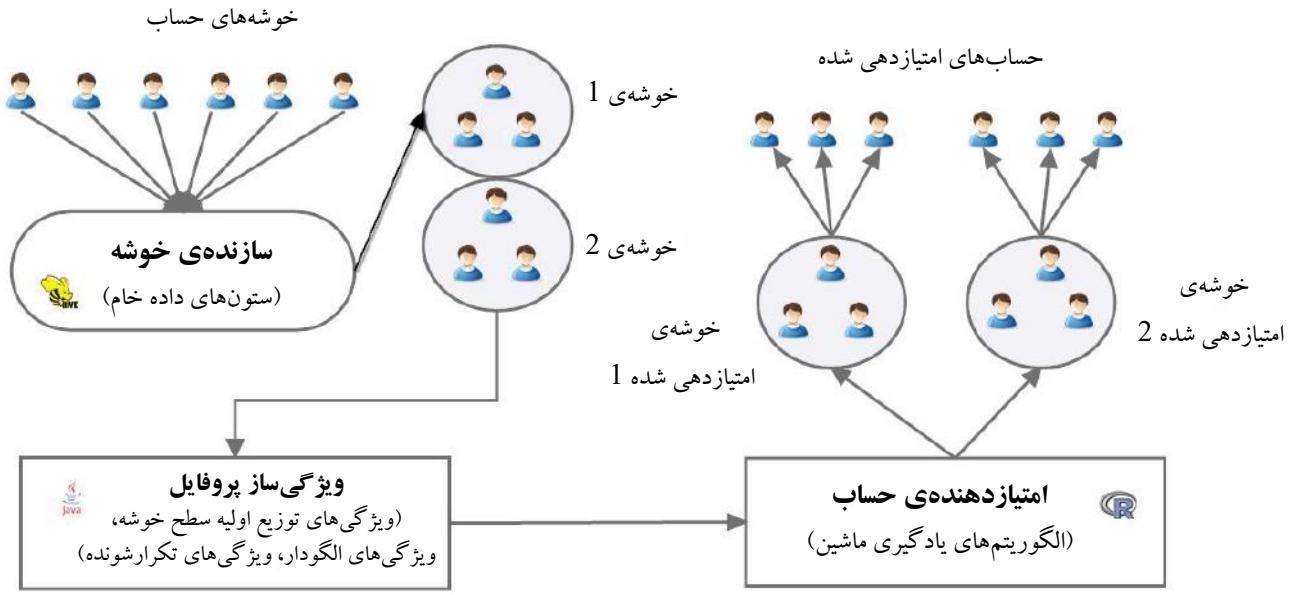
2. خط لوله یادگیری ماشین

برای سنجش‌پذیر کردن سیستم تشخیص حساب‌های جعلی معرفی شده، یک خط لوله‌ی یادگیری ماشین عملی طراحی و پیاده‌سازی شده است که شامل زنجیره‌ای از مراحل پیش‌پردازش داده، استخراج ویژگی‌ها، پیش‌بینی و تأیید است. این خط لوله شامل سه جزء مهم است که در زیر توصیف می‌کنیم و در شکل 1 آمده است.

3.1 سازنده‌ی خوشه

سازنده‌ی خوشه، چنان‌که از نامش پیداست، فهرست خام حساب‌ها را می‌گیرد و خوشه‌های حساب‌ها را همراه با ویژگی‌های خام آن‌ها می‌سازد. این ماژول پارامترهای تعیین‌شده توسط کاربر را برای (1) حداقل و حداکثر اندازه‌ی خوشه، (2) مدت زمان حساب‌های ثبت‌نام شده (برای مثال 24 ساعت گذشته، هفته‌ی اخیر)، و (3) معیار خوشه‌بندی، می‌گیرد. معیار خوشه‌بندی می‌تواند به سادگی گروه‌بندی کردن تمام حساب‌هایی باشد که یک ویژگی مشترک مانند آدرس آی‌پی دارند یا یک الگوریتم پیچیده‌تر خوشه‌بندی مانند k می‌نیمم باشد. وقتی خوشه‌های اولیه ساخته می‌شوند، معیار تعریف‌شده توسط کاربر می‌تواند برای فیلتر کردن برخی از خوشه‌ها اضافه شود تا خوشه‌هایی که احتمال دارد مشکوک نباشند یا ممکن است باعث مثبت کاذب بالایی شوند، حذف شود. برای مثال، یک نفر ممکن است بخواهد حساب‌هایی را حذف کند که از فضای آی‌پی شرکت شبکه‌ی اجتماعی اینترنتی ثبت نام کرده‌اند، چون این حساب‌ها احتمالاً حساب‌های آزمایشی هستند و نباید محدود شوند. سازنده‌ی خوشه جداول پروفایل خام اعضا را به عنوان ورودی می‌گیرد و جدول حساب‌ها همراه با ویژگی‌هایی که برای مهندسی ویژگی‌ها لازم است مانند نام اعضا، شرکت و تحصیلات را تولید می‌کند. هر سطر این جدول یک حساب را نشان می‌دهد و شامل یک مقدار انحصاری «شناسه‌ی خوشه» به خوشه‌ی آن حساب است. این جدول به عنوان ورودی ویژگی‌ساز پروفایل استفاده می‌شود.

در مرحله‌ی آموزش، سازنده‌ی خوشه باید از برچسب‌های سطح حساب نیز استفاده کند تا هر حساب را به عنوان حساب واقعی یا جعلی برچسب‌گذاری کند. در بیشتر خوشه‌ها یا بیشتر حساب‌ها به عنوان حساب جعلی برچسب‌گذاری شده و یا هیچ‌یک از حساب‌ها به عنوان جعلی برچسب‌گذاری نشده است، با این‌حال در حالت کلی تعداد کمی خوشه وجود خواهد داشت که چند حساب در هر دو گروه داشته باشد. بنابراین برای محاسبه‌ی برچسب‌های خوشه، آستانه‌ی X را طوری انتخاب می‌کنیم که خوشه‌هایی که کمتر از X درصد حساب جعلی برچسب‌خورده دارند، به عنوان حساب واقعی برچسب‌گذاری شده و اگر بیشتر از X درصد حساب جعلی وجود داشته باشد، به عنوان حساب جعلی برچسب‌گذاری می‌شوند. انتخاب بهینه‌ی X بستگی به دقت یا فراخوانی متقابل دارد (یعنی مقادیر بالاتر X دقت هزینه‌ی فراخوانی را افزایش می‌دهد). با این حال، چنان‌که در بخش 5.2 در زیر بحث خواهد شد، در عمل دریافتیم که این مدل نسبتاً به این انتخاب حساس نیست.



شکل 1: خط لوله ای که رویکرد تشخیص خوشه‌های حساب‌های جعلی را پیاده‌سازی می‌کند. ما حساب‌ها را به صورت خوشه جمع‌آوری می‌شوند، ویژگی‌ها را استخراج می‌کنیم، مدل را آموزش داده یا ارزیابی می‌کنیم و امتیازها را به حساب‌ها در هر خوشه اختصاص می‌دهیم.

3.2 ویژگی‌ساز پروفایل

ویژگی‌ساز پروفایل جزء کلیدی این خط لوله است. هدف آن تبدیل داده‌های خام هر خوشه (یعنی داده‌های همه‌ی حساب‌های انفرادی خوشه) به یک بردار عددی واحد و نمایش‌دهنده‌ی آن خوشه است که می‌تواند در الگوریتم یادگیری ماشین استفاده شود. این ویژگی‌ساز به صورت مجموعه‌ای از توابع پیاده‌سازی شده است که طراحی شده‌اند تا بیشترین اطلاعاتی را که ممکن است از ویژگی‌های خام بگیرند و بتوانند بین خوشه‌ها با حساب‌های جعلی و خوشه‌ها با حساب‌های قانونی تفاوت قائل شوند.

ویژگی‌های استخراج‌شده را می‌توان عمدتاً به سه دسته تقسیم کرد که در اینجا در سطح بالایی توضیح خواهیم داد؛ جزئیات بیشتر را می‌توان در فصل 4 پیدا کرد.

1. **ویژگی‌های توزیع اولیه.** در هر خوشه، اقدامات آماری پایه برای هر ستون (برای مثال نام شرکت) انجام خواهد شد. مثال‌ها شامل میانه یا یک چهارم ویژگی‌های عددی یا تعداد مقادیر یکتا برای ویژگی‌های متنی هستند.
2. **ویژگی‌های الگودار.** در این پژوهش «الگوریتم‌های کدگذاری شده‌ی الگو» به گونه ای طراحی شده که متن‌های تولید شده توسط کاربر را به یک فضای دسته‌بندی شده‌ی کوچک‌تر نگاشت می‌کند. سپس ویژگی‌های توزیع پایه از این متغیرهای دسته‌ای گرفته می‌شود. این ویژگی‌ها برای تشخیص آن کاربران مخرب (به ویژه ربات‌ها) طراحی شده‌اند که در ثبت حساب‌های خود از یک الگو پیروی می‌کنند.
3. **ویژگی‌های تکراری.** برای هر مقدار ویژگی، تعداد تکرار آن مقدار در کل پایگاه‌داده‌ی حساب‌ها محاسبه می‌شود. سپس ویژگی‌های توزیع پایه در آن تکرار محاسبه می‌گردد. به طور کلی انتظار می‌رود خوشه‌های حساب‌های واقعی، چندین داده با تکرار

زیاد و چندین داده با تکرار کم داشته باشند، درحالی‌که ربات‌ها یا کاربران مخرب اختلاف کمتری در تکرارها نشان می‌دهند؛ یعنی فقط از نام‌های رایج یا فقط از نام‌های نادر استفاده می‌کنند.

3.3 امتیازدهنده‌ی حساب

تابع امتیازدهنده‌ی حساب برای آموزش دادن مدل‌ها و ارزیابی آن‌ها در داده‌هایی است که قبلاً مشاهده نشده‌اند. امتیازدهنده‌ی حساب، خروجی ویژگی‌ساز پروفایل را به عنوان ورودی می‌گیرد، یعنی یک عامل عددی برای هر خوشه. الگوریتم خاص یادگیری که استفاده شده است، قابل تغییر توسط کاربر است؛ در آزمایش های انجام شده رگرسیون منطقی، جنگل‌های تصادفی و ماشین‌های بردار پشتیبانی را در نظر گرفته شده‌اند. در «حالت آموزش»، به امتیازدهنده‌ی حساب، یک مجموعه‌ی برچسب‌گذاری شده از داده‌های آموزشی داده می‌شود و توصیف مدل به همراه معیارهای ارزیابی را تولید می‌کند که می‌تواند برای مقایسه با مدل‌های دیگر استفاده شود. در «حالت ارزیابی»، به امتیازدهنده‌ی حساب توصیف مدل‌ها و یک بردار ورودی از ویژگی‌های خوشه داده می‌شود و یک امتیاز برای خوشه تولید می‌شود که نشان‌دهنده‌ی این احتمال است که آن خوشه از حساب‌های جعلی تشکیل شده باشد.

بر اساس امتیاز خوشه، سه اقدام می‌تواند بر روی حساب‌های آن خوشه انجام شود: محدودسازی خودکار (اگر احتمال جعلی بودن حساب بالا باشد)، بررسی دستی (اگر نتایج قابل نتیجه‌گیری نباشد)، یا هیچ اقدامی صورت نگیرد (اگر احتمال جعلی بودن حساب پایین باشد). آستانه‌ی دقیق انتخاب بین این سه اقدام طوری تنظیم می‌شود که کمترین منفی کاذب رخ دهد و به مرورکنندگان انسانی ترکیبی از حساب‌های خوب و بد را بدهد.

4. مهندسی ویژگی‌ها

کیفیت ویژگی‌های عددی تولیدشده توسط ویژگی‌ساز پروفایل مهم‌ترین عامل کارایی دسته‌بندهاست. اکنون این فرآیند را با جزئیات بیشتر شرح می‌دهیم.

4.1 ویژگی‌های توزیع پایه

نخست بررسی دستی خوشه‌های حساب‌های جعلی در مجموعه داده‌ی لینکداین انجام شد (برای جزئیات بیشتر بخش 5 را ببینید) که قبلاً به عنوان حساب جعلی شناسایی و برچسب‌گذاری شده بود. دیده شد که حساب‌های یک خوشه‌ی بزرگ عموماً الگوهایی را در داده‌های وارد شده توسط کاربر نشان می‌دهد، مانند نام، شرکت یا تحصیلات. گاهی اوقات این الگوها ممکن است واضح باشد؛ برای مثال، همه‌ی حساب‌ها ممکن است از یک متن یکسان برای شرح جایگاه شغلی فعلی استفاده کنند. چنین الگویی می‌تواند توسط چیزی که ما آن را *ویژگی توزیع پایه* می‌خوانیم، گرفته شود. در این حالت، این ویژگی تعداد شرح شغل‌های انحصاری خواهد بود. ویژگی‌های توزیع پایه که ما در نظر می‌گیریم به صورت زیر خواهد بود:

- برای ویژگی‌های عددی:

- حداقل، حداکثر و یک چهارم
- میانه و واریانس

- برای ویژگی‌های دسته‌ای:

- تعداد مقادیر ویژگی‌های متمایز در خوشه (هم شمارش سطرها و هم به عنوان کسری از اندازه‌ی خوشه)
- درصد مقادیر تهی (یعنی فیلدهای خالی)
- درصد مقادیر متعلق به مد
- درصد مقادیر متعلق به دو مقدار ویژگی بالاتر از بقیه
- درصد مقادیری که یکتا هستند
- ویژگی‌های عددی در آرایه‌ی شمارش مقادیر (بالا را ببینید)
- آنتروپی که به صورت $\sum_i -p_i \, log(p_i)$ محاسبه می‌شود، در اینجا محدوده‌ی i بین مقادیر ویژگی و مقدار زیر است:

$$p_i = \frac{\text{تعداد نمونه های } i}{\text{تعداد مقادیر ویژگی متمایز}}$$

در ویژگی‌های عددی که دو مقدار می‌گیرند، می‌توانیم آن مقدار را به صورت $0/1$ کدگذاری کنیم و ویژگی‌های عددی توضیح داده شده در بالا را محاسبه کنیم؛ همچنین می‌توانیم فیلدهای متنی را به عنوان فیلد دسته‌ای در نظر بگیریم و ویژگی‌های توزیع مربوطه را محاسبه کنیم.

4.2 ویژگی‌های الگودار

اغلب دیده می‌شود وقتی یک موجودیت واحد – چه انسان چه ربات – خوشه‌ای از حساب‌های جعلی را ثبت می‌کند، متن‌های وارد شده توسط کاربر در یک یا چند ستون همیشه منطبق با یک الگوی مشخص است. برای مثال، آدرس‌های ایمیل در آن حساب می‌تواند به صورت زیر باشد (این یک نمونه‌ی مصنوعی است):

charlesgreen992@domain.com
josephbaker247@domain.com
thomasadams319@domain.com
chrisnelson211@domain.com
danielhill538@domain.com
paulwhite46@domain.com
markcampbell343@domain.com
donaldmitchell92@domain.com
georgeroberts964@domain.com
kennethcarter149@domain.com

تمامی این آدرس‌های ایمیل به وضوح عبارت منظم *[a-z]+[0-9]+@domain\.com* را نشان می‌دهند. می‌توان این عبارت منظم را بر روی آدرس‌های ایمیل اعمال کرد تا یک ویژگی دودویی به‌دست آید، و با آن می‌توان ویژگی‌های توزیع پایه را که در بالا توضیح داده شد به‌دست آورد. در این پژوهش روش پراسه و همکاران [28] به صورت نظری بر روی مجموعه‌ی آموزشی اعمال شد تا فهرستی از عبارات منظم که اسپم هستند تولید شود و از هر عبارت منظم به عنوان یک ویژگی دودویی استفاده شود. با این حال، این رویکرد بردارهای ویژگی بسیار پراکنده‌ای تولید می‌کند و نمی‌تواند به الگوهای شناسایی نشده تعمیم داده شود.

به‌جای تکیه بر عبارات منظم، دو «الگوریتم کدگذاری الگو» طراحی شد که متن‌های دلخواه را به فضای کوچک‌تری نگاشت می‌کند. اولین الگوریتم دسته‌های کاراکتر را قانون‌مند می‌سازد: کلیه‌ی کاراکترها به دسته‌های متنی مانند حروف بزرگ، حروف کوچک، رقم، علائم و غیره دسته‌بندی می‌شوند، و هر کاراکتر به یک کاراکتر نماینده‌ی آن کلاس نگاشت می‌شود، چنان‌که در الگوریتم 1 در زیر توضیح داده شده است.

الگوریتم 1: الگوریتم کدگذاری الگو (با حفظ طول)

نیازمندی: s.length > 0

```
Require: s.length > 0
procedure ENCODE(s)                ▷ 'abc12' → 'LLLDD'
    i ← 0
    t ← ''
    while i < s.length do
        if isUpperCase(s[i]) then
            t ← t + 'U'
        else if isLowerCase(s[i]) then
            t ← t + 'L'
        else if isDigit(s[i]) then
            t ← t + 'D'
        else
            t ← t + 'O'
        end if
    end while
    return t
end procedure
```

این الگوریتم که طول رشته را حفظ می‌کند، قادر خواهد بود آدرس‌های ایمیلی را تشخیص دهد که همه دارای 8 حرف به علاوه‌ی 3 رقم در همان دامنه هستند. با این حال، این الگوریتم فهرست آدرس‌های ایمیل بالا را شناسایی نخواهد کرد، چون اسامی و اعداد آن‌ها طول‌های مختلفی دارند. برای حل این مشکل از یک الگوریتم مستقل از متن استفاده می‌شود که نمونه‌های متوالی یک دسته را در یک نماینده‌ی واحد جمع می‌کند، چنان‌که در الگوریتم 2 در زیر توضیح داده شده است.

خروجی این الگوریتم در فهرست نام کاربری ایمیل (یعنی متن قبل از علامت @) در بالا در همه‌ی حالت‌ها به صورت LD خواهد بود. تجربه نشان می‌دهد که خیلی به ندرت مجموعه‌ای از کاربران قانونی همگی از چنین الگویی پیروی می‌کنند، پس این یک ویژگی خوب برای تمایز خوشه‌هایی است که توسط یک موجودیت واحد ایجاد شده‌اند.

به علاوه در استفاده از الگوریتمی که در بالا شرح داده شد، می‌توان دسته‌های کاراکتر جدیدی به این الگوریتم اضافه کرد، مثلاً علائم نگارشی یا فاصله، یا می‌توان دسته‌ها را در هم ادغام کرد (برای مثال حروف کوچک و حروف بزرگ). در این روش اندازه‌های ساده مانند طول متن و تعداد لغات نیز می‌تواند زیرمجموعه‌ی این چارچوب باشد. برخی از الگوهایی که در تجزیه تحلیل‌ها در نظر گرفته شدند به صورت زیر هستند:

- Encode() (الگوریتم 1)
- ShortEncode() (الگوریتم 2)
- Len(Encode()) که از یک دسته کاراکتر واحد استفاده می‌کند (مثلاً طول متن)
- Len(ShortEncode()) که از دو دسته کاراکتر استفاده می‌کند، با فاصله و بدون فاصله (یعنی تعداد لغات)
- ویژگی‌های دودویی که وجود هر دسته کاراکتر را در Encode() بررسی می‌کند.
- Encode() در اولین کاراکتر متن.

وقتی نگاشت به فضای دسته‌ای کوچکتری نگاشت شد، ویژگی‌های توزیع پایه که در بخش 4.1 توضیح داده شد، برای محاسبه‌ی ویژگی‌های عددی اعمال می‌شود.

```
Require: s.length > 0
 1: procedure ShortEncode(s)          ▷ 'abc12' → 'LD'
 2:     i ← 0
 3:     s ← Encode(s)
 4:     curr ← ''
 5:     t ← ''
 6:     while i < s.length do
 7:         if curr ≠ s[i] then
 8:             t ← t + s[i]
 9:             curr ← s[i]
10:         end if
11:         i ← i + 1
12:     end while
13:     return t
14: end procedure
```

4.3 ویژگی‌های تکراری

پس از بررسی دقیق خوشه‌های حساب جعلی، اغلب الگوهایی یافت می‌شوند که برای چشم آموزش‌دیده آشکار هستند ولی برای توصیف الگوریتمی دشوارند. برای مثال، دو مجموعه نام زیر را در نظر بگیرید (بازهم یک نمونه‌ی مصنوعی است):

خوشه‌ی 1	خوشه‌ی 2
Charles Green	*Shirely Lofgren*
Joseph Baker	*Tatiana Gehring*
Thomas Adams	*China Arzate*
Chris Nelson	*Marcelina Pettinato*
Daniel Hill	*Marilu Marusak*

Bonita Naef	*Paul White*
Etta Scearce	*Mark Campbell*
Paulita Kao	*Donald Mitchell*
Alaine Propp	*George Roberts*
Sellai Gauer	*Kenneth Carter*

کاملاً واضح است که این اسامی به صورت تصادفی از جامعه در سطح وسیع نمونه‌برداری نمی‌شوند. اسامی خوشه‌ی 1 همگی اسامی رایج مردانه هستند (در واقع، آن‌ها از پرتکرارترین نام‌ها و نام خانوادگی‌های اطلاعات سرشماری آمریکا گرفته شده‌اند) و اسامی خوشه‌ی 2 بسیار نادر هستند – ممکن است شخصی در دنیا به نام Bonita Naef وجود داشته باشد، ولی احتمال این‌که او با همان آدرس آی‌پی در یک شبکه‌ی اجتماعی ثبت نام کرده باشد که Alaine Propp و بقیه ثبت نام کرده‌اند، بسیار کم است.

این شواهد با استفاده از اطلاعات کل پایگاه داده‌ی شبکه‌های اجتماعی بیان می‌شوند. به طور خاص، برای یک ستون مشخص از متن (مانند اسم کوچک)، تعداد تکرار آن متن بین اعضای شبکه‌ی اجتماعی محاسبه می‌شود. این کار عددی بین صفر و 1 را می‌دهد که با آن می‌توان ویژگی‌های توزیع پایه را که در بخش 4.1 توضیح داده شد، محاسبه کرد. همین کار را برای لگاریتم تعداد تکرارها یا رتبه‌های ویژگی‌ها در فهرست مرتب‌شده‌ی تکرارها می‌توان انجام داد، این کار می‌تواند به تشخیص ورودی‌های بسیار نادر از ورودی‌های نادر کمک کند.

4. نتایج تجربی

5.1 کسب اطلاعات

مدل این پژوهش بر روی داده‌های برچسب‌گذاری شده‌ی لینکداین ارزیابی شد، این برچسب‌ها توسط تیم امنیتی و یا تیم اعتماد و ایمنی لینکداین ارائه شده بود. رویکرد این پژوهش ابتدا اقتضا می‌کند روشی برای خوشه‌بندی حساب‌ها انتخاب شود. در مطالعه‌ی انجام شده، خوشه‌هایی از حساب‌های لینکداین با گروه‌بندی بر اساس آدرس آی‌پی[6] ثبت نام و تاریخ ثبت نام (به وقت پاسیفیک) ایجاد شد. انتخاب این رویکرد عمدتاً به این دلیل بود که در این گروه به دست آوردن مقدار زیادی داده‌ی برچسب‌گذاری شده‌ی دستی امکان پذیر امکان پذیر می‌شود؛ در بخش 7 سایر رویکردهای خوشه‌بندی توضیح داده خواهد شد.

در مجموعه‌ی آموزشی این پژوهش حساب‌های برچسب‌گذاری شده از یک دوره‌ی 6 ماهه از 1 دسامبر 2013 تا 31 می 2014 جمع‌آوری شد. در طول این مدت حساب‌های تمامی خوشه‌ها (آی‌پی، تاریخ ثبت) که یک معیار درونی برای ثبت‌نام‌های مخرب بودند، به تیم اعتماد و ایمنی لینکداین ارسال شدند تا به طور دستی بررسی شده و در برابر آن‌ها اقدام شود. اطلاعات خام پروفایل‌ها برای همه‌ی حساب‌های این خوشه‌ها استخراج شد و اگر آن حساب محدود شده بود، آن را به عنوان حساب جعلی برچسب‌گذاری کردند یا اگر در مدت بازبینی شرایط خوبی

6. برای حساب‌هایی که از *IPv6* استفاده کرده بودند، با سابنت 56/ گروه‌بندی کردیم.

داشت، آن را به عنوان حساب واقعی برچسب‌گذاری کردند. جمع کل تعداد حساب‌های برچسب‌گذاری شده 260,644 بود، که 153,019 تا از این حساب‌ها جعلی و 107,625 حساب قانونی بودند.

به روشی مشابه اطلاعات را، از ژوئن 2014 ، برای استفاده به عنوان داده‌های آزمایشی «خارج از نمونه» به دست آوردند. این اطلاعات شامل 30.550 حساب بود که 15.078 حساب، جعلی و 15,472 حساب، قانونی بودند.

5.2 برچسب‌گذاری خوشه

حساب‌های برچسب‌گذاری شده در مجموعه داده‌ها در 20.559 خوشه (آی‌پی، زمان) قرار گرفتند. متوسط اندازه‌ی خوشه‌ها 9 تا بود؛ نموداری از اندازه‌ی خوشه‌ها در شکل 2 آمده است. در هر خوشه درصد حساب‌های برچسب‌گذاری شده به عنوان اسپم را محاسبه شد؛ نموداری از این اطلاعات در شکل 3 نشان داده شده است. نتیجه این شد که 89٪ از این خوشه‌ها یا هیچ حساب جعلی نداشتند یا همه‌ی حساب‌های آن جعلی بودند، و تنها 3/8 ٪ خوشه‌ها بین 20٪ تا 80٪ حساب جعلی داشتند.

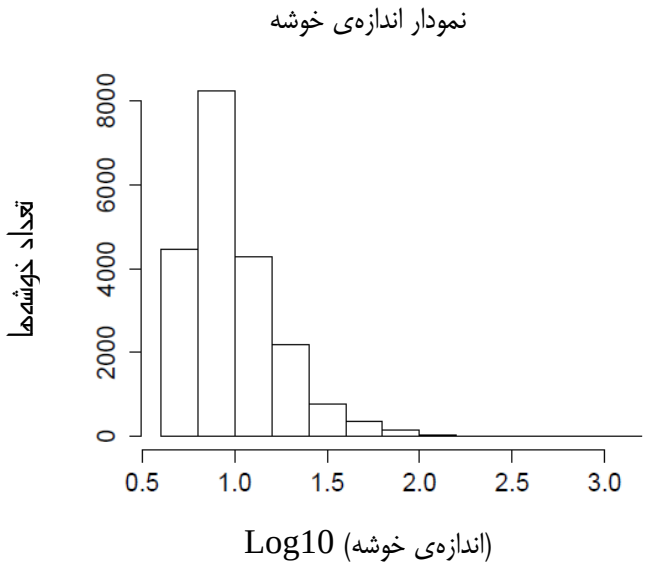
برای تعیین آستانه‌ی برچسب‌گذاری خوشه‌ها به عنوان حساب جعلی (بخش 3.1 را ببینید)، دسته‌بند جنگل تصادفی با استفاده از برچسب‌های خوشه‌ای اجرا شد که با تنظیم سه آستانه‌ی متفاوت تولید شده بود: 20٪ ، 50٪ و 80٪ . این اندازه‌های AUC حاصل در سطح حساب، به ترتیب 0/9765، 0/9777 و 0/9776 بودند. این نتیجه به دست آمد که ترتیب نسبی حساب‌های امتیازدهی شده به آستانه‌ی برچسب‌گذاری خوشه‌ها حساس نیست، و 50٪ را به عنوان آستانه‌ی آزمایش‌ها انتخاب شد. در این آستانه، 10.456 خوشه‌ی آموزشی به عنوان اسپم برچسب‌گذاری شده بود و 10.102 خوشه به عنوان خوشه‌ی قانونی برچسب خورده بود. مجموعه‌ی آزمایشی خارج از نمونه در 2.705 خوشه جای گرفت که 1.227 خوشه اسپم و 1.478 خوشه قانونی بودند.

5.3 تحلیل عملکرد

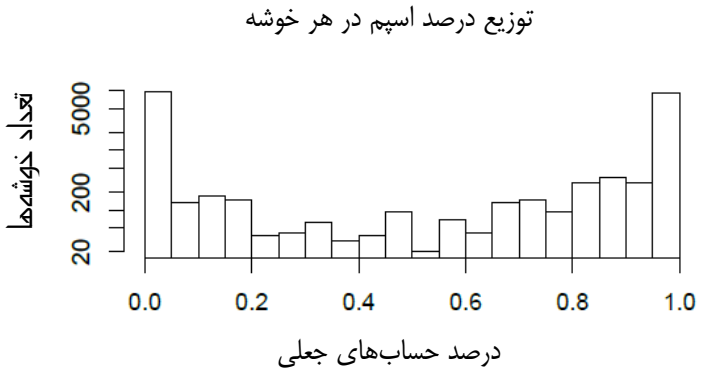
رویکرد معرفی‌شده با استفاده از الگوریتم‌های یادگیری ماشین که در بخش 2 شرح داده شد ارزیابی شد. رگرسیون منطقی، SVM و جنگل تصادفی. ما این سه الگوریتم را برای نمایش دادن رویکردهای ممکن انتخاب شدند؛ در اصل هر الگوریتم دسته‌بندی دودویی می‌تواند استفاده شود و بهترین الگوریتم ممکن است بر اساس منطقه‌ی دامین تغییر کند. برای انجام یک مقایسه و ارزیابی منصفانه، پارامترهای همه‌ی الگوریتم‌های یادگیری نظارت شده از طریق یک فرآیند جداسازی ارزیابی متقابل 80-20 تعریف شده بودند. به طور مشخص، 80٪ از داده‌های آموزشی برای ساخت دسته‌بند استفاده شده بودند و 20٪ باقیمانده‌ی داده‌ها برای آزمون‌های عملکرد «درون نمونه‌ای» استفاده شدند. پارامتر بهینه تنظیم‌شده، تنظیماتی است که AUC آزمایشی درون نمونه‌ای را حداکثر سازد.

سه الگوریتم فوق‌الذکر با استفاده از بسته‌های "glmnet" [14]، "e1071" [25] و «جنگل تصادفی» [23] به ترتیب بر روی داده‌های آموزشی اجرا شد. جدول 1 عملکرد پیش‌بینی درون‌نمونه‌ای را نشان می‌دهد چنان‌که با AUC اندازه‌گیری شده و با دقت 95٪ فراخوانی شده بود. این داده‌ها نشان می‌دهد جنگل تصادفی بر اساس هر دو اندازه بهترین عملکرد را داشته است. دسته‌بند غیرخطی دیگر، SVM با هسته‌ی

RBF نیز عملکرد خوبی از نظر مقدار AUC خود داشت. با این حال، فراخوانی آن با دقت 95٪ به خوبی جنگل تصادفی نیست که نشان می‌دهد در SVM، اگرچه اعتماد زیادی به حساب‌های جعلی گرفته شده نیست، اما هنوز حساب‌های جعلی زیادی وجود دارند که توسط این مدل گرفته نشده‌اند. در میان همه‌ی این مدل‌ها، رگرسیون منطقی بدترین عملکرد را داشت، چون غیرخطی بودن آن در الگوهای واقعی نمی‌تواند به خوبی با دسته‌بندهای غیرخطی مدل‌سازی شود.



شکل 2: توزیع اندازه‌ی خوشه‌ها در داده‌های آموزشی.



شکل 3: توزیع درصد اسپم در هر خوشه برای داده‌های آموزشی

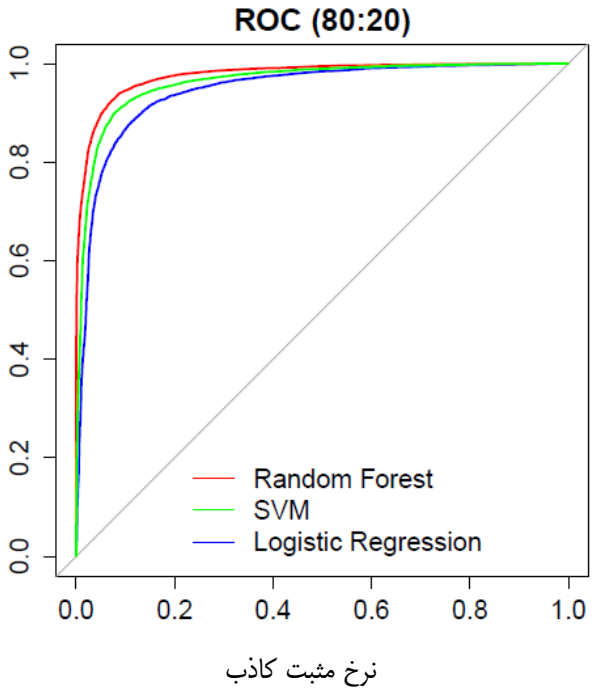
جدول 1: جداسازی 80–20 عملکرد آزمایش (سطح خوشه)

Algorithm	AUC	Recall@p95
Random forest	0.978	0.900
Logistic regression	0.936	0.657
SVM	0.963	0.837

جدول 2 AUC آزمایش شده و فراخوانی با دقت 95٪ را برای تمامی الگوریتم‌ها در سطح حساب نشان می‌دهد؛ این زمانی است که به هر حساب امتیاز محاسبه شده برای خوشه‌اش اختصاص داده می‌شود. داده‌ها نشان می‌دهند که پیش‌بینی ما برای هر الگوریتم برای هر حساب حتی دقیق‌تر هم بوده است، و آزمایش داده‌ها نشان می‌دهد که این ناشی از طبقه‌بندی است که در خوشه‌های بزرگ‌تر دقیق‌تر است (جدول 5 را برای شواهد بیشتر ببینید).

جدول 2: جداسازی 80–20 عملکرد آزمایش (سطح حساب)

Algorithm	AUC	Recall@p95
Random forest	0.978	0.935
Logistic regression	0.951	0.821
SVM	0.961	0.889



شکل 4: مقایسه‌ی منحنی‌های ROC در مدل‌های متفاوت داده‌های درون‌نمونه‌ای.

مدل پژوهش بر روی داده‌های خارج از نمونه از ژوئن 2014 نیز آزموده شد. انگیزه‌ی انجام آزمون خارج از نمونه این بود که الگوهای اسپمرها در داده‌های حساب‌های جعلی آن‌ها در طول زمان تغییر خواهد کرد چرا که آن‌ها از شکست تجربه گرفته و یاد می‌گیرند. انجام آزمایش خارج از نمونه این سناریو را در محصول شبیه‌سازی می‌کند، و ارزیابی عملی و مفیدی از عملکرد واقعی مدل ارائه می‌دهد.

جدول 3 و 4 مقایسه‌ی عملکرد خارج از نمونه را برای سه مدل آموزشی در داده‌های آموزشی به ترتیب در سطح خوشه و سطح حساب نشان می‌دهد. این داده‌ها نشان می‌دهند جنگل تصادفی همچنان بهترین عملکرد را بر اساس تمامی معیارها دارد. فراخوانی با دقت 95٪ در هر سه الگوریتم چنان‌که با نتایج ارزیابی متقابل مقایسه شد، کاهش می‌یابد، که فرضیه‌ی پژوهش را در مورد سطح مشخص دقت تأیید می‌کند (یعنی کسری از حساب‌هایی که پیش‌بینی شده جعلی باشند و واقعاً جعلی هستند)، با این حال حساب‌های جعلی بیشتری وجود دارند که در مجموعه‌ی داده‌ی جدیدتر گرفته نشده‌اند. همچنین نتایج نشان می‌دهد باید مدل پژوهش را به طور منظم دوباره آموزش داد تا بتوان الگوهای جدیدتر را گرفت و نسبت حساب‌های جعلی گرفته شده را بزرگ‌تر کرد.

جدول 3: عملکرد آزمون خارج از نمونه (سطح خوشه)

Algorithm	AUC	Recall@p95
Random forest	0.949	0.720
Logistic regression	0.906	0.127
SVM	0.928	0.522

جدول 4: عملکرد آزمون خارج از نمونه (سطح حساب)

Algorithm	AUC	Recall@p95
Random forest	0.954	0.713
Logistic regression	0.917	0.456
SVM	0.922	0.311

یک یافته‌ی جالب در اینجا این است که وقتی از سطح خوشه به سطح حساب می‌رویم، عملکرد دسته‌بند SVM *کاهش می‌یابد*. این نتایج نشان می‌دهد که برخلاف رگرسیون منطقی و جنگل‌های تصادفی، در دسته‌بندی خوشه‌های کوچک‌تر بهتر عمل می‌کند؛ همچنین یک رویکرد کلی ترکیب شده از همه‌ی دسته‌بندها را پیشنهاد می‌دهد که می‌تواند به عملکرد بهتر منجر شود.

تحلیل توسط اندازه‌ی خوشه. جدول 5 نشان می‌دهد که نتایج جنگل تصادفی توسط اندازه‌ی خوشه حذف شده است. می‌بینیم که هرچه خوشه بزرگ‌تر می‌شود، عملکرد مدل بهتر می‌شود. اگر خوشه‌ای بیش از 30 حساب داشته باشد، که به این معناست در یک روز از یک آدرس آی‌پی بیش از 30 حساب کاربری ایجاد شده است، تقریباً با اعتماد کامل می‌توان این خوشه و تمامی حساب‌های این خوشه را برچسب‌گذاری کرد. اگر خوشه‌ای بیش از 100 حساب داشته باشد، تقریباً به سطح دقت 100 درصد در تمامی معیارهای مجموعه‌ی ارزیابی متقابل می‌رسیم.

جدول 5: عملکرد جنگل تصادفی با اندازه‌ی خوشه

Cluster Size	AUC	Recall@p95
1 تا 10	0.967	0.817
11 تا 30	0.988	0.965
31 تا 100	0.988	0.989
بیش از 100	1.000	1.000

تجزیه و تحلیل ویژگی‌های برتر: برای داشتن دید بهتر به این ویژگی‌ها در این مطالعه انجام شده، آن‌ها را با استفاده از شاخص اهمیت جینی (Gini) رتبه‌بندی کردند که بر اساس مرجع جینی محاسبه شده است [4]. در مدل مورد مطالعه، بالاترین ویژگی‌ها شامل تعداد تکرار متوسط دو تا از کم‌رواج‌ترین نام‌ها یا نام‌های‌خانوادگی است، و همچنین نسبت الگوهای پرکاربردتر تولیدشده از الگوریتم کدگذاری الگوی استفاده شده در نام و آدرس ایمیل است.

5.4 تحلیل مثبت کاذب و منفی کاذب

تمامی حساب‌های مجموعه‌ی ارزیابی انجام شده و مجموعه‌ی آزمایشی خارج از نمونه به صورت دستی بررسی شد که پیش‌بینی شده بود حساب جعلی باشند ولی در واقع به عنوان حساب قانونی برچسب‌گذاری شده بودند. دیده شد که اکثریت این حساب‌ها از یک سازمان ثبت‌نام کرده بودند. تعدادی از اعضا از طریق سازمانی ثبت نام کرده بودند که احتمالاً با یک آدرس آی‌پی ثبت‌نام شده بودند و برخی از قسمت‌های پروفایل‌های آن‌ها مشابه بود. برای مثال، آدرس ایمیل آن‌ها ممکن است از یک الگوی معیار (مانند org.<نام سازمان>@<حروف اول نام ><نام خانوادگی>) پیروی کند. برای حل این مثبت‌های کاذب، یک مدل تشخیص حساب‌های سازمانی ایجاد شد و دسته‌بند چنان پیکربندی شد که حساب‌های سازمانی که آن‌ها را مدل به عنوان حساب جعلی شناسایی کرد، به‌جای این‌که به‌طور خودکار محدود شوند، برای بررسی دستی فرستاده شوند. این رویکرد کمک بزرگی به حل مشکل مثبت کاذب کرد.

همچنین تمامی حساب‌ها، در مجموعه داده‌ای که قانونی تشخیص داده شده بود ولی در واقع توسط افراد سازمان به عنوان حساب جعلی شناسایی شده بود، به طور دستی بررسی شدند. معمولاً اگر حجم زیادی ثبت نام قانونی وجود داشته باشد (برای مثال در طول رویداد بازاریابی لینکداین)، تعداد زیادی از ثبت‌نام‌ها در یک خوشه‌ی واحد قرار می‌گیرند. این اتفاق ممکن است برخی مدل‌های مبتنی بر قانون را فراخوانی کند که آن حساب‌ها را به صورت جعلی برچسب‌گذاری کند و برچسب‌گذار انسانی نیز ممکن است آن‌ها را به همان دلیل به عنوان جعلی برچسب بگذارد. با این حال، هرچه اندازه‌ی خوشه بزرگ‌تر شود، الگوهای پروفایل حساب کاربری در آن خوشه متفاوت‌تر خواهد شد، که این از نظر مدل عادی‌تر به نظر می‌رسد، بنابراین مدل می‌تواند آن خوشه را به درستی به عنوان خوشه‌ی خوب برچسب‌گذاری کند. این اتفاق همچنین توضیح می‌دهد که چرا وقتی خوشه‌ها بزرگ‌تر می‌شوند، چنان‌که در جدول 5 نشان داده شده است، مدل دقیق و دقیق‌تر می‌شود. وقتی مدل خطاهایی در برچسب انسانی قبلی می‌یابد (که با بررسی‌های دستی تأیید شد)، تصمیم قبلی تغییر داده شد و آن حساب‌ها دوباره به عنوان حساب قانونی برچسب‌گذاری شدند.

5.5 اجرا بر روی داده‌های حاضر

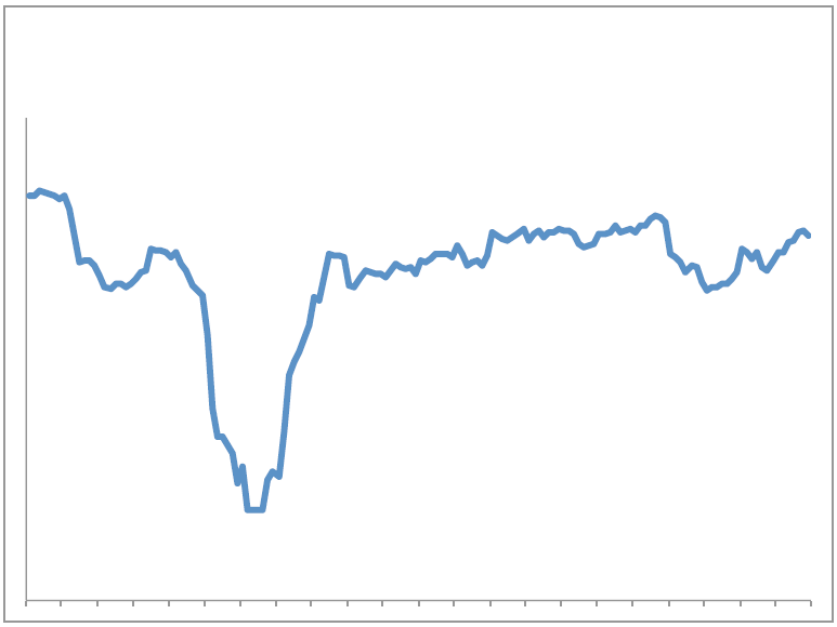
این سیستم با استفاده از جاوا، هایو و R پیاده‌سازی شد، و با مجموعه داده‌ی مورد بحث در بخش 5.1 آموزش داده شد. با استفاده از جریان هدوپ، روزانه بر روی ثبت‌نام‌های جدید لینکداین اجرا شد. حساب‌هایی که بیشترین امتیاز منفی را داشتند به طور خودکار محدود شدند.

امتیازات در «محدوده‌ی خاکستری» برای بررسی و اقدام دستی، به تیم اعتماد و ایمنی لینکداین ارسال شدند. این فرآیند اجازه می‌دهد داده‌های برچسب‌گذاری شده در موارد مرزی برای مدل‌های آموزشی آینده جمع‌آوری شوند.

از زمان انتشار این مدل، بیش از 15,000 خوشه توسط مدل گرفته شد که شامل بیش از 250,000 حساب جعلی لینکداین بود. روند دقت مدل را می‌توان در شکل 6 دید که یک میانگین متحرک دقت 14 روزه را در خود جای می‌دهد. کاهش دقت در یک نقطه به دلیل تعداد زیاد ثبت‌نام‌های سازمانی است که مدل آن‌ها را اشتباهاً گرفته است، پس از اضافه کردن «آشکارساز سازمان» دقت به مقدار قبلی خود بازگشته است.

دقت مدل در داده‌های حاضر در لینکداین

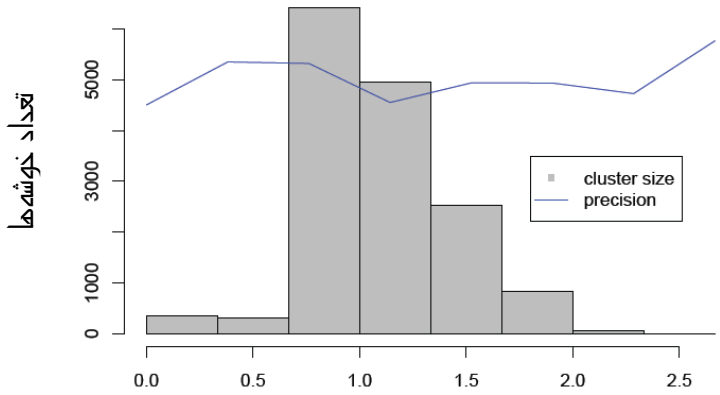
(میانگین متحرک 14 روزه)



شکل 6: میانگین متحرک 14 روزه‌ی دقت مدل (در سطح حساب) از زمان استقرار

شکل 7 نموداری از اندازه‌ی خوشه‌ها را در داده‌های حاضر و دقت (در سطح خوشه‌ها) را در هر مستطیل نشان می‌دهد. بیشتر خوشه‌ها نسبتاً کوچک تشخیص داده شدند؛ میانگین اندازه‌ی خوشه‌ها 11 حساب بود. در آزمایش‌های انجام شده برخلاف داده‌های آموزشی، دیده شد که عموماً دقت با اندازه‌ی خوشه افزایش نخواهد یافت، بجز در خوشه‌های بسیار بزرگ (با اندازه‌ی بیشتر از 100 حساب).

توزیع اندازه‌ی خوشه و دقت در داده‌های حاضر



شکل 7: توزیع اندازه‌ی خوشه و دقت در داده‌های حاضر

5. تحقیقات مرتبط

مشکل شناسایی حساب‌های جعلی در شبکه‌های اجتماعی اینترنتی از چندین دیدگاه متفاوت از جمله تحلیل رفتار، نظریه‌ی گراف، یادگیری ماشین و طراحی سیستم مطرح شده است.

با استفاده از دیدگاه رفتاری، مالهوترا و همکاران [24] ویژگی‌های را ایجاد کردند که کاربران مخربی را شناسایی کند که حساب‌های جعلی در شبکه‌های اجتماعی مختلف ایجاد می‌کنند. با این حال، ویژگی‌هایی که آن‌ها معرفی کردند همگی ویژگی‌های پایه‌ی پروفایل در سطح حساب بود. اگر همان اسپمر از پلت‌فرم‌های مختلف با همان اطلاعات پروفایل یکسان سوء استفاده نکند، تأثیر چنین ویژگی‌هایی کاهش خواهد یافت. تحقیقات زیادی برای تحلیل حساب‌های جعلی در شبکه‌های اجتماعی اینترنتی از دیدگاه نظریه‌ی گراف انجام شده است. دو تحقیق مرتبط تحقیقات یو (Yu) و همکاران [38] است که تعدادی از سازوکارهای امنیتی خاص سایبل را توصیف کرده است، و دیگری تحقیقات ویسوانات (Viswanath) و همکاران است که به بیشترین طرح‌های دفاع سایبل موجود اشاره می‌کند که با شناسایی اجتماعات محلی (یعنی خوشه‌هایی از گره‌ها که محکم‌تر از بقیه‌ی گراف به هم بسته شده‌اند) اطراف یک گره‌ی مورد اعتماد کار می‌کند.

در تحقیقات نظریه‌ی گراف اخیر، جیانگ (Jiang) و همکاران [17] پیشنهاد می‌کنند که حساب‌های جعلی را با ساخت گراف‌های تعامل نهفته به عنوان مدلی از رفتار مرور کاربر شناسایی کنیم. آن‌ها سپس ویژگی‌های ساختاری این گراف‌ها، تکامل، ساختار جامعه و دفعات ترکیب را با همین ویژگی‌ها از گراف‌های تعامل فعال و گراف‌های اجتماعی مقایسه می‌کنند. مهیسن (Mohaisen) و همکاران [27] گره‌های سایبل را شناسایی می‌کنند که ویژگی ترکیب سریع شبکه‌های اجتماعی را مختل می‌کند و به همین دلیل چندین اقدام اکتشافی را پیشنهاد می‌دهد تا ترکیب گراف‌هایی را که به کندی ترکیب می‌شوند با استفاده از ساختار توپولوژیکی آن‌ها بهبود دهد. کانتی (Conti) و همکاران [8] گراف‌های شبکه‌های اجتماعی را از یک دیدگاه پویا بررسی می‌کنند تا تبلیغ‌کنندگانی را شناسایی کنند که پروفایل‌های جعلی می‌سازند تا خود را به جای افراد واقعی جا بزنند و سپس با سایر افراد ارتباط برقرار کنند.

در حالی‌که روش‌های نظریه گراف بالا قابل اعمال بر روی خوشه‌های حساب‌هایی که در این مقاله مطالعه شد، هدف شناسایی خوشه‌ها *قبل از* آن است که بتوانند ارتباط برقرار کنند یا در رفتاری مشارکت کنند که ساختارهای گراف مرتبط را تولید می‌کند. بنابراین رویکرد اتخاذ شده بر

روی علائمی تمرکز می‌کند که در لحظه‌ی ثبت نام یا مدت بسیار کوتاهی پس از آن در دسترس است، و تنها شامل مقدار کمی از داده‌های فعالیت و اطلاعات ارتباطی است و یا شامل هیچ اطلاعات ارتباطی نیست.

بسیاری از محققان الگوریتم‌های یادگیری ماشین را در مسئله‌ی شناسایی اسپم در شبکه‌های اجتماعی اینترنتی اعمال کرده‌اند. فایر (Fir) و همکاران [12] از ناهنجاری‌های توپولوژی، درخت تصمیم‌گیری و دسته‌بندهای ساده‌ی بایز برای تشخیص اسپمر و پروفایل‌های جعلی در شبکه‌های اجتماعی مختلف استفاده می‌کنند. جین (Jin) و همکاران [18] رفتار را تحلیل می‌کنند تا حملات شبیه‌سازی شده را شناسایی کنند و یک چارچوب تشخیصی را پیشنهاد می‌دهند. کائو (Cao) و همکاران [5] یک الگوریتم رتبه‌بندی را برای رتبه دادن به کاربران در سرویس‌های اینترنتی و شناسایی حساب‌های جعلی معرفی کردند؛ این رتبه بر اساس احتمال درجهٔ قانون‌مند شده از یک گشت کوتاه تصادفی در منطقه‌ی غیر سایبل محاسبه می‌شود. تان (Tan) و همکاران [32] مسئله‌ی شناسایی اسپم در شبکه را در یک چارچوب یادگیری بدون نظارت قرار می‌دهند که افراد غیر اسپمر را عمداً از شبکه حذف کند و هم از گراف اجتماعی و هم گراف ارتباطات کاربر استفاده کند.

برخی از محققین به‌جای تمرکز بر شناسایی حساب‌های جعلی پس از نفوذ آن‌ها به شبکه، بر روی طراحی سیستمی تمرکز می‌کنند که خودش بتواند در مرحله‌ی اول از حملات جلوگیری کند. لِسنیوسکی-لاس (Lesniewski-Laas) و کاشوک (Kaashoek) [22] یک پروتکل مسیریابی جدید برای جداول هش(Hash) توزیع‌شده ارائه می‌دهند که کارآمد است و به شدت در برابر حملات سایبل مقاوم است. چیلوکا (Chiluka) و همکاران [6] یک دیدگاه طراحی جدید را در تبادلات بین اتصال شبکه و انعطاف‌پذیری حمله در طرح دفاع سایبل مبتنی بر شبکه معرفی می‌کنند که در آن هر گره تنها با چند همسایه‌ی انتخابی خود با فاصله‌ی 2 گام بر مبنای حداقل گسترش (MinEC) ارتباط اضافه می‌کند. ویسوانات و همکاران [37] سیستمی را ارائه می‌دهند که از روش‌های مبتنی بر مسیریابی برای تخمین کارآمد پرداخت‌های اعتباری در شبکه‌های بزرگ استفاده می‌کند.

در حالی که روش‌های طراحی شده‌ی سیستمی برای اجتناب از سوء استفاده می‌تواند مؤثر باشد، در شبکه‌ی بزرگی که در اصل برای بهینه‌سازی رشد طراحی شده است، عملاً چندان کارآمد نیستند و درگیر شدن زمان پیش از سوء استفاده، مسئله ای مهم است.

6. نتیجه‌گیری و تحقیقات آینده

در این مقاله یک خط لوله‌ی یادگیری ماشین برای شناسایی حساب‌های جعلی در شبکه‌های اجتماعی اینترنتی معرفی شد. به جای پیش‌بینی کردن برای هر حساب انفرادی این، سیستم خوشه‌های حساب‌های جعلی را دسته‌بندی می‌کند تا تشخیص دهد آیا توسط یک عامل واحد ایجاد شده‌اند. ارزیابی انجام شده در داده‌های درون‌نمونه‌ای و خارج از نمونه عملکردی قوی را نشان داد و از این سیستم در یک محصول برای یافتن و محدود کردن بیش از 250,000 حساب استفاده شد.

در این پژوهش چارچوب مورد نظر بر روی خوشه‌های تولید شده توسط دسته‌بندی ساده بر حسب تاریخ ثبت نام و آدرس آی‌پی ثبت نام ارزیابی کردیم. در تحقیقات آینده انتظار می‌رود مدل موردنظر بر روی خوشه‌هایی که با دسته‌بندی بر حسب ویژگی‌های دیگری ایجاد شده‌اند اجرا شود، مانند ISP یا شرکت، و دوره‌های زمانی دیگر مانند هفته یا ماه. یکی دیگر از خطوط امیدوارکننده در تحقیقات استفاده از الگوریتم‌های

خوشه‌بندی پیچیده‌تر مانند k می‌نیمم یا خوشه‌بندی سلسله مراتبی است. در عین حال که این رویکردها ممکن است پربار باشد، موانعی در راه اجرا در مقیاس بزرگ را نشان می‌دهد: k می‌نیمم ممکن است خوشه‌های زیادی لازم داشته باشد (یعنی مقادیر بزرگ k) تا نتایج مفیدی تولید کند، و خوشه‌بندی سلسله مراتبی ممکن است از لحاظ محاسباتی برای دسته‌بندی میلیون‌ها حساب دشوار باشد.

در دیدگاه مدل‌سازی، یک مسیر مهم برای تحقیقات آینده به کار بردن مجموعه ویژگی‌های استفاده شده در سایر مدل‌های شناسایی اسپم و در نتیجه درک پیش‌گویی با چند مدل تجمیع شده است. یک مسیر دیگر مستحکم کردن سیستم در برابر حملات مختلف است، مانند ربات‌های اینترنتی که همه‌ی ویژگی‌های آن‌ها با هم فرق می‌کند یا مهاجمی که از شکست‌هایش درس می‌گیرد. آخرین مسیر ساخت ویژگی‌های تطبیقی الگوی حساس به زبان است؛ ویژگی‌های ما فرض می‌کردند که متن با الفبای انگلیسی نوشته شده است که بتواند به تعداد دسته‌های کاراکتر کمتری نگاشت شود (برای مثال حروف بزرگ یا حروف کوچک) و این به راحتی با زبان‌های تصویری مانند چینی سازگار نیست.

قدردانی

شک ندارم که بهشت در دستان تو هم جاریست ای پـــــــدر

از پدر عزیزم نهایت تشکر و قدردانی را دارم و به روزی امید دارم که بتوانم همچون او برای فرزندانم باشم.

مراجع

[1] Cao Xiao and David Mandell Freeman and Theodore Hwa. Detecting Clusters of Fake Accounts in Online Social Networks. 2014, 2015.

[2] J. Beall. Publisher uses fake LinkedIn identities to attract submissions. http://scholarlyoa.com/2015/02/10/ publisher-uses-fake-linkedin-identities-to-attract-submissions.

[3] B. E. Boser, I. M. Guyon, and V. N. Vapnik. A training algorithm for optimal margin classi_ers. In Proceedings of the 5th Annual ACM Workshop on Computational Learning Theory, pages 144{152. ACM Press, 1992.

[4] L. Breiman. Random forests. Mach. Learn., 45(1):5{32, Oct. 2001.

[5] Q. Cao, M. Sirivianos, X. Yang, and T. Pregueiro. Aiding the detection of fake accounts in large scale social online services. In Proceedings of the 9th USENIX Conference on Networked Systems Design and Implementation, NSDI'12, pages 15{15, Berkeley, CA, USA, 2012. USENIX Association.

[6] N. Chiluka, N. Andrade, J. Pouwelse, and H. Sips. Social networks meet distributed systems: Towards a robust sybil defense under churn. In Proceedings of the 10th ACM Symposium on Information, Computer and Communications Security, ASIA CCS '15, pages 507{518, New York, NY, USA, 2015. ACM.

[7] D. B. Clark. The bot bubble: How click farms have inated social media currency. The New Republic, April 20 2015. Available at http://www.newrepublic.com/article/121551/bot-bubble-click-farms-have-inated-social-media-currency.
[8] M. Conti, R. Poovendran, and M. Secchiero. Fakebook: Detecting fake pro_les in on-line social networks. In Proceedings of the 2012 International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2012), ASONAM '12, pages 1071{1078, Washington, DC, USA, 2012. IEEE Computer Society.
[9] N. Cristianini and J. Shawe-Taylor. An Introduction to Support Vector Machines and Other Kernel-based Learning Methods. Cambridge University Press, New York, NY, USA, 2000.
[10] G. Danezis and P. Mittal. Sybilinfer: Detecting sybil nodes using social networks. Technical Report MSR-TR-2009-6, Microsoft, January 2009.
[11] Digital Trends Sta_. 40 pct. fake pro_les on Facebook? http://www.digitaltrends.com/computing/fake-pro_les-facebook/.
[12] M. Fire, G. Katz, and Y. Elovici. Strangers intrusion detection - detecting spammers and fake pro_les in social networks based on topology anomalies. ASE Human Journal, 1(1):26{39, Jan. 2012.
[13] D. M. Freeman. Using Naive Bayes to detect spammy names in social networks. In A. Sadeghi, B. Nelson, C. Dimitrakakis, and E. Shi, editors, AISec'13, Proceedings of the 2013 ACM Workshop on Arti_cial Intelligence and Security, Co-located with CCS 2013, Berlin, Germany, November 4, 2013, pages 3{12. ACM, 2013.
[14] J. Friedman, T. Hastie, and R. Tibshirani. Regularization paths for generalized linear models via coordinate descent. Journal of Statistical Software, 33(1):1{22, 2010.
[15] T. Hastie, R. Tibshirani, and J. Friedman. The Elements of Statistical Learning. Springer Series in Statistics. Springer New York Inc., New York, NY, USA, 2001.
[16] L. Huang, A. D. Joseph, B. Nelson, B. I. P. Rubinstein, and J. D. Tygar. Adversarial machine learning. In Proceedings of the 4th ACM Workshop on Security and Arti_cial Intelligence, AISec 2011, Chicago, IL, USA, October 21, 2011, pages 43{58, 2011.
[17] J. Jiang, C. Wilson, X. Wang, W. Sha, P. Huang, Y. Dai, and B. Y. Zhao. Understanding latent interactions in online social networks. ACM Trans. Web, 7(4):18:1{18:39, Nov. 2013.
[18] L. Jin, H. Takabi, and J. B. Joshi. Towards active detection of identity clone attacks on online social networks. In Proceedings of the First ACM Conference on Data and Application Security and Privacy, CODASPY '11, pages 27{38, New York, NY, USA, 2011. ACM.
[19] P. Judge. Social klepto: Corporate espionage with fake social network accounts. https://www.rsaconference.com/writable/presentations/_le upload/br-r32.pdf.
[20] K. Lee. Fake pro_les are killing LinkedIn's value. http://www.clickz.com/clickz/column/2379996/fake-pro_les-are-killing-linkedin-s-value.

[21] K. Lee, B. D. Eo_, and J. Caverlee. Seven months with the devils: a long-term study of content polluters on Twitter. In AAAI International Conference on Weblogs and Social Media (ICWSM), 2011.
[22] C. Lesniewski-Laas and M. F. Kaashoek. Whanau: A sybil-proof distributed hash table. In Proceedings of the 7th USENIX Conference on Networked Systems Design and Implementation, NSDI'10, pages 8{8, Berkeley, CA, USA, 2010. USENIX Association.
[23] A. Liaw and M. Wiener. Classi_cation and regression by randomforest. R News, 2(3):18{22, 2002.
[24] A. Malhotra, L. Totti, W. Meira Jr., P. Kumaraguru, and V. Almeida. Studying user footprints in di_erent online social networks. In Proceedings of the 2012 International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2012), ASONAM '12, pages 1065{1070, Washington, DC, USA, 2012. IEEE Computer Society.
[25] D. Meyer, E. Dimitriadou, K. Hornik, A. Weingessel, F. Leisch, and C. Chang. R package \e1071". 2014.
[26] A. Mislove, B. Viswanath, K. P. Gummadi, and P. Druschel. You are who you know: Inferring user pro_les in online social networks. In Proceedings of the Third ACM International Conference on Web Search and Data Mining, WSDM '10, pages 251{260, New York, NY, USA, 2010. ACM.
[27] A. Mohaisen and S. Hollenbeck. Improving social network-based sybil defenses by rewiring and augmenting social graphs. In Revised Selected Papers of the 14th International Workshop on Information Security Applications - Volume 8267, WISA 2013, pages 65{80, New York, NY, USA, 2014. Springer-Verlag New York, Inc.
[28] P. Prasse, C. Sawade, N. Landwehr, and T. Sche_er. Learning to identify regular expressions that describe email campaigns. In Proceedings of the 29th International Conference on Machine Learning, ICML 2012, Edinburgh, Scotland, UK, June 26 - July 1, 2012, 2012.
[29] A. Rakotomamonjy. Variable selection using svm based criteria. J. Mach. Learn. Res., 3:1357{1370, Mar. 2003.
[30] L. Ru_. Why do people create fake LinkedIn pro_les? http://integratedalliances.com/blog/ why-do-people-create-fake-linkedin-pro_les.
[31] M. Singh, D. Bansal, and S. Sofat. Detecting malicious users in Twitter using classi_ers. In Proceedings of the 7th International Conference on Security of Information and Networks, SIN '14, pages 247:247{247:253, New York, NY, USA, 2014. ACM.
[32] E. Tan, L. Guo, S. Chen, X. Zhang, and Y. Zhao. Unik: Unsupervised social network spam detection. In Proceedings of the 22nd ACM International Conference on Conference on Information & Knowledge Management, CIKM '13, pages 479{488, New York, NY, USA, 2013. ACM.
[33] K. Thomas, D. McCoy, C. Grier, A. Kolcz, and V. Paxson. Tra_cking fraudulent accounts: The role of the underground market in Twitter spam and abuse. In Proceedings of the 22nd USENIX Conference on Security, SEC'13, pages 195{210, Berkeley, CA, USA, 2013. USENIX Association.
[34] R. Tibshirani. Regression shrinkage and selection via

the Lasso. Journal of the Royal Statistical Society, Series B, 58:267{288, 1994.
[35] V. N. Vapnik. The Nature of Statistical Learning Theory. Springer-Verlag New York, Inc., New York, NY, USA, 1995.
[36] B. Viswanath, M. A. Bashir, M. Crovella, S. Guha, K. P. Gummadi, B. Krishnamurthy, and A. Mislove. Towards detecting anomalous user behavior in online social networks. In Proceedings of the 23rd USENIX Conference on Security Symposium, SEC'14, pages 223{238, Berkeley, CA, USA, 2014. USENIX Association.
[37] B. Viswanath, M. Mondal, K. P. Gummadi, A. Mislove, and A. Post. Canal: Scaling social network-based sybil tolerance schemes. In Proceedings of the 7th ACM European Conference on Computer Systems, EuroSys '12, pages 309{322, New York, NY, USA, 2012. ACM.
[38] H. Yu. Sybil defenses via social networks: A tutorial and survey. SIGACT News, 42(3):80{101, Oct. 2011.
[39] H. Yu, P. B. Gibbons, M. Kaminsky, and F. Xiao. Sybillimit: A near-optimal social network defense against sybil attacks. IEEE/ACM Trans. Netw., 18(3):885{898, June 2010.

[40] S. Adikari and K. Dutta. Identifying fake pro_les in LinkedIn. Paci_c Asia Conference on Information Systems Proceedings 2014, 2014.