شناسایی رانش مفهومی در نگاره‌های فرایند کسب‌وکار با استفاده از یادگیری عمیق

فاطمه خجسته*، محسن کاهانی و بهشید بهکمال

[۱و۲و۳] گروه رایانه دانشگاه فردوسی، مشهد، ایران

چکیده

فرایندهای کسب‌وکار در دنیای واقعی بسیار پیچیده هستند و متناسب با تحولات محیطی دچار تغییر می‌شوند. این در حالی است که روش‌های کشف فرایند پایه، قادر به شناسایی این تغییرات نیستند و تنها فرایندهای ثابت را تحلیل می‌کنند؛ ازاین‌رو، روش‌هایی به‌منظور شناسایی رانش مفهومی در فرایندهای کسب‌وکار مطرح شدند. همه روش‌های موجود در این حوزه، با انتخاب ویژگی‌ها و مقایسه آنها با استفاده از پنجره سعی در شناسایی این تغییرات دارد. انتخاب ویژگی مناسب و همچنین اندازه مناسب پنجره چالش‌های اصلی این روش‌ها به‌شمار می‌آیند. در این پژوهش، با بیان مفهوم تعبیه دنباله که برگرفته از تعبیه واژه در دنیای پردازش زبان طبیعی است، روشی خودکار و مستقل از پنجره به‌منظور شناسایی رانش ناگهانی در نگاره‌های کسب‌وکار ارائه کرده‌ایم. استفاده از روش تعبیه دنباله، این امکان را فراهم می‌کند که انواع روابط میان دنباله‌ها و رویدادها را استخراج و رانش‌های موجود در فرایندها را شناسایی کنیم. ارزیابی‌ها نشان می‌دهد که روش پیشنهادی نسبت به روش‌های موجود دقت بالاتر و تأخیر شناسایی رانش کمتری دارد.

واژگان کلیدی: فرایندکاوی، رانش مفهومی، تغییرات فرایند، تعبیه واژه

Concept drift detection in business process logs using deep learning

Fatemeh Khojasteh*, Mohsen Kahani & Behshis Behkamal

Departeman Computer Engineering, Ferdowsi University of Mashhad, Mashhad, Iran

Abstract

Process mining provides a bridge between process modeling and analysis on the one hand and data mining on the other hand. Process mining aims at discovering, monitoring, and improving real processes by extracting knowledge from event logs. However, as most business processes change over time (e.g. the effects of new legislation, seasonal effects and etc.), traditional process mining techniques cannot capture such "second-order dynamics" and analyze these processes as if they are in steady-state. Such changes can significantly impact the performance of processes. Hence, for the process management, it is crucial that changes in processes be discovered and analyzed. Process change detection is also known as business process drift detection.

All the existing methods for process drift detection are dependent on the size of windows used for detecting changes. Identifying convenient features that characterize the relations between traces or events is another challenge in most methods. In this thesis, we propose an automated and window-independent approach for detecting sudden business process drifts by introducing the notion of trace embedding. Using trace embedding makes it possible to automatically extract all features from the relations between traces.

* Corresponding author

*نویسندهٔ عهده‌دار مکاتبات

• تاریخ ارسال مقاله: ۱۳۹۷/۰۷/۱۸ • تاریخ پذیرش: ۱۳۹۸/۰۶/۱۱ • تاریخ انتشار: ۱۳۹۹/۱۲/۰۴ • نوع مطالعه: پژوهشی

[DOI: 10.29252/jsdp.17.4.33]

We show that the proposed approach outperforms all the existing methods in respect of its significantly higher accuracy and lower detection delay.

Keywords: **process mining, concept drifts, process changes, word embedding**

۱- مقدمه

فرایندکاوی[1] یک زمینه پژوهشی به‌نسبه جدید است که با استخراج دانش از نگاره‌های رویداد[2]، سعی در کشف[3]، بررسی انطباق[4] و بهبود[5] فرایندهای واقعی دارد [1]. دو دلیل اصلی برای افزایش پژوهش‌ها در حوزه فرایندکاوی وجود دارد. از یک‌سو، به‌کمک رویدادهای ثبت‌شده در نگاره رویداد می‌توان اطلاعات ارزشمندی درباره تاریخچه فرایندهای سازمانی به‌دست آورد. از سوی دیگر، با توجه به فضای رقابتی کسب‌وکار و تغییرات محیطی، نیاز به بهبود و پشتیبانی از فرایندهای کسب‌وکار اهمیت بسیاری دارد.

فرایندهای کسب‌وکار[6] را می‌توان از دیدگاه‌های[7] مختلف مانند کنترلِ جریان[8]، داده و منبع[9] مورد تحلیل قرار داد [2]. فرایندهای کسب‌وکار در دنیای واقعی پویا هستند؛ به‌گونه‌ای که در زمان‌های مختلف، گونه[10]های متفاوتی از یک فرایند اجرا می‌شود. وجود تغییرات بسیار در نرخ عرضه و تقاضا، تغییرات فصلی، وضع قوانین جدید، بلایای طبیعی و غیره ازجمله وقایعی هستند که سازمان‌ها را مجبور می‌سازد تا فرایندهای خود را تغییر دهند. این در حالی است که بیش‌تر روش‌های کشف فرایند، برای فرایندهای پایدار و ساخت‌یافته طراحی‌شده‌اند و مدل حاصل از آن‌ها برای فرایندهای دنیای واقعی یک مدل اسپاگتی شکل است. از طرفی دیگر، سازمان‌ها در تلاش هستند با ساده‌سازی فرایندهای کسب‌وکار، هزینه‌ها را کاهش داده و کارایی را بهبود بخشند. بنابراین، شناسایی تغییرات منتظره یا غیرمنتظره در رفتار فرایندها و تحلیل آن‌ها، اطلاعات ارزشمندی را برای ذی‌نفعان آشکار می‌سازد. رانش مفهومی[11] زمانی رخ می‌دهد که فرایند در هنگام تحلیل دچار تغییر شده باشد [1]. نیاز به مدیریت این تغییرات، پژوهش‌گران در حوزه فرایندکاوی را بر آن داشت تا روش‌های متنوعی را جهت شناسایی تغییرات فرایند ارائه دهند.

در سال‌های اخیر روش‌های متعددی برای شناسایی رانش مفهومی در نگاره‌های کسب‌وکار مطرح شده است که هرکدام بر روی دیدگاه (ها) و رانش‌های مختلفی تمرکز کرده‌اند. بیشتر روش‌های موجود نیازمند به استخراج ویژگی(ها) از دنباله[12]ها و رویدادها هستند. نوع ویژگی(های) انتخاب‌شده بر دقت روش‌ها تأثیرگذار است. از طرفی، همه روش‌های موجود از دو پنجره متوالی استفاده کرده و مقادیر ویژگی‌ها در این دو پنجره را به روش‌های مختلف مورد مقایسه قرار می‌دهند. نتایج ارائه‌شده در این روش‌ها نشان می‌دهد که دقت شناسایی رانش‌ها به‌شدت به نوع و اندازه پنجرهٔ استفاده‌شده وابسته است؛ بنابراین به‌نظر می‌رسد که می‌توان با ارائه روشی مستقل از پنجره که توانایی استخراج ویژگی‌ها به‌صورت خودکار را داشته باشد، چالش‌های پیشین را برطرف کرد. ایده پیشنهادی بدین‌صورت است که با توجه به نوع نگاره ورودی و با به‌کارگیری مفهوم تعبیه دنباله[13]، می‌توان دنباله‌های موجود در نگاره کسب‌وکار را به بردارهایی مدل کرد؛ به‌طوری‌که روابط میان بردارها در فضای برداری، بیان‌گر روابط میان دنباله‌ها در نگاره هستند؛ سپس می‌توان با استفاده از روش خوشه‌بندی، دنباله‌های هم‌رخداد را شناسایی کرده و نحوه توزیع آن‌ها را تعیین کرد؛ درنهایت، با استفاده از تبدیل فوریه و شناسایی تغییرات اصلی قادر خواهیم بود رانش‌ها و مکان دقیق آن‌ها را شناسایی کنیم.

نتایج ارزیابی حاکی از این است که روش پیشنهادی بهبود قابل‌توجهی در معیارهای امتیاز-F و متوسط تأخیر[14] نسبت به سایر روش‌ها ایجاد کرده است؛ بنابراین، نوآوری‌های این پژوهش به‌صورت زیر خلاصه می‌شود:

- ارائه ایده تعبیه دنباله به‌منظور استخراج خودکار روابط میان دنباله‌ها و رویدادها
- حذف پنجره به‌منظور شناسایی رانش‌ها و استفاده از معیار شباهت دنباله‌ها
- استفاده از تبدیل فوریه به‌منظور شناسایی مکان دقیق رانش و کاهش متوسط تأخیر

[1] Process mining
[2] Event log
[3] Discovery
[4] Conformance checking
[5] Enhancement
[6] Business process
[7] Prespective
[8] Control-flow
[9] Resource
[10] Variant
[11] Concept drift
[12] Trace
[13] Trace embedding
[14] Mean delay

[DOI: 10.29252/jsdp.17.4.33]

ادامه مقاله بدین صورت بخش‌بندی شده است. در بخش دوم، مرور کارهای گذشته مطرح می‌شود. در ابتدای این بخش به بیان مفاهیم اولیه پرداخته شده و سپس روش‌های موجود در شناسایی رانش فرایند بحث شده است. روش پیشنهادی در بخش سوم بیان شده است. بخش چهارم دربردارنده نتایج و ارزیابی‌های انجام‌شده بر روی نگاره‌های ساختگی و واقعی است؛ درنهایت، در بخش پنجم، نتیجه‌گیری و کارهای آینده بیان می‌شود.

۲- مرور کارهای گذشته

هدف از این بخش، بیان مفاهیم اولیه در حوزه فرایندکاوی، مفهوم رانش مفهومی و همچنین روش‌های مختلفی است که تاکنون در زمینه شناسایی رانش مفهومی در فرایندهای کسب‌وکار ارائه‌شده‌اند.

۲-۱- مفاهیم اولیه

نقطه شروع برای فرایندکاوی، وجود یک نگاره رویداد است. تمامی روش‌های فرایندکاوی بر این فرض هستند که امکان ثبت ترتیبی رویدادها به‌گونه‌ای که هر رویداد به یک فعالیت و به یک نمونه(مورد) خاص اشاره کند، وجود دارد. به‌عبارت دیگر، رویدادها به نمونه‌های فرایند مرتبط و از طریق نام فعالیت‌ها توصیف می‌شوند. رویدادها در یک نمونه فرایند مرتب هستند؛ بنابراین، یک نمونه فرایند، اغلب به‌صورت یک دنباله بر روی فعالیت‌ها نشان داده می‌شود؛ به‌علاوه، هر رویداد می‌تواند صفاتی مانند منبع (شخص یا دستگاه) آغازکننده فعالیت، برچسب زمانی رویداد، صفات داده‌ای (مقدار یا نوع مشتری) داشته باشد [3].

(رویداد و صفت): فرض کنید E مجموعه‌ای از رویداد[1]ها باشد. رویدادها با صفات متفاوتی مشخص می‌شوند. به‌عنوان‌مثال، هر رویداد می‌تواند یک برچسب زمانی، نام فعالیت[2]، نام منبع، هزینه و غیره داشته باشد. همچنین، فرض کنید AN مجموعه‌ای از نام صفت‌های مختلف باشد. برای هر رویداد $e \in E$ و نام $n \in AN$، $\#_n(e)$ نشان‌دهنده مقدار صفت n برای رویداد e است. اگر رویداد e، صفتی با نام n نداشته باشد، آن‌گاه $\#_n(e) = \perp$ خواهد بود [3].

(نمونه، دنباله، نگاره رویداد): فرض کنید C مجموعه‌ای از نمونه[3]ها است. نمونه‌ها نیز مانند رویدادها دارای صفات مختلفی هستند. برای هر نمونه $c \in C$ و هر نام $n \in AN$، $\#_n(c)$ نشان‌دهنده مقدار صفت n برای رویداد e است [3].

هر نمونه دارای یک صفت اجباری مخصوص به نام دنباله است: $\#_{trace}(c) \in E^*$. یک دنباله یک مجموعه محدود از رویدادها است، $\sigma \in E^*$.

نگاره رویداد یک مجموعه از نمونه‌های است، $L \subseteq C$.

۲-۲- رانش مفهومی

منظور از رانش مفهومی در یادگیری ماشین و داده‌کاوی، ایجاد تغییر پیش‌بینی‌نشده در رابطه بین داده ورودی و متغیر هدف در طول زمان است [4] یا به‌عبارت دیگر هر تغییر در نحوه توزیع داده در طی زمان را رانش مفهومی گویند.

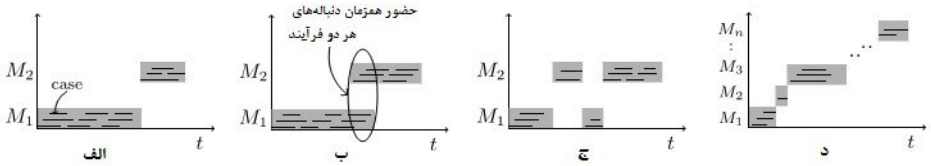
علاوه‌بر حوزه‌های یادگیری ماشین و داده‌کاوی، در سال‌های اخیر شناسایی رانش مفهومی در حوزه فرایندکاوی نیز مورد توجه و پژوهش قرار گرفته است. فرایندهای کسب‌وکار دنیای واقعی پویا هستند و در بازه‌های زمانی مختلف، گونه‌های اجرایی متفاوتی از یک فرایند اجرا می‌شوند. این در حالی است که روش‌های فرایندکاوی پایه، برای فرایندهای پایدار و ساخت‌یافته طراحی شده‌اند؛ بنابراین، مدل حاصل از این روش‌ها برای فرایندهای دنیای واقعی یک مدل اسپاگتی شکل است. مدل‌های اسپاگتی شکل، بسیار پیچیده هستند و به همین دلیل توصیف دقیق و واضحی از فرایند اصلی در حال اجرا ارائه نمی‌دهند؛ بنابراین، فرض ثابت بودن فرایند در حین اجرا، یک فرض غیرواقعی است و شناسایی تغییرات قابل‌انتظار و غیرقابل‌انتظار در رفتار فرایندها و تحلیل آن‌ها، اطلاعات ارزشمندی را برای ذی‌نفعان آشکار می‌سازد. در [1] از یازده موضوع به‌عنوان چالش‌های موجود در حوزه فرایندکاوی یاد شده که چهارمین چالش مطرح‌شده در آن، شناسایی رانش مفهومی در فرایندهای کسب‌وکار است و آن را به این صورت تعریف می‌کند:

"رانش مفهومی زمانی رخ می‌دهد که فرایند در هنگام تحلیل دچار تغییر شده باشد [1]."

همچنین، با توجه به آنچه در [5] گفته شده است، مسأله شناسایی رانش مفهومی در فرایندهای کسب‌وکار از دیدگاه آماری عبارت است از شناسایی نقطه‌ای از زمان که میان رفتار فرایند در قبل و بعد از این نقطه، اختلاف قابل‌ملاحظه آماری وجود داشته باشد.

[1] Event
[2] Activity
[3] Case

[DOI: 10.29252/jsdp.17.4.33]



(شکل-۱): انواع رانش. الف) رانش ناگهانی، ب) رانش تدریجی، ج) رانش متناوب، د) رانش افزایشی.[5]

(Figure-1): Different types of drifts. (a) Sudden drift. (b) Gradual drift. (c) Recurring drift. (d) Incremental drift

از روش‌های مطرح‌شده برای شناسایی رانش مفهومی در حوزه‌های یادگیری ماشین و داده‌کاوی نمی‌توان در فرایندکاوی استفاده کرد، چراکه تغییرات در این دو حوزه نسبت به فرایندکاوی، ساختار بسیار ساده‌تری دارند. ازجمله تغییراتی که در مدل فرایند کسب‌وکار ممکن است رخ دهد، می‌توان به هم‌زمانی[1] در مدل فرایند، حلقه‌ها[2]، انتخاب‌ها[3] و غیره اشاره کرد که این تغییرات، ساختار بسیار پیچیده‌تری نسبت به تغییرات در داده‌کاوی و یادگیری ماشین دارند [3]. با توجه به مدت‌زمانی که یک تغییر در فرایند وجود دارد، تغییرات را می‌توان در دو دستهٔ ناپایدار[4] و پایدار[5] طبقه‌بندی کرد. تغییرات ناپایدار، تغییراتی هستند که عمر کوتاهی دارند و فقط بر تعداد محدودی از نمونه‌ها تأثیر می‌گذارند؛ درحالی‌که تغییرات پایدار، دائمی هستند و برای یک مدت طولانی باقی می‌مانند [6]. شناسایی تغییرات ناپایدار به دلیل کمبود داده امکان‌پذیر نیست. به تغییرات ناپایدار، داده‌های پرت[6] یا نوفه نیز گفته می‌شود. به‌منظور شناسایی رانش‌های مفهومی باید تغییرات دائمی را شناسایی کرد [3]. چهار نوع رانش در ادامه بیان خواهد شد.

- **رانش ناگهانی**[7]: در این نوع رانش، یک فرایند جدید جایگزین فرایند موجود می‌شود. به‌عنوان‌مثال، سازمانی را در نظر بگیرید که به‌دلیل تغییر قوانین، باید فرایند جدیدی را جایگزین فرایند قبلی‌اش نماید. شکل (۱-الف)، نشان‌دهنده رانش ناگهانی است که در آن، فرایند M_2 جایگزین فرایند M_1 شده است.
- **رانش تدریجی**[8]: در این نوع رانش، یک فرایند جدید به‌صورت تدریجی جایگزین فرایند موجود می‌شود. همان‌طور که در شکل (۱-ب) نشان داده شده است، برخلاف رانش ناگهانی امکان حضور هم‌زمان هر دو فرایند وجود دارد. به‌عنوان‌مثال، سازمانی را در نظر بگیرد که فرایند تحویل محصولاتش را تغییر داده است. این سازمان برای مشتریانی که سفارش خود را قبل از تغییر فرایند ثبت کرده باشند از فرایند قدیمی و برای مشتریان جدید از فرایند جدید استفاده می‌کند.
- **رانش متناوب**[9]: مجموعه‌ای از فرایندها که بعد از مدت‌زمانی مجدد ظاهر می‌شوند. این تغییرات ممکن است به‌دلیل تغییرات فصلی به‌وجود آیند. به‌عنوان‌مثال، یک شرکت مسافرتی از فرایندی متفاوت در ایام عید نوروز استفاده می‌کند. همان‌طور که در شکل (۱-ج) مشاهده می‌شود فرایند M_2 در بازه‌های زمانی خاصی، جایگزین فرایند M_1 شده است.
- **رانش افزایشی**[10]: همان‌طور که در شکل (۱-د) مشاهده می‌شود، فرایند M_N با تغییرات کوچک و افزایشی جایگزین فرایند M_1 شده است. به‌عنوان‌مثال، می‌توان سازمانی را در نظر گرفت که برای بهبود عملکردش، تغییرات کم اما افزایشی و بلندمدت را بر روی فرایند اولیه خود اعمال می‌کند.

۳-۲- روش‌های شناسایی رانش فرایند

بیش‌تر روش‌های موجود، به‌منظور شناسایی روابط میان دنباله‌ها، نیاز به استخراج ویژگی دارند. چنانچه ویژگی‌های مناسبی انتخاب نشود، روش بیان‌شده توانایی شناسایی همه انواع تغییرات را نخواهد داشت و دقت آن کاهش می‌یابد؛ همچنین، همه روش‌های موجود به‌منظور شناسایی رانش فرایند از پنجره ثابت یا تطبیقی استفاده می‌کنند. استفاده از پنجره به این صورت است که دو پنجره متوالی بر روی جریانی از ویژگی‌های استخراج شده حرکت داده می‌شود و مقادیر آن‌ها به روش‌های گوناگون مورد مقایسه قرار می‌گیرد. اگرچه استفاده از پنجره تطبیقی، دقت را افزایش داده است، اما انتخاب اولیه اندازه پنجره، همچنان یک چالش مهم به‌شمار می‌آید.

[1] Concurrency
[2] Loops
[3] Choices
[4] Momentary
[5] Permanent
[6] Outlier
[7] Sudden drift
[8] Gradual drift
[9] Reccuring drift
[10] Incremental drift

[DOI: 10.29252/jsdp.17.4.33]

در [3,5] روشی به‌منظور شناسایی رانش‌های ناگهانی و انواع خاصی از رانش‌ها تدریجی ارائه شده است. ویژگی‌هایی به‌منظور شناسایی روابط میان دنباله‌ها، توسط کاربر استخراج می‌شود. در برخی موارد، به‌منظور انتخاب ویژگی مناسب، کاربر نیازمند دانش قبلی درباره نگاره است؛ سپس، از آزمون آماری، به‌منظور مقایسه ویژگی‌ها در دو پنجره متوالی استفاده می‌شود. روش ارائه‌شده در [3,5] دارای چند مشکل اصلی هستند. این روش به‌صورت خودکار عمل نمی‌کند. دقت روش به طور قابل ملاحظه‌ای تحت تاثیر اندازه پنجره انتخاب شده است. با توجه به نوع ویژگی انتخاب‌شده، ممکن است همه انواع رانش‌ها شناسایی نشوند.

روش مطرح‌شده در [3]، با بیان مفهوم پنجره تطبیقی قصد در توسعه متد پیشنهادی [7] را دارد. در این روش، دو مقدار کمینیه و بیشینه برای پنجره تنظیم می‌شود و اندازه کمینه پنجره تا زمانی که یک تغییر شناسایی شود یا به اندازه بیشینه برسد، افزایش می‌یابد. با توجه به این‌که، اندازه کمینه و بیشینه توسط کاربر تعیین می‌شود، دقت روش کاهش می‌یابد. به‌عبارت دیگر، اگر اندازه کمینهٔ تعیین‌شده خیلی کوچک باشد، نوفه‌ها را به‌عنوان رانش شناسایی می‌کند و اگر اندازه بیشینه خیلی بزرگ باشد، ممکن است برخی از رانش‌ها را شناسایی نکند. [7] با درنظرگرفتن پنجره‌هایی در مقیاس زمانی، قادر به شناسایی برخی از تغییرات تدریجی است. انتخاب بازه زمانی مناسب برای پنجره، چالش بعدی این روش به‌منظور شناسایی رانش تدریجی است.

الگوریتم بیان‌شده در [8] با استفاده از پنجره ثابت، قادر به شناسایی رانش‌های ناگهانی، تدریجی و متناوب است. در این روش، دنباله‌ها بر اساس فاصله بین زوج فعالیت‌هایشان خوشه‌بندی می‌شوند. پارامترهای مختلفی دقت این روش را تحت تأثیر قرار می‌دهند. از جمله مهم‌ترین پارامترها، اندازه پنجره استفاده شده است. همچنین، این روش برای نگاره‌هایی شامل گونه فرایندهای زیاد دقت مناسبی ندارد. با توجه به این‌که، این روش از فاصله بین زوج فعالیت‌ها به‌منظور شناسایی رانش فرایند استفاده می‌کند، آن دسته از رانش‌هایی را که منجر به تغییر در فاصله بین فعالیت‌ها نمی‌شوند شناسایی نمی‌کند.

روش ارائه‌شده در [9] تنها روشی است که هر دو دیدگاه کنترل جریان و داده را در نظر می‌گیرد. این روش از خوشه‌بندی به‌منظور شناسایی تغییرات ناگهانی استفاده می‌کند. خوشه‌بندی آن بر اساس الگوریتم خوشه‌بندی مارکوف است که از یک ماتریس شباهت به‌عنوان ورودی استفاده می‌کند. به‌منظور شناسایی نقاط تغییر، ماتریس شباهت به‌دست‌آمده از هر زیرنگاره در دو پنجره متوالی مورد مقایسه قرار می‌گیرد. از جمله نقاط ضعف این روش، استفاده از پنجره و نیازمندی به تنظیمات دستی است.

از مفهوم بردار پاریک[1] در [10] استفاده شده است. این بردار برای مجموعه‌ای از دنباله‌ها (دنباله‌های یادگیری) محاسبه شده و سپس یک چندضلعی با توجه به بردارهای تشکیل‌شده ایجاد می‌کند. زیرتوالی حاصل از هر دنباله بعدی، مدل و بررسی می‌شود که آیا در داخل چندضلعی قرار می‌گیرد یا خیر. چنانچه یک دنباله در داخل چندضلعی قرار بگیرد، یعنی متعلق به فرایند یکسان است؛ درحالی‌که اگر تعداد قابل توجهی از دنباله‌ها خارج از چندضلعی قرار بگیرند، نشان‌دهندهٔ وجود تغییر در فرایند است؛ یا به‌عبارت دیگر یک رانش شناسایی شده است. در این روش امکان شناسایی نقطه تغییر وجود ندارد و تنها می‌تواند رانش‌های ناگهانی را شناسایی کند. همچنین نیاز به محاسبات سنگینی دارد.

برخلاف روش ارائه‌شده در [3]، الگوریتم مطرح‌شده در [11] به ساختار توجه کرده است، احتمال را مدنظر قرار می‌دهد و از اتوماتای محدود قطعی احتمالی[2] (PDFA) استفاده می‌کند. روند کلی روش به این صورت است که زیرنگاره‌ها را به‌کمک دو پنجره متوالی متحرک استخراج، سپس از الگوریتم آلفا استفاده کرده و شبکه پتری مربوط به هر پنجره را تولید می‌کند. با توجه به این‌که شبکه پتری یک شبکه غیر احتمالی است، به‌منظور مقایسه با توزیع مدل پایه[3]، آن را به PDFA تبدیل می‌کند. با استفاده از آزمون آماری بررسی می‌کند که آیا نمایش PDFA (نحوه توزیع) هر زیرنگاره با مدل پایه به‌طور قابل‌ملاحظه‌ای تغییر کرده است یا خیر. دقت روش به مقدار داده مورد استفاده وابسته است. از طرفی مقدار زیاد داده باعث می‌شود که زمان اجرا افزایش یابد؛ بنابراین، ایجاد یک تعادل میان دقت و زمان اجرا یکی از چالش‌های این روش است. همچنین، این روش تنها توانایی شناسایی تغییرات بزرگ را دارد.

یک روش خودکار مبتنی بر مفهوم اجرا[4] در [12] مطرح شده است. در این روش به‌منظور شناسایی رانش، برای هر دنباله یک اجرا ساخته می‌شود؛ سپس با استفاده از

[1] Parikh
[2] Probabilistic determinestic finite automata
[3] Ground truth
[4] Run

[DOI: 10.29252/jsdp.17.4.33]

آزمون آماری، نحوه توزیع اجراها در دو پنجره تطبیقی متوالی مورد مقایسه قرار می‌گیرد. این روش تنها برای فرایندهای ساخت‌یافته طراحی شده و دقت آن به‌اندازه ابتدایی پنجره وابسته است.

الگوریتم مطرح شده در [13] سعی دارد با بهبود روش موجود در [12] رانش‌های تدریجی را شناسایی کند. ایده اصلی روش مطرح‌شده بر این اساس است که رانش‌های تدریجی به شکل دو رانش ناگهانی متوالی ظاهر می‌شوند. بنابراین، بعد از شناسایی رانش‌های ناگهانی با استفاده از روش [13]، یک آزمون آماری را بر روی سابقه‌نما[1] اجراها اعمال می‌کنند تا مشخص کنند که رانش‌های ناگهانی شناسایی‌شده، تغییرات جداگانه یا یک رانش تدریجی هستند. از جمله محدودیت‌های این روش، تغییر اندازه پنجره به منظور تعیین یک حد آستانه مناسب بین دقت و تأخیر شناسایی رانش است؛ همچنین، این روش برای نگاره‌ها با تعداد زیادی گونه متفاوت کارامد نیست.

روند کلی [14] به‌این‌ترتیب است که نگاره رویداد را به جریانی از دنباله‌ها تبدیل کرده و روابط میان دنباله‌ها در دو پنجره متوالی را با استفاده از معیار وابستگی استخراج کرده و آن‌ها را با استفاده از ماتریس روابط بیان می‌کند. معیار وابستگی استفاده‌شده در این روش کاوش‌گر ابتکاری است که نسبت به دیگر معیارهای شباهت عملکرد بهتری در مواجهه با نگاره‌های ناکامل دارد. در هر مرحله، ماتریس‌های حاصل از دو پنجره را مقایسه می‌کند. اگر ماتریس دارای اختلاف قابل‌ملاحظه باشد یعنی یک رانش اتفاق افتاده است. درواقع، با تبدیل نگاره به جریان، مسأله شناسایی رانش مفهومی به پیداکردن داده‌های پرت در جریان تبدیل شده است. با توجه به اینکه برای ساختن ماتریس روابط، می‌توان از انواع معیارهای وابستگی موجود در فرایندکاوی استفاده کرد، معیار r قابل‌تعمیم است [14]. مشکل اصلی این روش، همچون روش‌های پیشین، وابستگی آن به‌اندازه پنجره است.

روش مطرح شده در [15] بدین صورت است که دو پنجره متوالی تطبیقی را بر روی زیرنگاره‌ها حرکت می‌دهد. در هر گام، الگوریتم کشف فرایند ابتکاری[2] را اعمال کرده و مدل فرایند حاصل از دو پنجره را استخراج می‌کند. به‌منظور شناسایی رانش‌ها، معیارهای گراف را از پنجره‌ها استخراج کرده و آزمون آماری را بر روی آن‌ها اعمال می‌کند. اگر مدل فرایند حاصل از پنجره نخست با مدل فرایند حاصل از پنجره دوم خیلی متفاوت باشد، به این معناست که تغییر رخ داده است؛ همچنین، از طریق محاسبه و مقایسه معیارهای گراف، جزییاتی درباره ساختار رانش ارائه می‌دهند. همچون روش‌های پیشین، مشکل اصلی این روش تعیین اندازه کمینه بیشینه پنجره‌ها است. همچنین، با توجه به این‌که، الگوریتم ابتکاری قادر به شناسایی حلقه‌های طولانی نیست، این روش نمی‌تواند رانش‌های حاصل از حلقه‌های طولانی را شناسایی کند؛ بنابراین، می‌توان گفت که روش‌های موجود از چند مشکل اصلی رنج می‌برند. نخست این‌که، انتخاب ویژگی مناسب به‌منظور استخراج روابط میان دنباله‌ها یک چالش عمده به‌شمار می‌آید. چراکه اگر ویژگی مناسبی انتخاب نشود، انواع خاصی از تغییرات شناسایی نخواهند شد؛ همچنین، در برخی موارد، کاربر نیازمند دانشی قبلی درباره فرایند انجام‌شده در نگاره است. دوم این‌که، همه روش‌های موجود به‌منظور شناسایی رانش فرایند از پنجره ثابت یا تطبیقی استفاده می‌کنند. بنابراین، دقت این روش‌ها به‌شدت تحت تأثیر اندازه پنجره انتخاب شده است. اگرچه استفاده از پنجره تطبیقی بهبودی در دقت روش‌ها ایجاد کرده است، اما تعیین اندازه ابتدایی پنجره همچنان یک چالش بزرگ به‌شمار می‌آید.

در بخش بعد، به‌منظور برطرف‌کردن چالش‌های موجود، یک روش جدید مستقل از پنجره ارائه خواهیم کرد. در این روش، با مطرح‌کردن ایده تعبیه دنباله سعی در شناسایی رانش‌های ناگهانی در دیدگاه کنترلِ جریان داریم. با استفاده از تعبیه دنباله، می‌توانیم روابط میان دنباله‌ها را به‌طور خودکار استخراج کنیم. روش پیشنهادی، دقت روش‌های موجود را بهبود خواهد داد.

۳- روش پیشنهادی

در این بخش با ارائه روش پیشنهادی، سعی کرده‌ایم محدودیت‌های روش‌های موجود را برطرف کرده و دقت شناسایی رانش فرایند را افزایش دهیم. هدف اصلی ما ارائه روشی خودکار به‌منظور شناسایی رانش‌های ناگهانی در دیدگاه کنترل جریان با به‌کارگیری مفهوم تعبیه دنباله است. در ادامه بیان خواهد شد که مفهوم تعبیه دنباله الهام‌گرفته‌شده از مفهوم تعبیه واژه[3] است. با درنظرگرفتن نگاره رویداد به‌عنوان متن ورودی و دنباله‌های فرایند به‌عنوان واژه‌های فرهنگ لغت، این مفهوم به‌راحتی قابل تفسیر است. ازاین‌رو، روش پیشنهادی قادر است مستقل از

[1] Histogram
[2] Heuristic miner
[3] Word embedding

[DOI: 10.29252/jsdp.17.4.33]

پنجره (تطبیقی یا ثابت) همه انواع روابط میان دنباله‌ها را به‌صورت خودکار استخراج کرده و دقت روش را به‌منظور شناسایی رانش فرایند در مقایسه با روش‌های موجود افزایش دهد.

در ادامه ابتدا مفاهیم اصلی در حوزه تعبیه واژه مطرح می‌شود و سپس ساختار روش پیشنهادی و اجزای آن با جزییات گفته خواهد شد.

۱-۳- تعبیه واژه

در تعبیه واژه، تمام واژگان موجود در فرهنگ لغت به بردار مدل می‌شوند. دو نوع تعبیه واژه وجود دارد [16]. ۱) تعبیه مبتنی بر تکرار [17]، ۲) تعبیه مبتنی بر پیش‌بینی [18]، [19]. بردارهای مبتنی بر تکرار، بر اساس هم‌رخدادی یک واژه با گروهی خاصی از واژگان دیگر ساخته می‌شود؛ بنابراین، بردارهای تولیدشده پراکندگی[1] زیاد و همچنین ابعاد بالایی دارند [20]؛ اما در روش‌های مبتنی بر پیش‌بینی، بردارهای تولیدشده متراکم[2] و با ابعاد کم هستند. هدف اصلی روش‌های مبتنی بر پیش‌بینی این است که احتمال ظاهرشدن توالی خاصی از واژگان در یک متن را تخمین بزنند. در این مدل فرض شده است که واژگان نزدیک در یک توالی، از نظر آماری وابستگی بیشتری نسبت به یکدیگر دارند. یکی از مشهورترین روش‌های مبتنی بر پیش‌بینی، تعبیه واژه است [21]. تعبیه واژه یک شبکه عصبی دولایه است که متن را پردازش می‌کند. ورودی آن یک پیکره خام و خروجی آن یک فرهنگ لغت است که هر واژه به یک بردار مدل شده است. بردارهای واژگان مشابه به فضای برداری مشابهی نگاشت داده می‌شوند [22]. به‌عنوان‌مثال، واژه تهران و رم در فضای برداری نزدیک به هم قرار می‌گیرند؛ درحالی‌که بردار دو واژه تهران و سیب از هم دور است. دو الگوریتم اصلی برای تعبیه واژه وجود دارد: کیف لغات پیوسته[3] و Skip-gram [23].

•کیف لغات پیوسته

مدل CBOW با استفاده از واژگان متن موجود در یک پنجره معین، واژگان هدف را پیش‌بینی می‌کند. معماری این مدل در شکل (۲) نشان داده شده است. لایه ورودی، از واژگان متن موجود در پنجره تشکیل شده است که در آن بردارهای ورودی از ماتریس وزن ورودی بازیابی و میانگین‌گیری شده و به لایه تجسم[4] نگاشت داده می‌شوند؛ سپس با استفاده از وزن‌های ماتریس وزن خروجی، برای هر واژه یک امتیاز که نشان‌دهنده احتمال هدف‌بودن آن واژه است، محاسبه می‌شود. یک توالی از واژگان تعلیم $w_1, w_2, \ldots w_T$ و پنجره c را در نظر بگیرید. هدف مدل CBOW بیشینه‌کردن احتمال میانگین لگاریتم رابطه زیر است [24]:

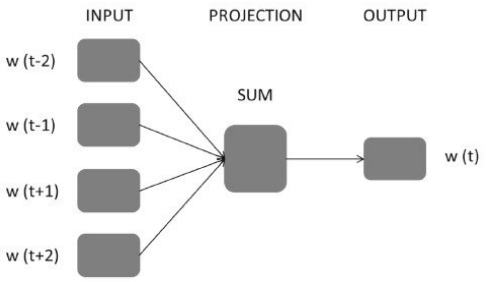
$$\frac{1}{T}\sum_{t=1}^{T}\log p(w_t \mid w_{t-c}\cdots w_{t+c}) \qquad (۱)$$

که احتمال $p(w_t \mid w_{t-c}\cdots w_{t-c})$ با استفاده از تابع softmax (یک مدل طبقه‌بندی log-linear) محاسبه می‌شود:

$$p(w_t \mid w_{t-c}\cdots w_{t+c}) = \frac{\exp(\bar{v}^T v'_{w_t})}{\sum_{w=1}^{V}\exp(\bar{v}^T v'_w)} \qquad (۲)$$

که v'_w بردار خروجی واژه w، V فرهنگ لغات واژگان و $\bar{v}$ بردار ورودی میانگین گرفته‌شده از همه واژگان متن است [24]:

$$\bar{v} = \frac{1}{2c}\sum_{-c\le j\le c,\, j\ne 0} v_{w_{t+j}} \qquad (۳)$$



(شکل-۲): معماری مدل CBOW

(Figure-2): Architreture of the CBOW model

•مدل *Skip-gram*

این مدل برخلاف مدل CBOW سعی در پیش‌بینی واژگان متن با استفاده از واژگان هدف دارد. معماری آن در شکل (۳) نشان داده شده است. یک توالی از واژگان تعلیم $w_1, w_2, \ldots w_T$ و پنجره c را در نظر بگیرید. هدف مدل Skip-gram بیشینه‌کردن احتمال میانگین لگاریتم رابطه زیر است [24]:

[1] Sparse
[2] Dense
[3] Continuous bag-of-word
[4] Projection

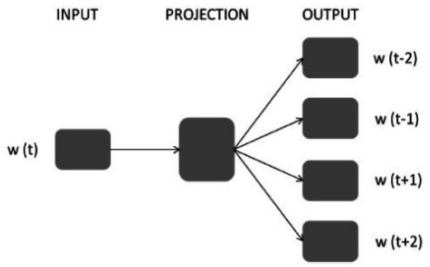
[DOI: 10.29252/jsdp.17.4.33]

$$\frac{1}{T}\sum_{t=1}^{T}\sum_{-c\le j\le c, j\ne 0}\log p(w_{t-j}\mid w_t) \qquad (۴)$$

که احتمال $p(w_{t+j}\mid w_t)$ با استفاده از تابع softmax محاسبه می‌شود:

$$p(w_o\mid w_i)=\frac{\exp(\bar{v}_{w_o}{}^{T} v'_{w_i})}{\sum_{w=1}^{V}\exp(v'_{w}{}^{T} v_{w_i})} \qquad (۵)$$

که v_w و v'_w بردارهای ورودی و خروجی واژه w و V فرهنگ لغت از واژگان است.



(شکل-۳): معماری مدل Skip-gram

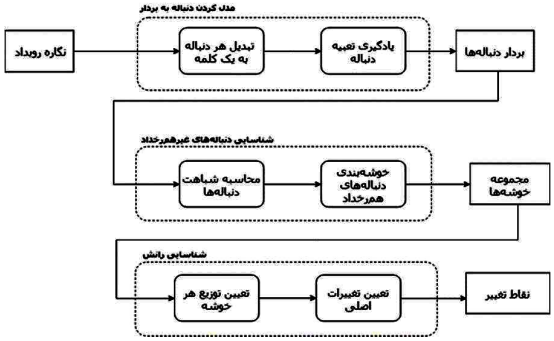
(Figure-3): Architrcture of the Skip-gram model

۲-۳- ساختار روش پیشنهادی

همان‌طور که پیش‌تر گفته شد، روش پیشنهادی یک روش مستقل از پنجره و مستقل از نوع فرایند ورودی است که می‌تواند به‌صورت خودکار روابط میان دنباله‌ها و رویدادها را استخراج کند و دقت شناسایی رانش فرایند در نگاره‌های کسب‌وکار را افزایش دهد. شکل (۴) روندنمای روش پیشنهادی را نشان می‌دهد. روش پیشنهادی از سه گام اصلی *مدل‌کردن دنباله‌ها به بردار*، *شناسایی دنباله‌های غیرهم‌رخداد* و *شناسایی رانش* تشکیل شده است که در ادامه هر گام با جزییات بیان می‌شود.

۱-۲-۳- مدل‌کردن دنباله‌ها به بردار

بیش‌تر روش‌های موجود به‌منظور شناسایی رفتار فرایند و مشخص‌کردن روابط بین فعالیت‌ها، نیاز به تعریف مجموعه‌ای از ویژگی‌ها دارند. تعداد و نوع ویژگی‌های انتخاب‌شده در دقت روش‌ها تأثیرگذار و حتی در بعضی از موارد نیاز است که کاربر از یک دانش قبلی نسبت به فرایند در حال اجرا بهره‌مند باشد تا ویژگی مناسبی را انتخاب کند. به‌منظور حذف این محدودیت، ما مفهوم تعبیه دنباله را مطرح می‌کنیم که قادر است به‌طور خودکار تمام انواع روابط میان دنباله‌ها را استخراج نماید.



(شکل-۴): روندنمای روش پیشنهادی

(Figure-4): Architrcture of the proposed approach

الف) تبدیل هر دنباله به یک واژه

با درنظرگرفتن فایل نگاره به‌عنوان متن و دنباله‌های فرایند به‌عنوان واژگان و با استفاده از روش تعبیه واژه، می‌توان ایده تعبیه دنباله را تفسیر کرد. هدف از ایده تعبیه دنباله، تولید بردار برای دنباله‌های موجود در نگاره رویداد است؛ به‌طوری‌که روابط میان برداها در فضای برداری نشان‌دهنده روابط میان دنباله‌ها در نگاره باشد.

به‌منظور به‌کارگیری ایده تعبیه دنباله، ابتدا باید فایل رویداد را آماده کرد؛ ازاین‌رو، هر دنباله در نگاره را با حذف فاصله میان فعالیت‌هایش به یک واژه تبدیل می‌کنیم. به‌عنوان‌مثال، دنباله مربوط به نمونه ۱ در شکل (۵) به‌صورت [abcdefgh] تبدیل می‌شود؛ سپس جریانی از دنباله‌ها تولید می‌کنیم که در آن دنباله‌ها بر اساس برچسب زمانی نخستین رویدادشان مرتب شده‌اند.

ب) یادگیری تعبیه دنباله

تعبیه دنباله با یادگیری[1] جریان تولیدشده، برای هر دنباله (یک‌واژه‌ای) یک بردار از مقادیر عددی در فضای ویژگی تولید می‌کند. درنتیجه، با کمک بردارهای تولیدشده، می‌توان جریان دنباله را به جریانی از بردارها با ابعاد یکسان تبدیل کرد که بیان‌گر اطلاعاتی درباره آمارهای هم‌رخدادی دنباله‌ها است.

شناسایی دنباله‌های غیرهم‌رخداد

همان‌طور که پیش‌تر گفته شد، در رانش ناگهانی فرایند، نمونه‌های فرایند جدید جایگزین نمونه‌های فرایند قبلی می‌شوند؛ بنابراین، دنباله‌های غیر هم‌رخداد، دنباله‌هایی هستند که منجر به وقوع رانش شده‌اند.

[1] Train

[DOI: 10.29252/jsdp.17.4.33]

Case id	Activity	Time stamp
1	a	2004/02/11 13:00:00.000
	b	2004/02/11 13:25:28.91
	c	2004/02/11 13:28:18.693
	d	2004/02/11 13:42:06.935
	e	2004/02/11 14:52:58.531
	f	2004/02/11 15:00:07.055
	g	2004/02/11 15:06:39
	h	2004/02/11 15:06:39.125
. . .	. . .	. . .
n	a	2004/03/02 17:30:00.000
	b	2004/03/02 18:00:06.067
	r	2004/03/02 18:03:27.651
	s	2004/03/02 18:18:29.374
	b	2004/03/02 18:49:11.161
	c	2004/03/02 19:31:05.327
	d	2004/03/02 19:38:43.799
	e	2004/03/02 19:44:37.811
	f	2004/03/02 19:59:02.032
	g	2004/03/03 13:37:11.601
	h	2004/03/03 13:37:11.601

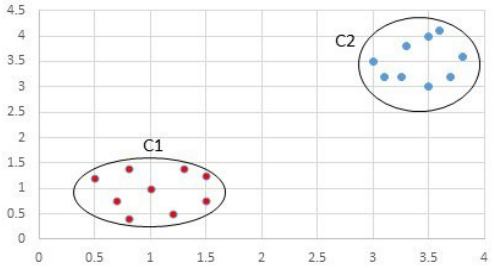
(شکل-۵): قسمتی از یک نگاره
(Figure-5): A fragment of an event log

الف) محاسبه شباهت دنباله

با محاسبه شباهت برداری میان دنباله‌های نگاره و شناسایی دنباله‌ها با کمترین شباهت یا بیشترین فاصله می‌توان دنباله‌های غیر هم‌رخداد را مشخص کرد. به‌عبارت‌دیگر، هر تغییر رفتار فرایند از A به B منجر به افزایش فاصله یا کاهش هم‌رخدادی در میان مجموعه نمونه‌های فرایند A و مجموعه نمونه‌های فرایند B می‌شود و درنتیجه بردار دنباله‌ها قبل از نقطه تغییر (وقوع رانش) با بردار دنباله‌ها بعد از آن نقطه متفاوت است. به‌منظور شناسایی مجموعه‌دنباله‌های غیر هم‌رخداد، از شباهت کسینوسی میان بردارها استفاده شده است.

ب) خوشه‌بندی دنباله‌های هم‌رخداد

در مرحله بعد، نیاز داریم از میان مجموعه‌دنباله‌های غیر هم‌رخداد شناسایی‌شده، مجموعه‌هایی هم‌رخداد را شناسایی کنیم. علت این امر که چرا از ابتدا مجموعه‌دنباله‌های هم‌رخداد شناسایی نمی‌شوند این است که ابتدا باید دنباله‌هایی را که در وقوع رانش تأثیرگذار نیستند، حذف کنیم؛ بنابراین، پس از شناسایی مجموعه‌های غیر هم‌رخداد، به‌منظور شناسایی مجموعه دنباله‌ها با بیشترین هم‌رخدادی از خوشه‌بندی سلسله‌مراتبی استفاده می‌کنیم. در ابتدا هر دنباله در یک خوشه قرار دارد و در هر مرحله دو خوشه با بیشترین هم‌رخدادی ادغام می‌شوند؛ درنهایت، خروجی این مرحله مجموعه‌ای از خوشه‌ها است که اعضای هر خوشه در فضای برداری مشابه و در مقابل، در فضای برداری نامشابه با تمام اعضای خوشه‌های دیگر قرار دارند؛ همچنین، هر دنباله تنها می‌تواند در یک خوشه قرار بگیرد و اشتراک خوشه‌ها تهی است. اگر فرض کنیم که بردارها دوبعدی هستند و خروجی خوشه‌بندی سلسله‌مراتبی دو خوشه C1 و C2 است، همان‌طور که در شکل (۶) نشان داده شده است، همه اعضای خوشه C1 در فضای ویژگی یکسان و درعین‌حال متفاوت با اعضای خوشه C2 قرار دارند.



(شکل-۶): تصویرسازی دوبعدی از فضای ویژگی. با توجه به این شکل اعضای هر خوشه در فضای ویژگی مشابه قرار دارند.
(Figure-6): Two-dimensional visualization of feature space. The figure illustrates that the members of each cluster are close to each other.

شناسایی رانش

هدف از این مرحله، استفاده از خوشه‌های تولیدشده به‌منظور شناسایی رانش فرایند و تعیین مکان دقیق وقوع آن است. در ادامه به‌منظور شناسایی رانش‌ها با بیشترین دقت ممکن، روش‌های خاصی اعمال شده است.

الف) تعیین توزیع هر خوشه

ابتدا برای هر خوشه یک بردار توزیع تولید می‌کنیم که نشان‌دهنده نحوه توزیع دنباله‌های هر خوشه در نگاره رویداد است. طول بردارهای توزیع، برابر با تعداد نمونه‌های نگاره است که یک به معنای حضور و صفر به معنای عدم حضور اعضای یک خوشه در بردار توزیع آن است. شکل ۷ نحوه توزیع دو خوشه C1 و C2 را نشان می‌دهد که نواحی متراکم نشان‌دهنده موقعیت دنباله‌های هر خوشه در نگاره است. طول بردارهای توزیع در شکل (۷) برابر با ۲۵۰۰ است یا به‌عبارت دیگر نگاره دارای ۲۵۰۰ نمونه است.

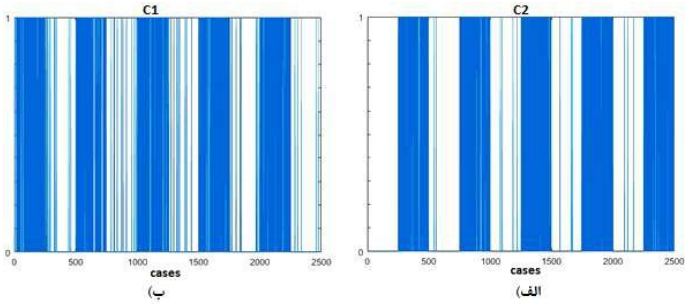
ب) تعیین تغییرات اصلی

به‌منظور تعیین مکان‌های پرتکرار، بردارهای توزیع را به فضای فوریه منتقل می‌کنیم. درحقیقت، با به‌کارگیری تبدیل فوریه می‌توانیم یک سیگنال را به فرکانس‌های مختلف آن تجزیه کنیم [25].

با توجه به این‌که فرکانس‌های پایین نشان‌دهنده تغییرات اصلی در یک سیگنال و فرکانس‌های بالا به‌طورعمومی نوفه هستند، پس با اعمال یک فیلتر پایین‌گذر[1] می‌توان تغییرات اصلی را شناسایی و نوفه‌ها را حذف کرد. در اینجا منظور از نوفه، دنباله‌هایی هستند که به‌اشتباه در یک خوشه قرارگرفته‌اند.

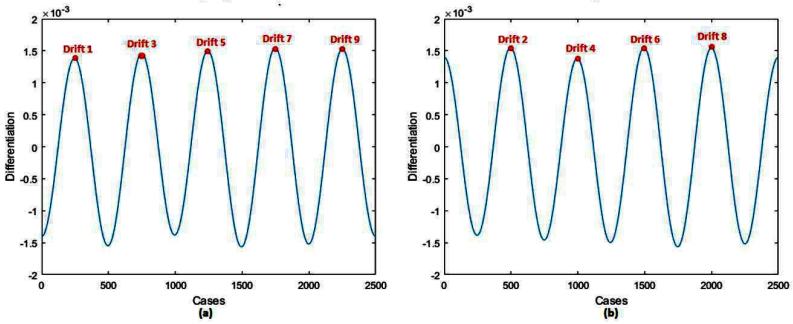
[1] Low pass filter

[DOI: 10.29252/jsdp.17.4.33]



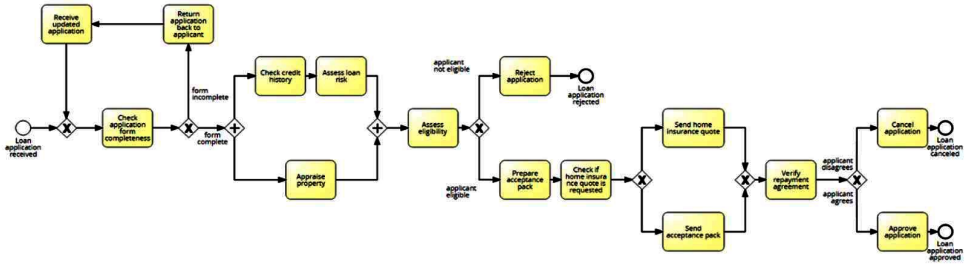
(شکل-۷): نحوه توزیع هر خوشه در نگاره

(Figure-7): Distribution of each cluster in a log



(شکل-۸): الف) نرخ تغییرات در بردار توزیع خوشه C2. ب) نرخ تغییرات در بردار توزیع خوشه C1. قله‌ها نشان‌دهنده تغییر هستند.

(Figur-8): (a) Rate of changes in the distribution vector of cluster C1. (b) Rate of changes in the distribution vector of cluster C2. Remarkable peaks are change points.



(شکل-۹):مدل BPMN مربوط به فرایند وام [12]

(Figure-9): Base BPMN model of loan application process

پس از اعمال فیلتر پایین‌گذر، بردارهای توزیع تغییریافته را به فضای زمان برگردانده و برای تشخیص نرخ تغییرات از آن مشتق می‌گیریم. نقطه‌ها با بیشترین نرخ تغییر یا به‌عبارت‌دیگر قله‌ها نشان‌دهنده مکان وقوع رانش فرایند در نگاره هستند. شکل (۸) خروجی مشتق گرفته‌شده از بردارهای توزیع شکل (۷) را نشان می‌دهد که نقاط قله نشان‌دهنده مکان وقوع رانش هستند. بنابراین، نقاط ۲۵۰، ۵۰۰، ۷۵۰، ۱۰۰۰، ۱۲۵۰، ۱۵۰۰، ۱۷۵۰، ۲۰۰۰، ۲۲۵۰ به‌عنوان شاخص دنباله‌هایی که در آن‌ها رانش رخ داده است شناسایی می‌شوند؛ بنابراین با ارائه روشی جدید در شناسایی رانش مفهومی، توانستیم با حل مشکلات پیشین، تغییرات را شناسایی کنیم.

۴- ارزیابی نتایج

به‌منظور ارزیابی روش پیشنهادی از مجموعه‌دادهٔ مطرح‌شده در [12] استفاده شده است[1]. [12] ۷۲ نگاره با ساختار متفاوت ارائه داده است. شکل (۹) مدل فرایند پایه در همه نگاره‌ها را نشان می‌دهد. این مدل مربوط به فرایند درخواست وام است که شامل پانزده فعالیت، یک رویداد آغازین و سه رویداد خاتمه است.

به‌منظور ایجاد رانش در این نگاره‌ها، دوازده الگو تغییر متفاوت به مدل فرایند پایه اعمال شده است. الگوهای تغییر در جدول (۱) نشان داده شده است. این الگوها، بیان‌کننده تغییرات ممکن بر مدل فرایند کسب‌وکار هستند مانند، اضافه یا حذف یا حلقوی‌کردن یک قطعه مدل،

[1] http://apromore.org/platform/tools

[DOI: 10.29252/jsdp.17.4.33]

جابه‌جاکردن دو قطعه مدل، یا موازی‌سازی دو قطعه متوالی. [12] الگوهای تغییر را در سه دسته افزودن (I)، تغییر ترتیب (R) و اختیاری (O) تقسیم‌بندی کرده و با ترکیب‌کردن آن‌ها شش الگو تغییر پیچیده‌تر را تولید می‌نماید. به‌عنوان‌مثال، الگوی ترکیبی IOR را می‌توان با اضافه‌کردن یک فعالیت جدید (I)، موازی قراردادن این فعالیت با یک فعالیت موجود (O) و درنهایت قراردادن آن در ساختار حلقوی (R) تولید کرد.

(جدول-۱): الگوهای تغییر در دیدگاه کنترل جریان
(Table-1): Simple control-flow change patterns

دسته	الگو تغییر	کد
I	حذف یا اضافه‌کردن قطعه	re
R	قراردادن دو قطعه به‌صورت شرطی/متوالی	cf
O	قراردادن یا خارج کردن قطعه در/ از حلقه	lp
R	قراردادن دو قطعه به‌صورت متوالی/موازی	pl
O	اضافه‌کردن یا حذف قابلیت پرش از قطعه	cb
I	قراردادن یا خارج‌کردن قطعه در/ از شاخه شرطی	cm
R	همگام‌سازی دو قطعه	cd
I	تکرار قطعه	cp
I	قرارداد یا خارج‌کردن قطعه در/ از شاخه موازی	pm
I	جایگزینی قطعه	rp
I	جابه‌جاکردن دو قطعه	sw
O	تغییر تکرار شاخه‌ها	fr

[12] به‌منظور قرار دادن رانش‌ها در فواصل متفاوت، ۴ نگاره با ۲۵۰۰، ۵۰۰۰، ۷۵۰۰ و ۱۰۰۰۰ دنباله برای مدل فرایند پایه و برای هر یک از هجده الگو تغییر تولید کرده که درمجموع برابر با ۷۲ نگاره خواهد شد. در هر نگاره ۹ رانش، در هر ده درصد از اندازه آن، قرار داده شده است. با آگاهی از تعداد و مکان هر رانش در نگاره‌ها، می‌توان یک استاندارد طلایی تولید و دقت روش را با آن ارزیابی کرد.

گفتنی است که هدف از ایجاد این نگاره‌های ساختگی تولید مجموعه‌ای از داده‌های آموزش و آزمون نیست؛ بلکه به‌منظور ارزیابی دقت روش ارائه‌شده لازم است نگاره‌هایی تولید شود که دربردارنده انواع تغییرات ممکن باشند. ازاین‌رو، هر نگاره به‌صورت جداگانه و به‌صورت تدرجی آموزش داده و دقت عملکرد روش پیشنهادی در مواجهه با هر نگاره به‌صورت جداگانه با روش‌های دیگر مقایسه می‌شود.

دو معیار امتیاز-F و متوسط تأخیر به‌عنوان معیارهای ارزیابی دقت در روش‌های شناسایی رانش مفهومی در حوزه داده‌کاوی ارائه شده‌اند [26]. این معیارها برای ارزیابی روش‌های موجود در حوزه فرایندکاوی نیز استفاده شده و لذا امکان مقایسه روش پیشنهادی را فراهم می‌آورد. معیار امتیاز-F، متوسط هارمونیک دقت[1] و فراخوانی[2] را اندازه می‌گیرد که به‌صورت زیر محاسبه می‌شوند.

$$precision = \frac{TP}{TP + FP} \quad (۶)$$

$$recall = \frac{TP}{TP + FN} \quad (۷)$$

$$F-score = \frac{2 \times precision \times recall}{precision + recall} \quad (۸)$$

منظور از مثبت صحیح[3]، منفی کاذب[4] و مثبت کاذب[5] در هر حوزه متفاوت است. در زمینه شناسایی رانش فرایند این مفاهیم به‌صورت زیر تعریف می‌شوند:

- مثبت صحیح (TP): تعداد نقاطی که به‌درستی به‌عنوان رانش شناسایی شده‌اند.
- منفی کاذب (FN): تعداد نقاطی که به‌عنوان رانش شناسایی نشده‌اند، اما در واقعیت وجود دارند.
- مثبت کاذب (FP): تعداد نقاطی که به‌عنوان رانش شناسایی‌شده‌اند، اما در واقعیت وجود ندارند.

متوسط تأخیر معیار دیگری برای ارزیابی دقت است که بیان‌کنندهٔ متوسط تعداد دنباله‌ها میان نقطه واقعی تغییر و نقطه شناسایی شده است. هر چه متوسط تأخیر یک روش کمتر باشد، عملکرد آن در شناسایی رانش فرایند بهتر خواهد بود.

در این پژوهش از روش ارزیابی مقایسه‌ای استفاده شده است. از میان روش‌های یادشده در بخش کارهای گذشته، روش‌های مطرح‌شده در [3, 15, 12] جهت مقایسه و ارزیابی دقت روش پیشنهادی استفاده شده‌اند. با توجه به این‌که ارزیابی انجام‌شده در این روش‌ها، بر روی مجموعه‌دادهٔ یکسانی است، نیاز به پیاده‌سازی مجدد آن‌ها نبود. همان‌طور که پیش‌تر گفته شد، این مجموعه‌داده یکسانی، مجموعه‌داده ارائه‌شده در [12] است.

[1] Precision
[2] Recall
[3] True positive
[4] False negative
[5] False positive

[DOI: 10.29252/jsdp.17.4.33]

معیارهای امتیاز-F و متوسط تأخیر برای هر یک از الگوهای تغییر محاسبه شده است. نتایج دقیق این معیارها برای هر الگو در روش پیشنهادی به‌دست آمده و به‌همراه نتایج روش‌های دیگر در جدول (۲) نشان داده شده است. نتایج این جدول به ازای هر الگوی تغییر بر روی چهار نگاره متفاوت میانگین گرفته شده است. نمودارهای شکل‌های (۱۰ و ۱۱) نیز به‌ترتیب مقایسه روش پیشنهادی با روش‌های پایه را از نظر امتیاز-F و متوسط تأخیر به‌ازای هر الگو تغییر (میانگین گرفته‌شده بر روی چهار نگاره) نشان می‌دهند. همان‌طور که در نمودار شکل (۱۰) مشاهده می‌کنید، روش پیشنهادی به امتیاز-F برابر یک برای همهٔ الگوها به‌جز OIR دست یافته است که در مقایسه با روش‌های پایه بهبود فراوانی را نشان می‌دهد. همچنین، روش پیشنهادی در سیزده الگو تغییر به تأخیر کمتر از بیست دنباله، در سه الگو تغییر به تأخیر کمتر از سی دنباله و تنها در دو الگو تغییر به تأخیر بیشتر از سی دنباله دست یافته که در مقایسه با روش‌های دیگر، در شانزده الگو، متوسط تأخیر را در شناسایی مکان رانش به‌طور مناسبی کاهش داده است. به‌طور میانگین، امتیاز-F تقریبی به‌دست‌آمده برای ۷۲ نگاره مورد بررسی برابر با ۰/۹۹، ۰/۹۷، ۰/۹۴ و ۰/۷ به‌ترتیب برای روش پیشنهادی، run، process-graph و Bose است که روش پیشنهادی بهبود قابل توجهی نسبت به روش‌های دیگر دارد؛ همچنین، متوسط تأخیر تقریبی به‌دست‌آمده برای ۷۲ نگاره مورد بررسی، به‌طور میانگین برابر با ۱۳، ۳۲، ۲۴ و ۴۷ دنباله به‌ترتیب برای روش پیشنهادی، run، process-graph و Bose است. به‌عبارت دیگر، به‌طور میانگین، مکان رانش شناسایی‌شده توسط روش پیشنهادی حدود سیزده دنباله با مکان واقعی رانش فاصله دارد که سه روش دیگر را بهبود داده است.

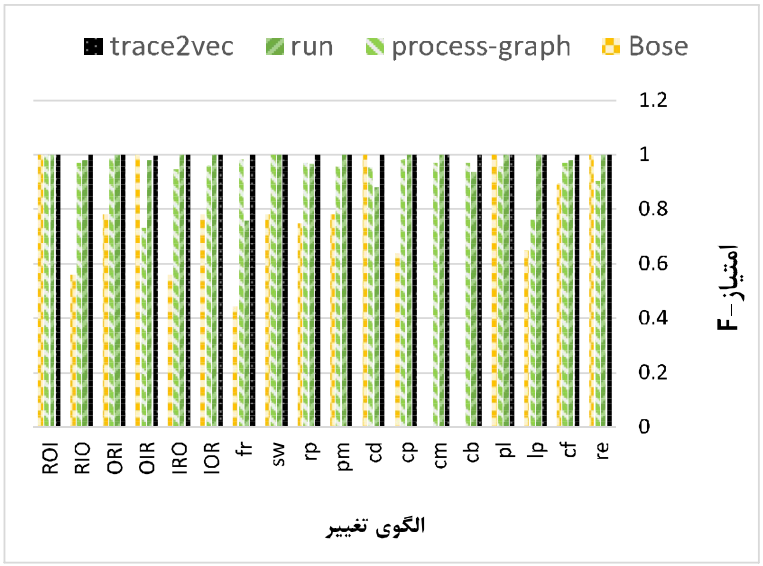
در آزمایش بعدی، روش پیشنهادی را بر روی نگاره دنیای واقعی اعمال کردیم. نگاره مورد استفاده مربوط به پنج منطقه مختلف از شهرداری مشهد است. در ابتدا، به‌منظور بالابردن کیفیت تحلیل، پیش‌پردازش‌هایی برای مدیریت نوفه‌ها، مقادیر ازدست‌رفته و همچنین ناسازگاری‌ها بر روی پایگاه داده انجام شد.

(جدول-۲): مقادیر دقیق روش پیشنهادی از دو دیدگاه امتیاز-F و متوسط تأخیر در مقایسه با روش‌های run، process-graph و Bose.

(Table-2): Comparison of average F-score and mean delay values

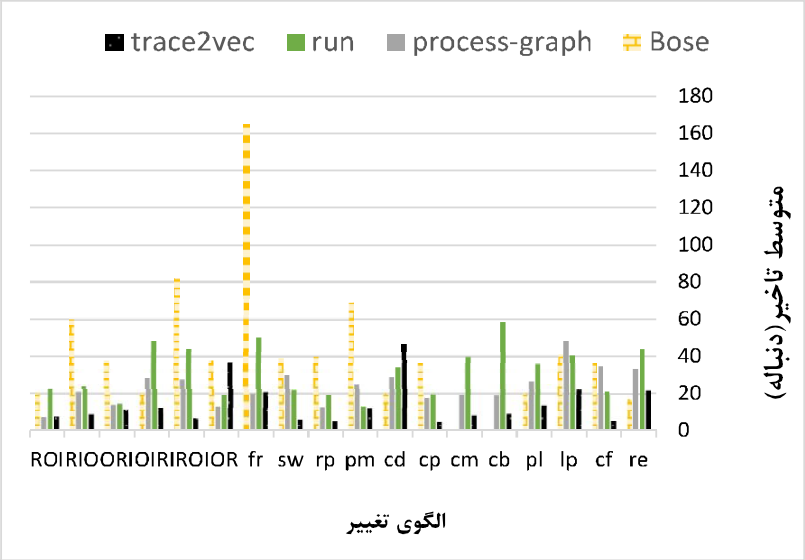
الگو تغییر	trace2vec		run[۱۲]		process graph [۱۵]		Bosc[۳]	
	امتیاز-F	*متوسط تأخیر*	*امتیاز-F*	*متوسط تأخیر*	*امتیاز-F*	*متوسط تأخیر*	*امتیاز-F*	*متوسط تأخیر*
Re	1	21.83	1	44.03	0.9036	33.02	1	17
Cf	1	5.08	0.9824	21	0.9853	34.62	0.8950	36
Lp	1	21.69	1	40.29	0.7618	48.03	0.6484	41
Pl	1	13.8	1	35.74	0.9575	26.33	1	20
Cb	1	9.1	0.9387	58.55	0.9722	18.94	0	0
Cm	1	8.52	1	39.85	0.9722	19.24	0	0
Cp	1	4.58	1	19.66	0.9853	17.59	0.6394	36
Cd	1	46.44	0.8799	34.62	0.9546	28.62	1	20
Pm	1	11.97	1	12.88	0.9869	24.78	0.7804	69
Rp	1	4.86	0.9666	19.18	0.9722	12.67	0.75	40
Sw	1	6.02	1	21.67	1	29.61	0.7804	39
Fr	1	20.5	0.7569	49.92	0.9853	19.92	0.4420	165
IOR	1	36.66	1	19.11	0.9606	13	0.7804	38
IRO	1	6.8	1	43.96	0.9487	27.22	0.5611	82
OIR	0.9967	12.11	0.9803	47.89	0.7331	28.06	1	20
ORI	1	11.38	1	14.51	0.9869	14.25	0.7804	38
RIO	1	9.08	0.9824	23.81	0.9722	20.77	0.5611	60
ROI	1	7.66	1	22.51	1	7.31	1	20
میانگین	**0.9998**	**12.15**	**0.9715**	**31.62**	**0.9466**	**23.56**	**0.7010**	**46.31**

[DOI: 10.29252/jsdp.17.4.33]



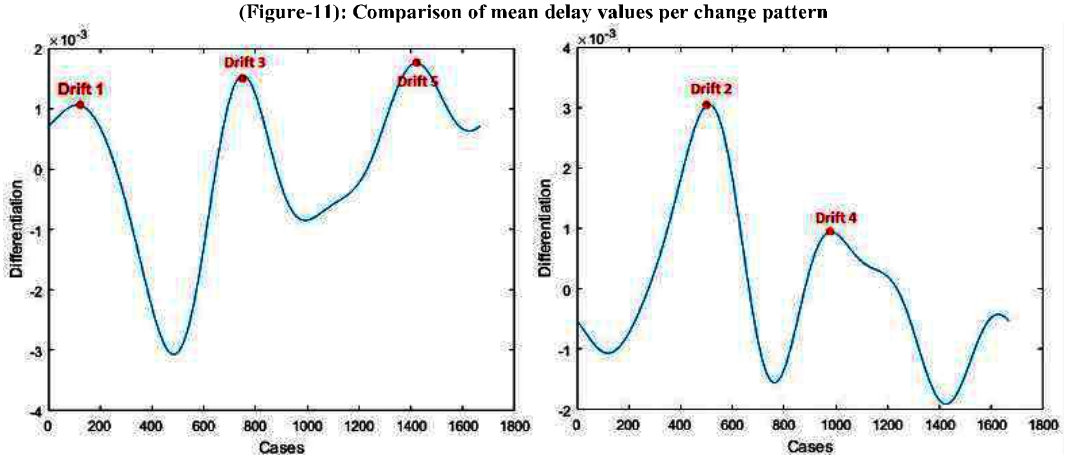
(شکل -۱۰): امتیاز F- به‌دست‌آمده از روش پیشنهادی در مقایسه با run، process-graph و Bose

(Figure-10): Comparison of F-score values per change pattern



(شکل-۱۱): متوسط تأخیر به‌دست‌آمده از روش پیشنهادی در مقایسه با run، process-graph، Bose

(Figure-11): Comparison of mean delay values per change pattern



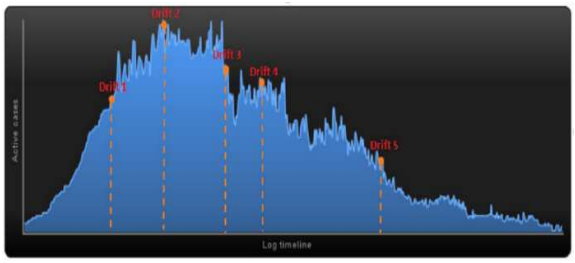
(شکل-۱۲): خروجی روش پیشنهادی بر روی نگاره منطقه ۲ شهرداری

(Figure-12): Plots of the output results after applying the trace2vec approach to M2

[DOI: 10.29252/jsdp.17.4.33]

به‌دلیل محدودیت فضا، تنها نتایج حاصل بر روی یک منطقه شهرداری ارائه شده است. شکل (۱۲) نتایج حاصل از اعمال روش پیشنهادی بر روی M2 را نشان می‌دهد. پنج رانش در اندیس‌های ۱۱۱، ۵۱۰، ۷۵۱، ۹۸۱ و ۱۴۲۱ برای شهرداری منطقه ۲ شناسایی شده، این درحالی است که با اعمال روش مطرح‌شده در [12] هیچ رانشی شناسایی نشد.

نتایج به‌دست‌آمده را با سه تحلیل‌گر کسب‌وکار مورد ارزیابی قرار دادیم. نتیجه ارزیابی‌ها حاکی از عملکرد صحیح روش پیشنهادی بود. تحلیل‌گران کسب‌وکار علت اصلی این رانش‌ها را وجود دو تغییر در فرایند سازمان اعلام کردند: تغییرات مدیریتی و تغییرات نرم‌افزاری. آن‌ها علت اصلی تغییرات نرم‌افزاری را بهبود کارایی فرایند کسب‌وکار بیان کردند که منجر به کاهش تعداد نمونه‌ها در فرایند می‌شود. شکل (۱۳) تاثیر تغییرات شناسایی‌شده را بر روی توزیع تعداد نمونه‌ها در طی زمان نشان می‌دهد. همان‌طورکه در شکل (۱۳) مشاهده می‌شود، اگرچه در برخی بازه‌ها تعداد نمونه‌ها افزایش یافته، اما درنهایت تعداد نمونه‌های متمایز در این منطقه از شهرداری کاهش یافته که این دلیلی بر بهبود کارایی فرایند است.



(شکل-۱۳): موقعیت رانش‌های شناسایی‌شده بر روی نمودار فعالیت‌ها در طی زمان برای نگاره منطقه۲ شهرداری مشهد

(Figure-13): The position of the drifts detected by our approach shown in a plot of active cases vs log timeline for M2

۵- نتیجه‌گیری و کارهای آینده

در این مقاله، سعی شد با ارائه مفهومی جدید دقت روش‌های پیشین در شناسایی رانش فرایند را افزایش دهیم. با بیان ایده تعبیه دنباله و همچنین به‌کارگیری روش‌های خوشه‌بندی و تبدیل فوریه روشی را ارائه کردیم که همه انواع روابط میان دنباله‌ها را به‌صورت خودکار استخراج و بدون استفاده از پنجره، رانش‌های ناگهانی را شناسایی کند. نتایج ارزیابی بر روی یک مجموعه داده استاندارد حاکی از این است که روش پیشنهادی منجر به افزایش دقت در شناسایی رانش فرایند شده است؛ همچنین، عملکرد روش پیشنهادی بر روی نگاره واقعی نیز نشان داده شده است.

با وجود پیشرفت‌های زیادی که در زمینه شناسایی رانش فرایند انجام شده است، اما هنوز موضوعات باز برای پژوهش در این راستا وجود دارد. برخی از آن‌ها عبارت‌اند از:

- **ارائه روشی به‌منظور شناسایی رانش‌ها در دیدگاه داده و منبع**: هیچ یک از روش‌های موجود توانایی شناسایی تغییرات در دیدگاه منبع و با دقت مناسب در دیدگاه داده را ندارند؛ ازاین‌رو، ارائه روشی با کارایی مناسب، اطلاعات ارزشمندی در این دو دیدگاه ارائه خواهد کرد.
- **ارائه روشی به‌منظور شناسایی رانش‌های تدریجی با دقت بالاتر**: همان‌طور که پیش‌تر گفته شد، تنها دو روش مطرح‌شده در [7] و [13] توانایی شناسایی رانش‌های تدریجی را دارند. با این حال، نتایج ارائه‌شده توسط آن‌ها دقت مناسب را نشان نمی‌دهد؛ بنابراین، روشی جدید به‌منظور افزایش دقت در شناسایی رانش‌های تدریجی حائز اهمیت است.
- **ارائه روشی به‌منظور شناسایی رانش‌های متناوب و افزایشی**: شناسایی رانش‌های متناوب و افزایشی با دقت مناسب، اطلاعات ارزشمندی درباره عوامل تأثیرگذار بر فرایند سازمان ارائه می‌دهد.

6- References ۶- مراجع

[1] W. Van Der Aalst, A. Adriansyah, A. K. A. De Medeiros, F. Arcieri, T. Baier, T. Blickle, et al., "Process mining manifesto", *in International Conference on Business Process Management*, vol. 37 No.3, pp. 169-194, 2011.

[2] M. U. Lavanya and M. S. K. Talluri, "Dealing With Concept Drifts In Process Mining Using Event Logs", *International Journal Of Engineering And Computer Science*, vol. 4, pp. 13433-13437, 2015.

[3] R. J. C. Bose, W. M. Van Der Aalst, I. Žliobaitė, and M. Pechenizkiy, "Dealing with concept drifts in process mining", *IEEE transactions on neural networks and learning systems*, vol. 25 No.1, pp. 154-171, 2014.

[4] J. C. Schlimmer and R. H. Granger, "Beyond Incremental Processing: Tracking Concept Drift", *the Association for the Advancement of Artificial Intelligence*, pp. 502-507, 1986.

[5] R. J. C. Bose, W. M. van der Aalst, I. Žliobaitė, and M. Pechenizkiy, "Handling concept drift in process mining", *23rd International Conference on Advanced Information Systems*

[DOI: 10.29252/jsdp.17.4.33]

[16] M. Baroni, G. Dinu, and G. Kruszewski, "Don't count, predict! A systematic comparison of context-counting vs. context-predicting semantic vectors", ACL (1), pp. 238-247, 2014.

[17] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman, "Indexing by latent semantic analysis," *Journal of the American society for information science*, vol.41, p. 391, 1990.

[18] A. Mandelbaum and A. Shalev, "Word Embeddings and Their Use In Sentence Classification Tasks," arXiv preprint arXiv:1610.08229, 2016.

[19] T. Mikolov, W.-t. Yih, and G. Zweig, "Linguistic regularities in continuous space word representations", HLT-NAACL, Atalanta, USA, pp.746-751, 2013.

[20] Q. V. Le and T. Mikolov, "Distributed Representations of Sentences and Documents", ICML, Beijing, China, pp. 1188-1196, 2014.

[21] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space", arXiv preprint arXiv:1301.3781, 2013.

[22] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality", *the 27th Conference on Neural Information Processing Systems(NIPS)*, Nevada, United States, pp. 3111-3119, 2013.

[23] X. Rong, "word2vec parameter learning explained", arXiv preprint arXiv:1411.273, 2014.

[24] P. Ristoski and H. Paulheim, "Rdf2vec: Rdf graph embeddings for data mining", *15th International Semantic Web Conference(ISWC)*, Kobe, Japan, pp. 498-514, 2016.

[25] M. Rahman, "Applications of Fourier transforms to generalized functions", WIT Press, 2011.

[26] S.-S. Ho, "A martingale framework for concept change detection in time-varying data streams", *the 22nd international conference on Machine learning, Germany, ACM*, pp. 321-327, 2005.

فاطمه خجسته دانش‌آموختهٔ کارشناسی ارشد مهندسی رایانه نرم‌افزار از دانشگاه فردوسی مشهد و حوزه پژوهشی ایشان فرآیندکاوی و داده‌کاوی است.

نشانی رایانامه ایشان عبارت است از:

fatemeh.khojasteh@mail.um.ac.ir

Engineering(CAiSE), London, UK, Springer, pp. 391-405, 2011.

[6] H. Schonenberg, R. Mans, N. Russell, N. Mulyar, and W. van der Aalst, "Process flexibility: *A survey of contemporary approaches", 4th International Workshop on Advances in Enterprise Engineering I*, Springer, pp. 16-30, 2008.

[7] J. Martjushev, R. J. C. Bose, and W. M. van der Aalst, "Change Point Detection and Dealing with Gradual and Multi-order Dynamics in Process Mining", *14th International Conference on Business Informatics Research*, Tartu, Estonia, Springer, pp. 161-178, 2015.

[8] R. Accorsi and T. Stocker, "Discovering workflow changes with time-based trace clustering" , *International Symposium on Data-Driven Process Discovery and Analysis*, Italy, Springer, pp. 154-168, 2011.

[9] B. Hompes, J. Buijs, W. van der Aalst, P. Dixit, and J. Buurman, 2015, "Detecting Change in Processes Using Comparative Trace Clustering", *the 5th International Symposium on Data-driven Process Discovery and Analysis (SIMPDA)*, Vienna, Austria , pp. 95-108.

[10] J. Carmona and R. Gavalda, "Online techniques for dealing with concept drift in process mining", *11th International Symposium on Intelligent Data Analysis*, Finland, Springer, pp.90-102, 2012.

[11] P. Weber, B. Bordbar, and P. Tiňo, "Real-Time Detection of Process Change using Process Mining", ICCSW, United Kingdom, 2011.

[12] A. Maaradji, M. Dumas, M. La Rosa, and A. Ostovar, "Fast and accurate business process drift detection", *the 13th International Conference on Business Process Management*, Austria, Springer, pp. 406-422, 2015.

[13] A. Maaradji, M. Dumas, M. L. Rosa, and A. Ostovar, "Detecting Sudden and Gradual Drifts in Business Processes from Execution Traces", *IEEE Trans. Knowl. Data Eng.*, vol. 29, pp. 2140-2154, 2017.

[14] T. Li, T. He, Z. Wang, Y. Zhang, and D. Chu, "Unraveling Process Evolution by Handling Concept Drifts in Process Mining", *in Services Computing (SCC), IEEE International Conference on*, pp. 442-449, 2017.

[15] A. Seeliger, T. Nolle, and M. Mühlhäuser, "Detecting Concept Drift in Processes using Graph Metrics on Process Graphs", *Proceedings of the 9th Conference on Subject-oriented Business Process Management*, 2017.

[DOI: 10.29252/jsdp.17.4.33]

محسن کاهانی استادتمام گروه رایانه و سرپرست آزمایشگاه فناوری وب در دانشگاه فردوسی مشهد است. حوزه‌های فعالیت ایشان وب‌معنایی، پردازش زبان طبیعی، فرایندکاوی، پردازش متن، تحلیل نظرات، داده‌های حجیم است.

نشانی رایانامه ایشان عبارت است از:

kahani@um.ac.ir

بهشید بهکمال استادیار دانشگاه فردوسی مشهد و حوزه‌های فعالیت ایشان داده‌کاوی، فرایندکاوی و داده‌های پیوندی است.

نشانی رایانامه ایشان عبارت است از:

behkamal@um.ac.ir

[DOI: 10.29252/jsdp.17.4.33]