



دانشکده مهندسی برق و کامپیوتر
بخش مهندسی و علوم کامپیوتر و فناوری اطلاعات

رساله دکتری
در مهندسی کامپیوتر
(سیستمهای نرم‌افزاری)

کاوش الگوهای تکراری در جریان های داده بر اساس مدل پنجره لغزنده

به کوشش:
محمود دی پیر

استاد راهنما:
دکتر محمد هادی صدرالدینی

آذر ۱۳۹۰

بِسْمِ اللّٰهِ الرَّحْمٰنِ الرَّحِیْمِ

به نام خدا


اظہارنامہ

اینجانب محمود دی پیر (۸۵۳۹۳۴) دانشجوی رشته‌ی مهندسی کامپیوتر گرایش سیستمهای نرم‌افزاری دانشکده‌ی مهندسی برق و کامپیوتر اظہار می‌کنم که این پایان‌نامه حاصل پژوهش خودم بوده و در جاهایی که از منابع دیگران استفاده کرده‌ام، نشانی دقیق و مشخصات کامل آن را نوشته‌ام. همچنین اظہار می‌کنم که تحقیق و موضوع پایان‌نامه‌ام تکراری نیست و تعهد می‌نمایم که بدون مجوز دانشگاه دستاوردهای آن را منتشر ننموده و یا در اختیار غیر قرار ندهم. کلیه حقوق این اثر مطابق با آیین‌نامه مالکیت فکری و معنوی متعلق به دانشگاه شیراز است.

نام و نام خانوادگی: محمود دی پیر

تاریخ و امضا:

۹۶/۱۱/۳۰



به نام خدا

کاوش الگوهای تکراری در جریان های داده بر اساس مدل پنجره لغزنده

به کوشش:

محمود دی پیر

پایان نامه

ارائه شده به تحصیلات تکمیلی دانشگاه شیراز به عنوان بخشی
از فعالیتهای تحصیلی لازم برای اخذ درجه دکتری

در رشته‌ی:

مهندسی کامپیوتر (سیستمهای نرم‌افزاری)

از دانشگاه شیراز

شیراز

جمهوری اسلامی ایران

ارزیابی و تصویب شده توسط کمیته پایان نامه با درجه: عالی

دکتر محمدهادی صدرالدینی، دانشیار بخش مهندسی و علوم کامپیوتر و فناوری اطلاعات (رئیس کمیته)

دکتر غلامحسین دستغیبی فرد، استادیار بخش مهندسی و علوم کامپیوتر و فناوری اطلاعات

دکتر ستار هاشمی، استادیار بخش مهندسی و علوم کامپیوتر و فناوری اطلاعات

دکتر طاهر نیکنام، دانشیار بخش مهندسی برق دانشگاه صنعتی شیراز

دکتر شاپور گلپهار حقیقی، استادیار بخش مهندسی برق دانشگاه شیراز

آذر ۱۳۹۰

سپاسگزاری:

اکنون که این پایان نامه به پایان رسیده است از زحمات بی دریغ استاد راهنما جناب آقای دکتر صدرالدینی تشکر و قدردانی می نمایم. انجام این پایان نامه بدون راهنمایی ایشان امکانپذیر نبود. همچنین از راهنمایی های اساتید مشاور جناب آقای دکتر دستغیبی فرد و جناب آقای دکتر هاشمی سپاسگزارم. از همسر و فرزندم که در این مدت مرا همراهی نموده و کوتاهی های مرا تحمل کردند، نیز تشکر و قدردانی می نمایم.

تقدیم به

همسر مهربانم

چکیده

کاوش الگوهای تکراری در جریان های داده بر اساس مدل پنجره لغزنده

به کوشش

محمود دی پیر

در مسئله کاوش الگوهای تکراری به دنبال مجموعه هایی هستیم که در تعداد قابل توجهی از تراکنش ها (رکوردها) دیده می شوند. کاوش الگوهای تکراری در جریان های داده کاربرد زیادی در صنعت، تجارت و علوم مختلف دارد. به دلیل سرعت زیاد، بی پایان بودن، حجم بالا و ماهیت تغییر کننده محتوای جریان های داده، یافتن الگوهای تکراری در این نوع داده ها مسئله ای چالش برانگیز است. مدل پنجره لغزنده یکی از مدل های محبوب و پرکاربرد برای حل این مسئله است، که در آن میزان ثابتی از تراکنش های جدید برای کاوش در نظر گرفته می شوند. کاهش میزان حافظه مصرفی، افزایش سرعت کاوش و تعیین اندازه پنجره مهمترین چالش های این مدل اند. ما در این رساله برای غلبه بر این چالش ها الگوریتم ها و تکنیک هایی ارائه داده ایم. نخست دو الگوریتم پیشنهاد داده ایم که در آنها محتویات پنجره به صورت پویایی نگهداری می شود و در صورت درخواست کاربر، مجموعه ارقام تکراری کاوش می شود. سپس الگوریتمی تقریبی ارائه داده ایم که همواره مجموعه ارقام تکراری را نگهداری و به روز رسانی می کند. علاوه بر این الگوریتم جدیدی معرفی کرده ایم که قادر است با استفاده از پنجره های زمانی، سرعت متغییر جریان داده ورودی را در نظر بگیرد. آزمایش های ما نشان می دهند که همگی این الگوریتم ها نسبت به نمونه های مشابه خود در بیشتر موارد، از نظر حافظه و سرعت بهتر عمل می کنند. ما همچنین الگوریتمی ارائه داده ایم که در آن اندازه پنجره لغزنده بر اساس میزان تغییر مفهوم در جریان داده ورودی تنظیم می شود. در نهایت معیار جدیدی را برای تشخیص بهتر میزان تغییر مفهوم ارائه داده ایم.

فهرست

| | |
|--|-----------|
| ۱- مقدمه..... | ۲ |
| ۱-۱ مقدمه..... | ۲ |
| ۲-۱ تعریف مسئله..... | ۴ |
| ۳-۱ مشکلات مطرح در کاوش جریان های داده..... | ۶ |
| ۴-۱ اهداف این پایان نامه..... | ۸ |
| ۵-۱ مراحل انجام تحقیق..... | ۱۰ |
| ۶-۱ ساختار پایان نامه..... | ۱۱ |
| ۲ بررسی روش های گذشته..... | ۱۴ |
| ۱-۲ مدل های پردازش و کاوش جریان های داده..... | ۱۴ |
| ۲-۲ جریان کاوی مجموعه اقلام تکراری در مدل نشانه..... | ۱۷ |
| ۱-۲-۲ الگوریتم شمارش پراتلاف..... | ۱۸ |
| ۲-۲-۲ الگوریتم جنگل مجموعه اقلام تکراری پسوندی..... | ۱۹ |
| ۳-۲-۲ الگوریتمی براساس دامنه چرنوف..... | ۲۳ |
| ۳-۲ کاوش مجموعه اقلام تکراری در مدل زوال..... | ۲۷ |
| ۱-۳-۲ محو زمانی به وسیله وزن دادن مستقیم..... | ۲۷ |
| ۲-۳-۲ الگوریتم استدک..... | ۳۱ |
| ۳-۳-۲ الگوریتم استماکس..... | ۳۷ |
| ۴-۳-۲ پنجره زمانی شیبدار..... | ۴۱ |

| | | |
|-----|-------|--|
| ۴۴ | | ۴-۲ کاوش مجموعه اقلام تکراری در مدل پنجره ای |
| ۴۵ | | ۱-۴-۲ الگوریتم های بر اساس درخواست |
| ۵۶ | | ۲-۴-۲ الگوریتم های به روز رسان |
| ۷۸ | | ۵-۲ سایر الگوریتم ها |
| ۸۰ | | ۳ الگوریتم های بر اساس درخواست پیشنهادی |
| ۸۰ | | ۱-۳ الگوریتم اکلات دی. اس |
| ۸۰ | | ۱-۱-۳ مقدمه |
| ۸۲ | | ۲-۱-۳ الگوریتم |
| ۸۹ | | ۴-۱-۳ ارزیابی کارایی اکلات دی. اس |
| ۹۷ | | ۲-۳ الگوریتم ال. دی. اس |
| ۹۷ | | ۱-۲-۳ مقدمه |
| ۹۸ | | ۲-۲-۳ الگوریتم پیشنهادی ال. دی. اس |
| ۱۱۵ | | ۴-۲-۳ ارزیابی کارایی |
| ۱۲۰ | | ۴ الگوریتم های به روز رسان پیشنهادی |
| ۱۲۰ | | ۱-۴ الگوریتم پیوین |
| ۱۲۰ | | ۱-۱-۴ مقدمه |
| ۱۲۱ | | ۲-۱-۴ الگوریتم پیشنهادی پیوین |
| ۱۳۱ | | ۳-۱-۴ ارزیابی کارایی الگوریتم پیوین |

۲-۴ الگوریتم ان. تی. اس ۱۳۶

۱-۲-۴ پنجره های زمانی ۱۳۷

۲-۲-۴ الگوریتم ان. تی. اس ۱۳۹

۳-۲-۴ ارزیابی کارایی ۱۴۵

۵ تغییر مفهوم ۱۵۰

۱-۵ مقدمه ۱۵۰

۲-۵ تغییر مفهوم در مسئله کاوش مجموعه اقلام تکراری ۱۵۲

۳-۵ کاوش الگوهای تکراری در پنجره لغزان با اندازه متغیر ۱۵۴

۱-۳-۵ مقدار دهی اولیه پنجره و کاوش مجموعه اقلام تکراری ۱۵۴

۲-۳-۵ حذف اطلاعات کهنه از پنجره ۱۵۵

۳-۳-۵ الگوریتم پیشنهادی وی. اس. دبلیو ۱۶۰

۴-۵ ارزیابی ۱۶۳

۵-۵ معیار پیشنهادی برای دنبال کردن میزان تغییر مفهوم ۱۶۷

۶ جمع بندی و تحقیقات آینده ۱۷۲

۱-۶ جمع بندی و نتیجه گیری ۱۷۲

۲-۶ ادامه تحقیقات ۱۷۳

۷ منابع ۱۷۵

فهرست شکل ها

| صفحه | عنوان |
|------|--|
| ۶ | شکل ۱-۱: جریان داده همراه با پنجره های تعریف شده روی آن |
| ۱۵ | شکل ۱-۲: کاوش مجموعه اقلام تکراری در مدل نشانه |
| ۱۶ | شکل ۲-۲: کاوش مجموعه اقلام تکراری در مدل محو شدن تدریجی (مدل زوال) |
| ۱۶ | شکل ۳-۲: کاوش مجموعه اقلام تکراری در مدل پنجره لغزان |
| ۲۱ | شکل ۴-۲: ساخت IsFI-forest بعد از پردازش دسته اول |
| ۲۲ | شکل ۵-۲: ساخت IsFI-forest بعد از پردازش دسته دوم |
| ۲۵ | شکل ۶-۲: الگوریتم اف. پی. دی. ام برای کاوش قلم های تکراری |
| ۲۶ | شکل ۷-۲: الگوریتم اف. پی. دی. ام برای کاوش مجموعه قلم های تکراری |
| ۲۸ | شکل ۸-۲: مقایسه روش وزن دهی به تراکنش ها در مدل های مختلف |
| ۳۳ | شکل ۹-۲: شبه کد الگوریتم استدک |
| ۳۵ | شکل ۱۰-۲: حد بالای تکرار یک مجموعه قلم جدید e |
| ۴۲ | شکل ۱۱-۲: پنجره زمانی طبیعی |
| ۴۳ | شکل ۱۲-۲: پنجره زمانی لگاریتمی |
| ۴۵ | شکل ۱۳-۲: روش نمایش محتویات پنجره در MFI-TRANSW |
| ۴۶ | شکل ۱۴-۲: کاوش مجموعه اقلام تکراری پنجره دوم |
| ۴۹ | شکل ۱۵-۲: درخت های مربوط به پنجره های اول و دوم |
| ۵۰ | شکل ۱۶-۲: روش ایجاد درخت CPS-Tree |

- شکل ۲-۱۷: حذف پین قدیمی با استفاده از نودهای پایانی ۵۲
- شکل ۲-۱۸: اضافه شدن پین سوم ۵۲
- شکل ۲-۱۹: درخت پیشوندی قبل و بعد از بازسازی ۵۳
- شکل ۲-۲۰: جداسازی و ادغام در فرآیند تنظیم شاخه ۵۴
- شکل ۲-۲۱: مرتب سازی شاخه ۵۵
- شکل ۲-۲۲: شمارش شرطی مجموعه اقلام با استفاده از اف. پی. تری ۵۸
- شکل ۲-۲۳: وضعیت یک مجموعه قلم در دو پنجره ۶۱
- شکل ۲-۲۴: شبه کد الگوریتم استوین ۶۵
- شکل ۲-۲۵: چگونگی رخداد یک مجموعه قلم با تکرار حداکثری در یک پنجره ۶۶
- شکل ۲-۲۶: به روز رسانی مجموعه اقلام ۷۰
- شکل ۲-۲۷: شبه کد الگوریتم پنجره زم ۷۴
- شکل ۳-۱: مراحل اجرای الگوریتم اکلا، ۸۰
- شکل ۳-۲: زیر برنامه حذف پین قدیمی از پنجره ۸۵
- شکل ۳-۳: زیر برنامه اضافه شدن پین جدید به پنجره ۸۶
- شکل ۳-۴: عملیات تنظیم پنجره های W_1 و W_2 ۸۷
- شکل ۳-۵: مقایسه زمان اجرا ۹۱
- شکل ۳-۶: مقایسه میزان حافظه مصرفی ۹۴
- شکل ۳-۷: نمایش ماتریس بیتی ۹۶
- شکل ۳-۸: لیست های بیتی ۹۷
- شکل ۳-۹: الگوریتم حذف قدیمی ترین پین از پنجره ۱۰۶
- شکل ۳-۱۰: الگوریتم اضافه کردن پین ۱۰۹
- شکل ۳-۱۱: الگوریتم کاوش مجموعه اقلام در پنجره ۱۱۱
- شکل ۳-۱۲: یک جریان داده نمونه و محتوای دو پنجره اخیر آن با استفاده از روش ال. دی. اس ۱۱۳
- شکل ۳-۱۳: مقایسه زمان اجرا ۱۱۴
- شکل ۳-۱۴: مقایسه حافظه مصرفی ۱۱۵
- شکل ۴-۱: به روز رسانی درخت مجموعه اقلام با پین جدید ۱۲۳
- شکل ۴-۲: ترتیب نودها در پیمایش اول عمق معکوس ۱۲۶
- شکل ۴-۳: الگوریتم اضافه شدن پین جدید ۱۲۸
- شکل ۴-۴: مقایسه زمان اجرا برای الگوریتم های استوین و پیوین با توجه به مقادیر مختلف حد آستانه پشتیبانی (ضریب حد آستانه اهمیت = ۵, ۰) ۱۳۲
- شکل ۴-۵: مقایسه دو الگوریتم استوین و پیوین با توجه به مقادیر مختلف حد آستانه

- ۱۳۳ اهمیت (حد آستانه پشتیبانی = ۰.۵)
- شکل ۴-۶: مقایسه حافظه مصرفی در الگوریتم های استوین و پیوین (حد آستانه
 ۱۳۴ اهمیت=۰.۵)
- شکل ۴-۷: پنجره زمانی در یک جریان داده نمونه
 ۱۳۷
- شکل ۴-۸: کاوش و به روز رسانی همزمان مجموعه اقلام تکراری
 ۱۳۹
- شکل ۴-۹: شبه کد الگوریتم اضافه کردن بلوک جدید (کاوش و به روز رسانی)
 ۱۴۰
- شکل ۴-۱۰: مقایسه میزان حافظه مصرفی
 ۱۴۵
- شکل ۴-۱۱: مقایسه زمان اجرا
 ۱۴۵
- شکل ۵-۱: تغییر مفهوم در مسئله کاوش مجموعه اقلام تکراری در جریان های داده
 ۱۵۲
- شکل ۵-۲: حذف تراکنش های قبل از نقطه بررسی در صورت رخداد تغییر مفهوم
 ۱۵۷
- شکل ۵-۳: الگوریتم وی. اس. دبلیو
 ۱۶۰
- شکل ۵-۴: مقادیر نسبت تغییر و اندازه پنجره (اعداد کنار نقاط) بر اساس اندازه های
 متفاوت پنجره اولیه و پین با استفاده از حد آستانه تغییر ۰.۱ و مجموعه داده
 ۱۶۳ T40I10D200K-AB
- شکل ۵-۵: تأثیر مقدار حد آستانه تغییر (CT) بر روی تشخیص تغییر مفهوم و اندازه
 ۱۶۴ پنجره (اعداد نوشته شده کنار نقاط)
- شکل ۵-۶: الگوی تغییر اندازه پنجره برای الگوریتم VSW در مقایسه با MFW برای
 ۱۶۵ مجموعه داده T40I10D200K-AB
- شکل ۵-۷: مقایسه الگوی تغییرات در دو مجموعه داده واقعی با استفاده از معیارهای
 ۱۶۸ RCR و Fchange (حد آستانه پشتیبانی = ۰.۱)
- شکل ۵-۸: مقایسه الگوی تغییرات در مجموعه داده مصنوعی T40I10D200K-AB
 ۱۶۹ (حد آستانه پشتیبانی = ۰.۲)

فهرست جداول

| صفحه | عنوان |
|------|---|
| ۴۸ | جدول ۱-۲: یک جریان داده تراکنشی |
| ۸۵ | جدول ۱-۳: لیست نمادهای استفاده شده در الگوریتم ها |
| ۸۹ | جدول ۲-۳: ویژگی های مجموعه داده های مورد استفاده در آزمایش ها |
| ۹۲ | جدول ۳-۳: توزیع زمان اجرا در بخش های مختلف الگوریتم ها |
| ۹۵ | جدول ۴-۳: متوسط تعداد تبدیل لیست ها برای هر اندازه پنجره برای مجموعه داده Connect-4 |
| ۱۰۵ | جدول ۵-۳: همه نماد های استفاده شده در الگوریتم ال. دی. اس |
| ۱۱۶ | جدول ۶-۳: نحوه استفاده از لیست های مختلف و انجام تبدیلات در الگوریتم ال. دی. اس |
| ۱۲۱ | جدول ۱-۴: اطلاعات موجود در هر نود در درخت پایش الگوریتم پیوین |
| ۱۳۵ | جدول ۲-۴: خطای مثبت و منفی الگوریتم های استوین و پیوین |
| ۱۴۶ | جدول ۳-۴: مقایسه خطای مثبت و خطای منفی الگوریتم های تی. اس و ان. تی. اس |

فصل اول

مقدمه

۱-۱ مقدمه

مسئله کاوش الگوهای تکراری^۱ در پایگاه های داده در سال ۱۹۹۳ توسط آگروال و دیگران ارائه شد [۱]. این مسئله به علت کاربرد وسیع در تجارت، صنعت و علوم مختلف، موضوعی مهم و اساسی در زمینه کشف دانش و داده کاوی است. برای حل این مسئله الگوریتم پایه ای و شناخته شده ای به نام *ای.پریوری (Apriori)* وجود دارد [۲]. این الگوریتم، معروف ترین الگوریتم در این زمینه است که در نرم افزارهای داده کاوی تجاری نیز استفاده شده است. در سال های اخیر تحقیقات بسیار زیادی در مورد این مسئله انجام گرفته و پیشرفت های چشمگیری صورت گرفته است که حاصل آنها ارائه الگوریتم های کارا و مقیاس پذیر از یک طرف و مطرح شدن آن در کاربرد های مختلف بوده است. تحقیقات انجام شده عموماً در مورد کاوش الگوهای تکراری در پایگاه داده های ایستا بوده است به طوری که الگوریتم های مختلفی در این مورد ارائه شده اند [۲،۳،۴،۵،۶،۷]. اگرچه تعداد الگوریتم های ارائه شده بسیار زیاد است اما اکثر آنها غیر از ای.پریوری [۲]، اف.پی.گروت [۳] واکلات [۷] چندان معروف نبوده و کاربرد آنها فراگیر نشده است. در پایگاه داده های ایستا فرض بر این است که داده ها در طول عملیات کاوش تغییر نمی کنند و الگوریتم داده کاوی می تواند چندین بار داده های ورودی مشخص شده ای را بخواند. به عبارت دیگر کل پایگاه داده در زمان کاوش در اختیار است. یک الگو در یک پایگاه داده تکراری است اگر تعداد رخداد آن از یک حد آستانه بیشتر باشد. این حد آستانه توسط کاربر عملیات داده کاوی تعیین می شود. تحلیل سبد خرید معروف ترین و شناخته شده ترین کاربرد الگوهای تکراری است. در این مسئله کالاهایی که توسط مشتریان مختلف یک فروشگاه به صورت تکراری با هم خریداری می شوند، در پایگاه داده آن فروشگاه

^۱ Frequent pattern mining

کاوش می شوند. نتایج این نوع کاوش بسیار جالب است، به نحوی که گاهی باعث تعجب کاربران شده است.

کاوش الگوهای تکراری علاوه بر داده های ایستا، اخیراً در پایگاه داده های پویا و جریان های داده^۲ نیز مطرح شده است [۸]. در داده های جریانی فرض بر این است که داده ها به صورت پشت سرهم و بی وقفه به سیستم می رسند. در یک جریان داده، داده ها با سرعت و به شکلی بی پایان دریافت می شوند. برنامه های کاربردی و سیستم های زیادی وجود دارند که جریان داده تولید می کنند. از جمله آنها می توان برنامه های پایش^۳ کارایی شبکه، رکورد های جزئیات تماس در مخابرات، کالاهای فروخته شده در فروشگاه های زنجیره ای، سیستم اطلاعات هوا شناسی، پیام های پست الکترونیک، رکورد های ثبت در وب سرورها، سیستم دنبال کردن جریان کلیک در وب سایت ها و غیره را نام برد. انجام محاسبات مختلف آماری و داده کاوی، از جمله مسائل مطرح در مدل جریانی می باشند. در جریان های داده، در هر لحظه، حجم داده هایی که تاکنون رسیده و در آینده نیز خواهد رسید، بسیار زیاد است. بنابراین ذخیره آن امکانپذیر نیست. تنها یک خلاصه کوچک از آن را می توان محاسبه و نگهداری کرد و از مابقی داده ها باید صرف نظر کرد، و یا اینکه فقط داده های اخیراً رسیده و نتایج به روز شده را می توان در حافظه نگهداشت. حتی با فرض اینکه بتوان همه جریان داده را ذخیره کرد، بررسی و خواندن مجدد آن برای پردازش بیشتر امکانپذیر نیست. سرعت رسیدن داده ها به شکلی است که هر عنصر داده ای باید در همان زمان به صورت بلادرنگ پردازش و سپس از آن صرف نظر شود. بنابراین زمان و حافظه دو فاکتور مهم در کاوش جریان های داده هستند. فاکتور مهم دیگر چگونگی برخورد با مسئله تغییر مفهوم است. خصوصیات آماری مفهومی که می خواهیم پیش بینی و یا توصیف کنیم با گذشت زمان در یک جریان داده تغییر می کند به طوری که مدلی که با استفاده از داده های قبلی ساخته شده است، برای داده های جدید معتبر نبوده و یا با خطا همراه خواهد بود. در کاوش الگوهای تکراری مدل مورد نظر مجموعه اقلام تکراری هستند. تغییر مفهوم به صورت تبدیل الگوهای تکراری به غیر تکراری و برعکس با گذشت زمان نمود پیدا می کند. این تغییرات در طول زمان ممکن است چندین بار

^۲ Data streams
^۳ Monitor

اتفاق بیفتد که خود مسئله جریان کاوی الگوهای تکراری را مشکل تر می کند. به عبارت دیگر مجموعه ی کل الگوهای تکراری و مقدار تکرار آنها با گذشت زمان تغییر می کند.

تا کنون سه مدل برای کاوش الگوهای تکراری در جریان های داده ارائه شده اند. این سه مدل، عبارت از مدل نشانه^۴، مدل پنجره لغزنده^۵ و مدل زوال^۶ (یا مدل محو شدن) هستند. در مدل نشانه، داده های رسیده از یک نقطه عطف زمانی تا کنون مورد نظر هستند. این نقطه عطف زمانی می تواند زمان شروع به کار سیستم باشد. در مدل پنجره لغزنده، مقدار مشخص شده ای از داده های اخیراً دریافت شده به منظور کاوش مدنظر قرار می گیرند. این مقدار مشخص شده می تواند عددی و یا زمانی باشد. مورد اول را پنجره تراکنشی و مورد دوم را پنجره زمانی گویند. به عبارت دیگر پنجره W روی یک جریان داده به معنی این است که $|W|$ تراکنش اخیر و یا همه تراکنش های مربوط به $|W|$ واحد زمانی اخیر مورد توجه هستند، که در اینجا $|W|$ اندازه پنجره است. واضح است که تعداد تراکنش های پنجره در حالت اول ثابت و در حالت دوم مطابق سرعت دریافت داده ها می تواند متغییر باشد. در هر دو حالت، داده های جدید از یک طرف وارد پنجره شده و از طرف دیگر داده های قدیمی خارج می شوند، که به این عملیات لغزش پنجره به جلو گفته می شود. در مدل سوم یعنی مدل زوال از یک تابع زوال به منظور وزن دادن به عناصر داده ای بر اساس ترتیب رسیدن آنها از جریان داده، استفاده می شود به نحوی که داده های اخیر وزن بیشتری نسبت به داده های قدیمی تر می گیرند. در این مدل از داده هایی که وزن شان صفر می شود صرف نظر می شود. ما در این پایان نامه مسئله جریان کاوی الگوهای تکراری را در مدل پنجره لغزنده مورد توجه قرار داده ایم چون در این مدل به داده های اخیر اهمیت بیشتری داده می شود و علاوه براین نیاز به حافظه محدودی دارد. با مطالعه و مقایسه الگوریتم ها و همچنین تحقیقات قبلی انجام شده، سعی در ارائه روش های جدید و بهینه تر در این زمینه داریم.

۲-۱ تعریف مسئله

Landmark^۴
Sliding Window^۵
Decay^۶

فرض کنید $I = \{i_1, i_2, \dots, i_m\}$ مجموعه ای از ارقام باشد. همچنین فرض کنید DS جریانی از تراکنش ها به صورت پشت سرهم باشد، به طوریکه هر تراکنش زیر مجموعه ای از I است. برای هر مجموعه قلم (الگو) X که زیر مجموعه ای از I است، گفته می شود که تراکنش T در DS شامل الگوی X است اگر $X \subseteq T$ باشد. کسری از تراکنش ها در DS که شامل X باشند، به عنوان پشتیبانی X شناخته می شود. پشتیبانی مطلق یا تعداد تکرار X، تعداد تراکنش هایی در DS است که شامل این مجموعه قلم باشند. با داشتن یک حداقل حد آستانه پشتیبانی S، گوئیم مجموعه قلم X، تکراری است اگر حداقل S% از تراکنش های DS شامل X باشند. پنجره لغزنده روی جریان داده DS شامل |W| تراکنش اخیر در جریان داده DS است، که |W| در اینجا طول پنجره است. با گذشت زمان، پنجره با اضافه شدن تراکنش جدید و حذف تراکنش قدیمی، به طرف جلو حرکت می کند. جهت بدست آوردن کارایی بهتر می توان تراکنش ها را به صورت دسته ای اضافه و حذف نمود. یک پنجره روی جریان داده DS به صورت:

$$W_{n-|w|+1} = \{T_{n-|w|+1}, T_{n-|w|+2}, \dots, T_n\} \quad (1-1)$$

تعریف می شود که در آن $n-|w|+1$ و T_i به ترتیب شماره پنجره و i امین تراکنش در جریان داده DS هستند. در حقیقت پنجره شامل تراکنش های اخیراً رسیده جریان داده DS می باشد. یک مجموعه قلم به نام X در این پنجره تکراری است اگر رابطه $Sup(X) \geq |W|.S$ برقرار باشد که در این رابطه $Sup(X)$ به معنی تعداد تکرار X در این پنجره است. بنابراین با داشتن پنجره W و حد آستانه پشتیبانی S تعیین شده بوسیله ی کاربر، مسئله یافتن همه ی الگوهای تکراری در این پنجره با استفاده از حداقل زمان و حافظه است. به عنوان مثال شکل 1-1، یک جریان داده از تراکنش ها را نشان می دهد. ستون سمت چپ شماره تراکنش ها و ستون سمت راست ارقام موجود در هر تراکنش است. دو پنجره W_1 و W_2 بر روی تراکنش هایی که تا کنون رسیده اند، در نظر گرفته شده اند.