ارائه یک مدل ترکیبی جهت افزایش دقت روش استدلال مبتنی بر رویداد در برآورد تلاش توسعه نرم افزار

مژده صباغ نژاد[۱،*] و عمید خطیبی بردسیری[۲]

اطلاعات مقاله

دریافت مقاله: ۱۳۹۵/۱۰/۱۲
پذیرش مقاله: ۱۳۹۶/۰۸/۱۳

واژگان کلیدی:
تخمین تلاش توسعه نرم افزار،
مدل استدلال مبتنی بر رویداد،
الگوریتم کرم شب تاب،
الگوریتم ازدحام ذرات.

چکیده

امروزه تخمین تلاش تو سعه نرم افزار در مدیریت پروژه‌های نرم‌افزاری امری حیاتی ا ست. برآورد دقیق هزینه نه تنها به م شتریان و سرمایه گذاران کمک می‌کند، بلکه در ت صمیم گیری منطقی حین انجام پروژه و مدیریت پروژه نرم‌افزاری نیز تاثیر گذار خواهد بود. تا کنون مدل‌های تخمین بی‌شماری ابداع و مورد استفاده قرار گرفته است. بسیاری از رویکردهای تخمین تلاش فعلی با جمع آوری داده‌ها از پروژه‌های قبلی انجام می‌شود. روش اس تدلال مبتنی بر رو یداد یکی از تکن یک های موفق در زمی نه تخمین تلاش پروژه‌های نرم‌افزاری است. این روش به تنهایی از دقت پایینی برخوردار است که این نقص را می‌توان با ایجاد مدل‌های ترکیبی بر طرف کرد. در این مقاله سعی شده است که با ترکیب مدل استنتاج مبتنی بر رویداد و دو الگوریتم فرا اکتشافی مستقل از جمله الگوریتم ازدحام ذرات و الگوریتم کرم شب تاب مدل ترکیبی جدیدی پی شنهاد و عملکرد مدل پیشنهادی را مورد ارزیابی قرار دهیم. با توجه به نتایج بد ست آمده مدل پی شنهادی بر روی سه مجموعه داده کوکومو، آلبرشت و ماکسول، می‌توان گفت که الگوریتم کرم شب تاب عملکرد قابل قبولی داشته است.

۱- مقدمه

تخمین هزینه‌های یک پروژه نرم‌افزاری یکی از مهمترین فعالیت‌ها در مدیریت پروژه نرم‌افزاری است. دلیل اصلی از برآورد هزینه پروژه‌های نرم‌افزاری این است که برنامه‌ریزی دقیق، نظارت و کنترل پروژه امکان پذیر نیست، متاسفانه در پروژه‌های نرم‌افزاری نمی‌توان به فرآیند تخمین اعتماد کرد هیچ کدام از مدل‌های تخمین نمی‌تواند به طور دقیق هزینه‌های پروژه نرم‌افزاری را برآورد کند. برآورد تلاش‌های انسانی در پروژه‌های نرم‌افزاری به شدت تحت تاثیر امور غیر مرتبط و اطلاعات گمراه کننده قرار می‌گیرند. بعلاوه توسعه دهندگان نرم‌افزار حتی بعد از دریافت باز خورد برآوردهای انجام شده باز میل به بهبود برآوردهای پیشنهادی خود دارند. یکی از راه های برآورد تلاش، استفاده از برآورد کننده‌های خودکار است. در واقع مدل برآورد تلاش را می‌توان به عنوان ابزارهای پشتیبان در تصمیم‌گیری بکار برد که امکان بررسی میزان تاثیر ویژگی-های پروژه و تیم را بر افزایش هزینه فراهم می‌سازد [۲،۱]. در طول سه دهه گذشته، محققان تمایل زیادی به استفاده از نرم‌افزارهای گوناگون برای مدل‌های برآورد تلاش جهت غلبه بر تنوع فرآیندهای توسعه نرم‌افزار داشتند. اگرچه زمان و هزینه بسیار زیادی به بهبود دقت مدل‌های مختلف برآورد اختصاص داده شده است، به دلیل عدم قطعیت در

* پست الکترونیک نویسنده مسئول: mozhdeh.sabagh@yahoo.com
۱. گروه کامپیوتر، واحد کرمان، دانشگاه آزاد اسلامی، کرمان، ایران
۲. گروه کامپیوتر، واحد بردسیر، دانشگاه آزاد اسلامی، بردسیر، ایران

پروژه‌های توسعه نرم‌افزار و عواملی مانند ویژگی‌های پویا، پیچیدگی های ذاتی نرم‌افزار، مشکلات ناشی از عدم استاندارد سازی و کمبود اطلاعات مربوط به نرم‌افزار، نمی-توان انتظار داشت که نرم‌افزارهای برآورد تلاش دقت بسیار بالایی داشته باشند [۴،۳]. مدل‌های برآورد تلاش نرم‌افزار سعی دارند که میزان تلاش لازم برای توسعه نرم‌افزار را به بهترین شکل تخمین بزنند. البته فرآیند برآورد انجام شده مطمئن نیست چرا که به ارزش ویژگی‌های بستگی دارد که در مراحل ابتدایی پروژه نامشخص می‌باشند. اما به هر حال این فرآیند باید انجام شود چرا که سرمایه گذاری وسیعی برای ساخت نرم‌افزار انجام می‌شود [۵،۶]. مطالعات نشان می‌دهد که میزان موفقیت اجرای پروژه‌های نرم افزاری بسیار پایین است تنها ۳۰٪ تا ۳۵٪ از تمام پروژه‌های نرم-افزاری در زمان و بودجه تعیین شده به پایان رسیده‌اند [۷،۶،۲]. یکی از مهمترین دلایل برای عدم موفقیت، برآورد نادرست و فاقد دقت است. مشکلات مربوط به برآوردهای بیش حد و یا پایین بودن میزان تلاش است، که هر دو نوع آن به نوبه خود بر روی روند پروژه نرم‌افزاری تاثیر منفی می‌گذارد. استفاده از انواع مدل‌های برآورد تلاش رو به رشد است و تیم‌های نرم‌افزاری از روش‌های گوناگون برای برآورد تلاش پروژه خود استفاده می‌کنند. مهم آن است که بدون وجود اطلاعات کافی در حوزه و دامنه سیستم و همچنین شرایط محیطی و فرهنگی تیم تولید نرم‌افزار و پیچیدگی تکنیکی، برآورد تلاش واقع بینانه‌ای ارائه گردد [۸،۲،۱]. روش‌های تخمین تلاش موجود خطای نسبی را به عنوان مهمترین هدف خود معرفی می‌کنند و تلاش می‌کنند تا میزان خطا را تا حد امکان کاهش دهند. به دلیل عدم قطعیت و خواص غیر خطی و پیچیده پروژه‌های نرم‌افزاری، تمرکز مدل تخمین بر روی این معیار به تنهایی نمی‌تواند زمینه دستیابی به دقت‌های بالا و مطمئن را فراهم سازد. علاوه بر این، مدل‌های تخمین مبتنی بر بهینه سازهای تک هدفه قادر به مدیریت پروژه‌های نیستند و بطور قابل ملاحظه‌ای نتایج حاصل از این نوع برآوردگرها در یک پایگاه داده نسبت به پایگاه داده دیگر متفاوت است [۱۱-۹]. در نتیجه تعمیم دقت برآوردگر تلاش موجود برای انواع پروژه‌های نرم‌افزاری به دلیل سطح بالای پیچیدگی امکان پذیر نیست[۸،۳]. لذا در بخش دوم این مقاله به بررسی کارهای مرتبط انجام شده پیشین پرداخته خواهد شد، در بخش۳ نگاه جامع بر روش استدلال مبتنی بر رویداد خواهد داشت. بخش ۴ و ۵ توصیفی از الگوریتم‌های فرابتکاری بکار رفته در مدل پیشنهادی از جمله الگوریتم کرم شب تاب و الگوریتم ازدحام ذرات بیان خواهد شد. مدل پیشنهادی و همچین ارزیابی مدل پیشنهادی در بخش های ۶ و ۷ بیان خواهند شد، بخش ۸ شامل نتیجه‌گیری و بحث می‌باشد.

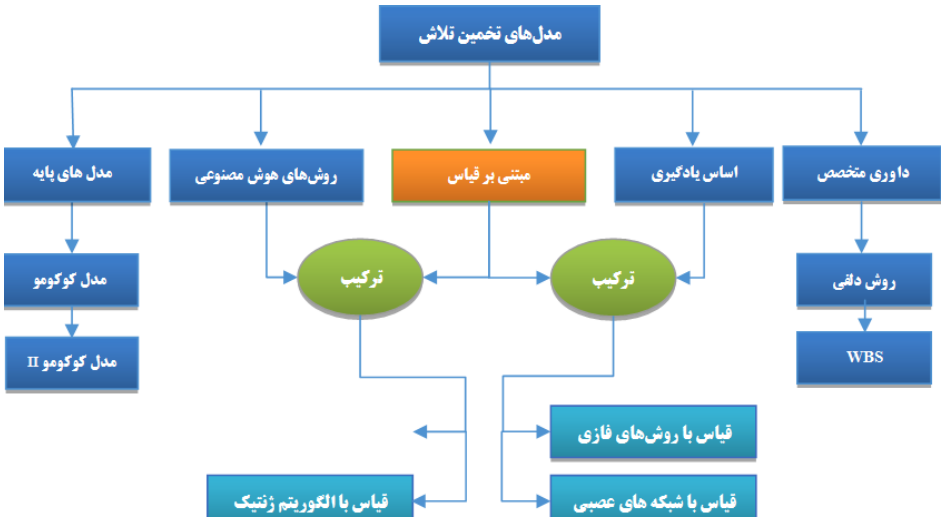
۲- کارهای مرتبط

به طور کلی برآورد هزینه به پیش‌بینی میزان تلاش، میزان نیروی انسانی، زمان و نیازهای دیگر برای به انجام رساندن یک پروژه نرم‌افزاری اطلاق می‌گردد. اغلب مدل‌ها که امروزه استفاده می‌شوند مدل‌های دو مرحله‌ای هستند. در مرحله اول اندازه پروژه تعیین می‌شود و در مرحله دوم فاکتورهای نرخ تولید بدست می‌آیند. با توجه به شکل (۱) می‌توان مدل‌های مختلف نرم‌افزاری را معرفی کرد [۱۲،۸]. توسعه مدل‌های تخمین تلاش نرم‌افزار موثر، منجر به تحقیقات هدفمندی در زمینه‌های مدیریت پروژه‌های نرم‌افزاری گردیده است. عملکرد روش‌های برآورد تلاش توسعه نرم-افزار به محتوای آن و تنوع پروژه‌ها بستگی دارد [۹،۵]. برآوردهای انجام شده بر اساس تجربه‌های واقعی انجام می-شود و عیب آن در این است که پروژه در حال اجرا نمی‌تواند کاملاً منطبق با پروژه های قبلی باشد [۱۳،۵]. مقایسه نا آگاهانه و اشتباه پروژه‌ها می‌تواند روند تخمین تلاش را منحرف سازد [۱۴،۱۱].

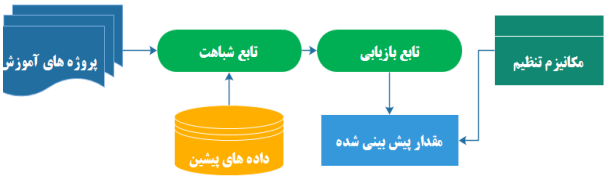
۳- مدل استدلال مبتنی بر رویداد

مدل استدلال مبتنی بر رویداد در سال ۱۹۹۴ توسط واتسون و ماری به عنوان یک تکنیک حل مسئله با رویکرد تطبیق راه حل مشکلات قدیمی بر مسائل کنونی مورد بررسی و استفاده قرار گرفت [۱۵]. مدل CBR به بازیابی یک یا چند مورد مشابه به مسئله فعلی و تلاش برای تغییر پارامترها و برجسته شدن مشکل فعلی می‌باشد. روش مبتنی بر مقایسه یک شکل کامل از استدلال مبتنی بر رویداد بدون کارشناسی می‌باشد[۱۶]. ساختار و مراحل کلی مدل CBR در شکل (۲) نمایش داده شده است [۱۷]. فرآیند برآورد تلاش در این روش با جمع آوری داده‌ها از پروژه‌های قبلی و تولید مجموعه‌ای از اطلاعات اولیه آغاز می‌شود و انتخاب پارامترهای اندازه‌گیری مناسب مانند: تابع امتیاز و خطوط کد منبع که با استفاده از پروژه‌های قبلی جمع‌آوری می‌شوند. بازیابی پروژه‌های قبلی و محاسبه

شباهت بین پروژه هدف و پروژه‌های قبلی رایج‌ترین تابع شباهت فاصله اقلیدوسی و فاصله منتهن است، هر دو تابع به طور گسترده برای اندازه‌گیری درجه شباهت بین پروژه‌های نرم‌افزاری استفاده می‌شوند. هیچ مبنایی برای انتخاب فاصله اقلیدسی یا فاصله منتهن برای مجموعه داده خاص وجود ندارد و این امر به وسیله آزمون و خطا انجام شده است[۱۸]



شکل ۱: دسته بندی مدل های برآورد تلاش نرم افزار



شکل ۲: نمودار استنتاج مبتنی بر رویداد CBR

ماهیت پروژه‌ها در مجموعه داده و سطح نرمال می‌تواند اثر قابل توجهی بر روی عملکرد توابع شباهت داشته باشد. طبق معادله شماره۱ فاصله اقلیدسی محاسبه می‌گردد [۱۹-۲۱]:

$$Sim(p,p') = \frac{1}{\left[\sqrt{\sum_{i=1}^{n} w_i Dis(fi,fi')} + \delta\right]} \qquad (۱)$$
$$\delta = 0.0001$$

$$Dis(fi,fi') = \begin{cases} (fi-fi')^2 & if\ fi\ and\ fi'\ are\ numerical\ or\ ordinal \\ 0 & if\ fi\ and\ fi'\ are\ nominal\ and\ fi=fi' \\ 1 & if\ fi\ and\ fi'\ are\ nominal\ and\ fi\neq fi' \end{cases} \qquad (۲)$$

که در آن P' و P نشان‌دهنده پروژه‌ها، Wi وزن تعیین شده برای هر ویژگی، fi' و fi نشان‌دهنده ویژگی هر پروژه و n تعداد ویژگی می‌باشند. از δ نیز برای بدست آوردن نتایج غیر صفر استفاده می‌شود. فرمول فاصله منهتن بسیار مشابه فرمول فاصله اقلیدسی است و برای محاسبه اختلاف مطلق ویژگی استفاده می‌شود. طبق معادله شماره۲ فاصله منهتن محاسبه می‌گردد[۲۲]:

$$Sim(p,p') = \frac{1}{\left[\sqrt{\sum_{i=1}^{n} w_i Dis(fi,fi')} + \delta\right]} \qquad (۳)$$
$$\delta = 0.0001$$

$$Dis(fi,fi') = \begin{cases} |fi-fi'| & if\ fi\ and\ fi'\ are\ numerical\ or\ ordinal \\ 0 & if\ fi\ and\ fi'\ are\ nominal\ and\ fi=fi' \\ 1 & if\ fi\ and\ fi'\ are\ nominal\ and\ fi\neq fi' \end{cases} \qquad (۴)$$

به علاوه توابع شباهت دیگری نیز وجود دارد همانند شباهت رتبه میانگین و شباهت حداکثر فاصله و شباهت منیکوفسکی این سه تابع شباهت به طور گسترده مورد استفاده قرار می‌گیرند [۲۳،۲۴].

۴- الگوریتم کرم شب تاب

الگوریتم کرم شب تاب یا به اختصار FA در اواخر سال ۲۰۰۷ و توسط هین شی یانگ معرفی شده است که ایده اصلی آن از ارتباط نوری میان کرم‌های شب تاب الهام گرفته شده است. این الگوریتم را می‌توان از مظاهر هوش ازدحامی دانست، یکی از الگوریتم‌های موفق و در عین حال کم هزینه

به شمار می‌آید [۲۵]. با توجه قوانین تابش نور شدت نورI را در فاصله r منبع نور از قانون مربع معکوس تبعیت می-کند، شدت نور در فاصله r کاهش می‌یابد I∝1/r^2 ، علاوه بر این جریان هوا باعث جذب نور می‌شود و رفته رفته از شدت نور کاسته می‌شود. میزان روشنایی می‌تواند به عنوان تابع هدف ساده در نظر گرفته شود. یا می‌توان روشنایی حشره را به عنوان تابع برازش مشابه الگوریتم ژنتیک تعریف کرد. الگوریتم کرم شب تاب FA را می‌توان به صورت شبه کد در شکل ۳به صورت خلاصه نمایش داد [۲۷،۲۶].

```
Firefly Algorithm
Objective function f (x),    x = (x1, ... , xd)T
Generate initial population of fireflies xi (i = 1, 2, ... , rt)
Light intensity It at xi is determined by f (xi)
Define light absorption coefficient y
  while (t < MaxGeneration )
    for i = 1 :n all n fireflies
      for j = 1 : n all n fireflies (inner loop)
        if (Ii < Ij), Move firefly i towards j; end if
        Vary attractiveness with distance r via exp[ - Γ r ]
        Evaluate new solutions and update light intensity
      end for j
    end for i
    Rank the fireflies and find the current global best g*
  end while
Postprocess results and visualization
```

شکل ۳: شبه کد الگوریتم کرم شب تاب FA

برای مسأله ماکزیمم‌سازی بهینه سازی، روشنایی I یک کرم شب تاب در یک منطقه خاص x. می‌تواند به صورت $I(x)$ $\alpha f(x)$. به هرحال جذابیت β نسبی ست، چرا که باید در چشمان بیننده توسط سایر کرم‌های شب تاب دیده شود. بنابراین با فاصله r_{ij} کرم شب تاب i و کرم شب تابjتغییر خواهد کرد. به علاوه، کاهش شدت روشنایی با فاصله از منبع آن متناسب است. همچنین نور در رسانه‌ها جذب می‌شود، بنابراین باید در نظر گرفت که جذابیت با میزان جذب تغییر می‌کند [۱۵،۲۸].

$$I(r) = \frac{I_S}{r^2} \qquad (۳)$$

که I_s شدت نور منبع است. برای یک محیط با یک ضریب جذب نور[1] ثابت γ، شدت نور I با فاصله r تغییر می‌کند:

$$I = I_0 e^{-\gamma r} \qquad (۴)$$

که I_0 شدت نور اصلی می‌باشد. به منظور اجتناب از یکتایی $r = 0$ در عبارت $\frac{I_S}{r^2}$ ، ترکیب اثر هر دو قانون مربع معکوس و جذب می‌توان فرمول را به فرم گوسین زیر تقریب زد:

$$I(r) = I_0 e^{-\gamma r^2} \qquad (۵)$$

از آنجایی که جذابیت یک کرم شب تاب با شدت نور دیده شده توسط کرم‌های مجاور آن متناسب است، می‌توانیم جذابیت β را به صورت زیر تعریف نماییم:

$$\beta = \beta_0 e^{-\gamma r^2} \qquad (۶)$$

که β_0 جذابیت[2] در $r = 0$ است. در پیاده‌سازی واقعی، تابع جذابیت $\beta(r)$ می‌تواند هر تابع یکنواخت کاهشی مانند شکل کلی زیر باشد:

$$\beta(r) = \beta_0 e^{-\gamma r^m} \quad , \quad (m \geq 1) \qquad (۷)$$

ویژگی فاصله $\Gamma = 1/\sqrt{\gamma}$ را تعریف می‌کنند که جذابیت به طور قابل توجهی از β_0 به $\beta_0 e^{-1}$ تغییر می‌کند. برای γ ثابت، ویژگی طول می‌شود:

$$\Gamma = \gamma^{-1/m} \to 1 \quad , \quad m \to \infty \qquad (۸)$$

به طور برعکس، برای مقیاس طول Γ در یک مسأله بهینه سازی، پارامتر γ می تواند به عنوان یک ارزش اولیه معمول استفاده شود:

$$\gamma = \frac{1}{\Gamma^m} \qquad (۹)$$

فاصله بین هر دو کرم شب تاب i و jدر x_i و x_j ، به ترتیب، فاصله دکارتی است:

$$r_{ij} = \|x_i - x_j\| = \sqrt{\sum_{k=1}^{d}(x_{i,k} - x_{j,k})^2} \qquad (۱۰)$$

که k، $x_{i,k}$ امین مؤلفه از فاصله مختصاتی x_i کرم شب تاب i ام می‌باشد. در ۲ بعد خواهیم داشت:

$$r_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \qquad (۱۱)$$

جابجایی یک کرم شب تاب i به سمت کرم شب تاب جذاب تر (روشن تر)jبه صورت زیر تعریف می‌شود:

$$x_i = x_i + \beta_0 e^{-\gamma r_{ij}^2}(x_j - x_i) + a \qquad (۱۲)$$

که در آن واژه دوم مربوط به جذابیت، واژه سوم پارامتر رندم ساز[3] α و ϵ_i بردار تصادفی اعداد نشان داده شده از یک توزیع گوسین و یا توزیع یکنواخت می‌باشند. الگوریتم کرم شب تاب با مقدار دهی اولیه جمعیت آغاز می‌شود، با توجه به خصوصیات و قابلیت‌های داخلی هر کرم شب تاب

[1] Light Absorption Coefficient
[2] Attraction Coefficient Base Value
[3] Mutation Coefficient

روشنایی و حرکات آن تعیین می‌گردد[۱۷و۳۱-۳۳]. شکل (۴) مراحل اجرای الگوریتم کرم شب تاب را نشان می‌دهد.

شکل ۴: فلوچارت مراحل کار الگوریتم کرم شب تابFA

۵- الگوریتم ازدحام ذرات[1]

ایده الگوریتم ازدحام ذرات، برای اولین بار توسط کندی و ابرهارت در سال ۱۹۹۵ مطرح شد، یک الگوریتم تکاملی الهام گرفته از طبیعت و بر اساس تکرار می‌باشد. منبع الهام این الگوریتم، رفتار اجتماعی حیوانات، همانند حرکت دسته جمعی پرندگان و ماهی ها بود است. در واقع الگوریتم PSO از تعداد مشخصی از ذرات تشکیل می‌شود که به طور تصادفی، مقدار اولیه می‌گیرند. برای هر ذره دو مقدار وضعیت و سرعت، تعریف می‌شود که به ترتیب با یک بردار مکان و یک بردار سرعت مدل می‌شوند. این ذرات به صورت تکرار شونده در فضای n بعدی مسئله حرکت می‌کنند، یک حافظه به ذخیرهٔ بهترین موقعیت هر ذره در گذشته و یک حافظه به ذخیره بهترین موقعیت پیش آمده در میان همه ذرات، اختصاص می‌یابد [۳۴،۲۵].

$$v_i(t+1) = wv_i(t) + c_1 \times rand \times (pbest_i - x_i(t)) + c_2 \times rana \times (gbest_i - x_i(t)) \quad (۱۳)$$

$$x_i(t+1) = x_i(t) + v_i(t+1) \quad (۱۴)$$

که در آن V_i سرعت ذره، $pbest_i$ بهترین تجربه شخصی خود ذره را نگه می‌دارد، gbest بهترین تجربه کل اجتماع را نگه می‌دارد، xi مکان و موقعیت فعلی ذرات، rand اعداد تصادفی مستقل با توزیع یکنواخت، c_1، c_2 ضرایب شتاب می‌باشند. پیاده سازی این الگوریتم ساده بوده و نیاز به تعیین پارامترهای کمی دارد. شبه کد الگوریتم ازدحام ذرات در شکل (۵) نمایش داده شده است [۳۵].

PSO Algorithm

```
For each particle
    Initialize particle
End For

Do
    For each particle
        Calculate fitness value
        If the fitness value is better than the best fitness value (Pbest) in history Then
            Set current value as the new Pbest
        End If
    End For

    Choose the particle with the best fitness value of all the particles as the Gbest

    For each particle
        Calculate particle velocity according to Equation (4)
        Update particle position according to Equation (5)
    End For
```

شکل ۵: شبه کد الگوریتم ازدحام ذرات PSO

۶- مدل پیشنهادی

با توجه به تنوع پروژه‌های نرم‌افزاری و همچنین ویژگی‌های متنوع سیستم‌های نرم‌افزاری، مدل‌های مبتنی بر مقایسه از جمله مدل CBR با وجود سادگی و انعطاف پذیری بالا نمی‌توانند عملکرد خوبی داشته باشند. تعیین تشابه بین دو پروژه بدون توجه به اهمیت هر یک از ویژگی-های نرم‌افزاری ممکن است بر یکپارچگی روند مقایسه‌ها تاثیر منفی بگذارد. با توجه به ماهیت نامطلوب و پیچیدگی پروژه‌های نرم‌افزاری روند مقایسه نیاز به توجه بیشتری نسبت به سایر پروژه‌ها دارد. تجزیه و تحلیل ویژگی‌های جامع قبل از مقایسه پروژه می‌تواند عملکرد مدل CBR را بهبود بخشد. همانطور که پیش از این ذکر شد، مرحله مقایسه از طریق تابع شباهت در مدل CBR انجام می‌شود. بنابراین، مدل پیشنهادی بر بهبود عملکرد تابع شباهت تاکید دارد. جهت بهبود عملکرد مدل استنتاج مبتنی بر رویداد روش‌های مختلفی برای تعیین ارزش ویژگی‌های موجود در پروژه‌های نرم‌افزاری ارائه شده است. در این مقاله، الگوریتم FA به عنوان ابزاری مناسب برای وزن دهی ویژگی‌های پروژه نرم‌افزاری مورد استفاده قرار گرفت است.

[1]Particle swarm optimization Algorithm

انعطاف پذیری و سازگاری دو ویژگی ارزشمند الگوریتم کرم شب تاب می‌باشند، که این الگوریتم را قادر می‌سازد که بر مشکل پیچیدگی و ابهام ویژگی‌های پروژه نرم‌افزاری غلبه کند. بنابراین الگوریتم FA می تواند در بهبود وزن دهی ویژگی‌ها موثر باشد. مدل پیشنهادی شامل دو مرحله است:

آ. مرحله آموزشی[1]: در این مرحله مدل متناسب با ویژگی‌ها نرم‌افزاری به تخصیص و توزیع وزن مناسب در ویژگی‌ها می‌پردازد.

ب. مرحله تست[2]: در این مرحله مدل پیشنهادی مورد ارزیابی و آزمون قرار می‌گیرد.

مقایسه ویژگی‌های دو پروژه نرم‌افزاری بسیار پیچیده است، روش پیشنهادی می‌تواند عملکرد CBR را بهبود ببخشد. معیارهای عملکردی معرفی شده در این مقاله می‌تواند ما را در ارزیابی بهتر مدل پیشنهادی یاری کند [۳۴-۳۷].

۶-۱-معیارهای عملکرد

دقت تخمین تلاش یکی از عوامل مهم در موفقیت پروژه است. عامل اصلی در انتخاب الگوی تخمین تلاش، دقت پیش‌بینی آن در امر تخمین است. بدین منظور جهت بررسی میزان دقت مدل پیشنهادی از معیارهای خطای نسبی[3]، مقدار خطای نسبی[4]، میانگین مقدار خطای نسبی[5] که در روابط زیر آمده است استفاده می‌شود [۳۸،۱۲،۵].

$$RE = |Estimated - Actual| \quad (۱۵)$$

$$MRE = \frac{|Estimated - Actual|}{Actual} \quad (۱۶)$$

$$MMRE = \sum_{i=1}^{N} \frac{MRE_i}{N} \quad (۱۷)$$

اساس محاسبه مقدار خطای نسبی و میانگین مقدار خطای نسبی استفاده از خطای نسبی می‌باشد. میانگین، با در نظر گرفتن مقدار تلاش واقعی و برآورد آن از مجموعه داده‌های به کار رفته محاسبه می‌شود. وقتی چندین پروژه با خطاهای نسبی بزرگ وجود دارد، ارزیابی صحت برآورد مدل ممکن است منحرف شود[۳۹]. یک جایگزین برای میانگین، میانه می‌باشد که یک سنجش مرکز گرا را نشان می‌دهد به طوریکه کمتر به وجود چندین خطای نسبی بزرگ حساس است. میانه خطای نسبی[6]، درصد پیش بینی[7]

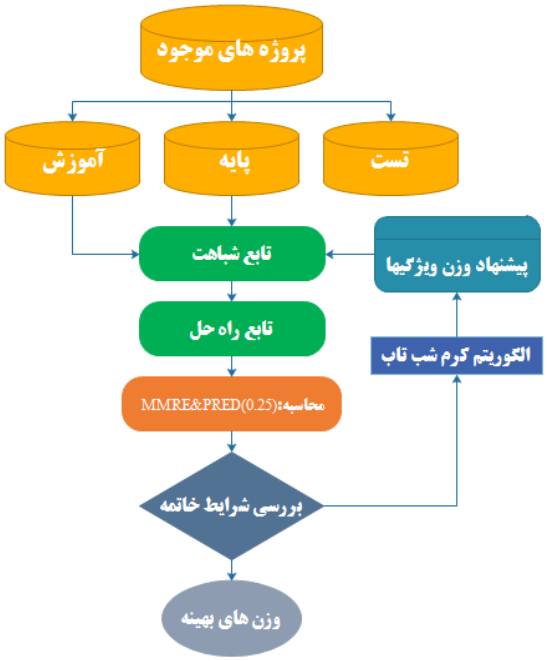
$$MdMRE = Median(MRE) \quad (۱۸)$$

$$PRED(X) = \frac{A}{N} \quad (۱۹)$$

که در آن A تعداد پروژه ها با MRE کمتر یا مساوی X و تعداد پروژه‌های مورد نظر می‌باشد [۴۰،۳۳،۱۰].

۶-۲-گام آموزش در مدل پیشنهادی

در ابتدا پروژه‌های به صورت تصادفی به سه دسته تقریباً مساوی تقسیم می‌شوند، که یک دسته آنها آزمون و دو دسته دیگر مجموعه آموزش را تشکیل می‌دهند. پروژه‌های آموزش برای آموزش مدل پیشنهادی استفاده می‌شوند، در حالی که پروژه‌های آزمون برای ارزیابی کارایی مدل پیشنهادی بکار گرفته می‌شوند. به عبارت دیگر از پروژه‌های آموزش برای بدست آوردن مناسب‌ترین وزن و مناسب ترین پارامتر ها استفاده می‌شود، پروژه‌های تست برای ارزیابی عملکرد گام آموزش است. روند اجرای گام آموزش در شکل (۶) نمایش داده شده است.



شکل ۶: فلوچارت مربوط به گام آموزش مدل پیشنهادی

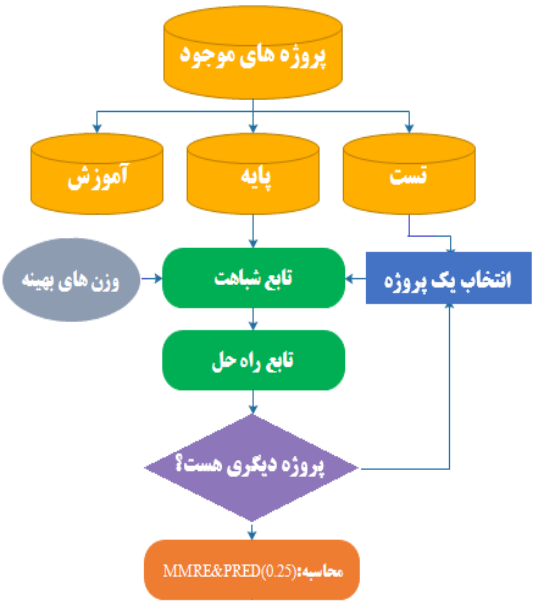
علاوه بر آن در این مرحله تعدادی وزن توسط الگوریتم بهینه‌سازی کرم شب تاب به ویژگی‌های پروژه‌ها اختصاص داده می‌شود. این وزن ها طبق رابطه‌های شماره ۱ و ۲ در

[1] Train stage
[2] Test stage
[3] Relative Error (RE)
[4] Magnitude of Relative Error (MRE)
[5] Mean Magnitude of Related Error (MMRE)
[6] Median Magnitude of Relative Error (MdMRE)
[7] Percentage of Prediction (PRED)

تابع شباهت بکار می‌روند. تعداد وزن‌ها برابر تعداد ویژگی‌ها خواهد بود و به منظور تعیین درجه اهمیت هر ویژگی مورد استفاده قرار می‌گیرد. در گام بعد پروژه (یا پروژه‌های مشابه) به تابع راه حل ارسال می‌گردد[۴۵]. در این مرحله الگوریتم کرم شب تاب سه پیشنهاد به عنوان تابع راه حل ارائه می‌دهد، مقدار خطای نسبی (MRE) محاسبه می‌شود. این فرآیند تا زمانی که تمام پروژه‌های آموزش تخمین زده شوند تکرار می‌گردد. به عبارت دیگر به تعداد پروژه‌های آموزش MRE وجود خواهد داشت. در صورت عدم وجود پروژه دیگر گام بعدی محاسبه PRED و MMRE خواهد بود. بنابراین در روش پیشنهادی اختلاف مقدار MMRE وPRED به عنوان میزان شایستگی در نظر گرفته می‌شود.

۶-۳- گام آزمون در مدل پیشنهادی

ارزیابی دقت مدل پیشنهادی توسط پروژه‌های آزمون انجام می‌شود. در این گام به منظور بررسی عملکرد مدل پیشنهادی پروژه‌های آموزش، تابع شباهت پیشنهادی از مرحله قبل به عنوان ورودی‌های تابع شباهت در نظر گرفته می‌شوند [۴۱]، بعلاوه وزن‌های بهینه شده که از گام آموزش بدست آمده‌اند به این تابع ارسال می‌گردد. همانطور که در شکل۷ مشاهده می‌شود، مشابه آنچه که در گام آموزش صورت گرفته است ابتدا یک پروژه از میان پروژه-های آزمون انتخاب می‌شود و به تابع شباهت ارسال می-گردد.



شکل ۷: فلوچارت مربوط به گام آزمون مدل پیشنهادی

در ادامه مقدار تلاش لازم برای پروژه تحت بررسی تخمین زده می‌شود و در گام بعدی مقدار خطای نسبی محاسبه می‌گردد. این فرآیند تا زمانی که پروژه‌های آزمون به اتمام نرسیده‌اند تکرار می‌شود و در نهایت میزان MMRE وPRED محاسبه می‌گردد، که این پارامترها نشان دهنده دقت مدل پیشنهادی هستند. شکل ۷مراحل اجرای گام آزمون را نشان می‌دهد.

۷- ارزیابی

برای ارزیابی عملکرد مدل پیشنهادی از سه مجموعه داده واقعی استفاده شده است. در جدول ۱ مشخصات کلی هر مجموعه داده به اختصار بیان شده است.

جدول ۱: معرفی مجموعه داده‌های مورد استفاده در مدل پیشنهادی[۴۲]

نام مجموعه داده	تعداد پروژه ها	تعداد ویژگی ها	متوسط تلاش	بیشترین تلاش	کمترین تلاش
Cocomo 81	۶۳	۱۷	۶۸۳	۱۱۴۰۰	۵,۹
Albrecht	۲۴	۸	۲۲,۸	۱۰۵,۲	۲,۹
Maxwell	۶۲	۲۶	۸۲۲۳	۶۳۶۹۴	۵۸۳

در این مقاله سعی شده است نتایج بدست آمده از روش‌های مختلف تخمین جهت مقایسه مدل پیشنهادی با مدل‌های رایج و بدست آوردن میزان دقت مدل پیشنهادی استفاده شود. از جمله این مدل‌های OABE، مدل LSE، مدل MLEF، مدل RTM، مدل RIR، مدل LMS، مدل CBR، مدل LOG+OLS، مدل OLS+BC، مدل ROR و مدل GA می‌باشند[۴۳،۴۲،۴۱]. برای ارزیابی بهتر عملکرد مدل پیشنهادی از دو الگوریتم به صورت جداگانه استفاده شده است. این دو الگوریتم از نظر عملکرد بسیار شبیه به هم و در عین حال خصوصیات خواص به خود را دارا می‌باشند. سعی شده است، ارزیابی عملکرد بین مدل پیشنهادی و تغییر الگوریتم ترکیبی پرداخته شود. تنظیم پارامترها مربوط به الگوریتم‌ها را می-توان در جدول ۲ مشاهده کرد.

۷-۱- مجموعه داده کوکومو

برآورد تلاش انجام شده مدل‌های مختلف بر روی مجموعه داده کوکومو نشان می‌دهد که مجموعه داده کوکومو می‌تواند برای بررسی عملکرد مدل پیشنهادی ما مفید واقع

شود. نتایج حاصله از مدل پیشنهادی نشان می‌دهد که این مدل بر روی این مجموعه داده چندان خوب عمل نکرده است و نتایج نسبتاً ضعیفی بدست آمده است. نمودار۱ این مشاهدات را نشان می‌دهند.

جدول ۲: مقدار دهی پارامترهای موجود در الگوریتم ازدحام ذرات و کرم شب تاب

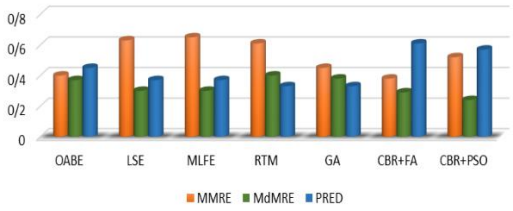
الگوریتم	پارامترها
FA	$\alpha=1$, $\beta 0=1.5$, $\gamma = 2$
PSO	C1= C2=2 , W=1
Population size = 50, Number of iteration = 100, Fitness function =(MMRE + MdMRE)-PRED (0.25)	

نمودار۱: نتایج بدست آمده توسط مدل‌های مختلف بر روی مجموعه داده کوکومو

کمترین مقدار بدست آمده برای پارامتر MMRE مربوط به مدل OABE با میزان ۰/۵۰ و بدترین مقدار آن مربوط به GA با میزان ۱/۵۹ می باشد. که در مدل ترکیبی پیشنهادی ما ترکیب CBR+FA در رتبه چهارم قرار می‌گیرد و مدل ترکیبی پیشنهادی CBR+PSO نیز در رتبه پنجم قرار گرفته است. همچنین کمترین مقدار بدست آمده برای پارامتر MdMRE مربوط به مدلLOG+OLS با میزان ۰/۲۸ و بدترین مقدار آن مربوط به مدل GA با میزان ۰/۸۱ می‌باشد. در مورد این پارامتر مدل پیشنهادی CBR+FA در رتبه ۲ قرار دارد و مدل پیشنهادی CBR+PSO در رتبه ۹ قرار دارد. در مورد ستون آخر PRED بهترین مقدار مربوط به مدل LOG+OLS با مقدار ۰/۴۹ و بدترین مقدار بدست آمده برای این پارامتر مربوط به مدل MLFE وGA با مقدار ۱۴,۰ می باشد و مدل پیشنهادی CBR+FA در رتبه ۲ و مدل پیشنهادی CBR+PSO و مدل LSE به طور مشترک رتبه ۳ قرار دارند. مدل پیشنهادی نتوانسته است در هیچ کدام از سه پارامتر عملکردی نتایج مطلوبی را بدست آورد. دلیل این امر تنوع بسیار محدود پروژه‌ها در پایگاه داده کوکومو است. بطوریکه دامنه نوسان ۱۵ محرک هزینه در این پایگاه داده بسیار محدود است، یعنی تغییر پارامترهای تابع شباهت تاثیر چندانی در دقت تخمین‌ها ندارد و لذا کشف پروژه‌های شبیه به هم در این مجموعه داده نمی‌تواند یک راه حل مناسب باشد.

۷-۲- مجموعه داده آلبرشت

مجموعه داده آلبرشت است این پایگاه داده دارای ۲۴ پروژه توسعه یافته توسط زبان نسل سوم و شامل ۸ ویژگی عددی می‌باشد که ممکن است تلاش پروژه را تحت تاثیر قرار دهد. بهترین مقدار بدست آمده برای پارامتر MMRE توسط مدل OABE با مقدار ۰/۴۰ و بدترین مقدار آن مربوط به مدل MLFE با مقدار ۰/۶۵ می‌باشد. همچنین بهترین مقدار بدست آمده برای پارامتر MdMRE مدل پیشنهادی CBR+PSO با مقدار ۰/۲۴ و مدل پیشنهادی CBR+FA با مقدار ۰/۲۹ در رتبه دوم قرار می‌گیرد، و بدترین مقدار آن مربوط به مدل RTM با مقدار ۰/۴ می‌باشد بهترین مقدار بدست آمده برای پارامتر PRED نیز مربوط به مدل پیشنهادی CBR+FA با مقدار ۰/۶۱ و مدل پیشنهادی CBR+PSO با مقدار ۰/۵۷ در رتبه دوم قرار می‌گیرد. بدترین مقدار آن مربوط به مدل‌های RTM و GA با مقدار ۰/۳۳ می‌باشد. نتایج نشان می‌دهد که مدل پیشنهادی بر روی این مجموعه داده آلبرشت بسیار خوب عمل کرده است و دقت بدست آمده از مدل پیشنهادی بر روی این مجموعه داده مطلوب است. در نمودار۲ این نتایج به تصویر کشیده است.



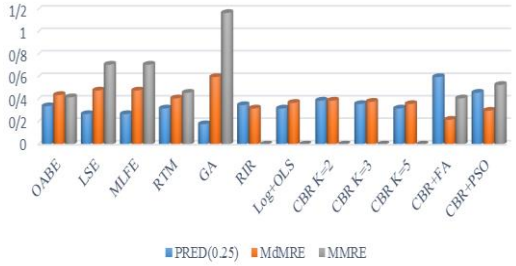
نمودار۲: نتایج بدست آمده توسط مدل‌های مختلف بر روی مجموعه داده آلبرشت

نتایج مدل پیشنهادی روی این مجموعه داده بهبود چشمگیری داشته است و مهم‌ترین دلیل آن را می‌تواند وزن‌دهی مناسب و انتخاب دقیق تابع راه‌حل و شباهت می-باشد. مجموعه داده آلبرشت یک مجموعه داده نسبتاً

استاندارد است و همین امر منجر به افزایش دقت مدل پیشنهادی شده است.

۷-۳- مجموعه داده ماکسول

محققان زیادی از این مجموعه داده برای ارزیابی عملکرد مدل پیشنهادی خود استفاده کرده‌اند. بهترین مقدار بدست آمده برای پارامتر MMRE توسط مدل پیشنهادی CBR+FA با مقدار ۰/۴۱ و بدترین مقدار آن مربوط به مدل‌های MLFE و LSE با مقدار ۰/۷۱ می‌باشد. همچنین بهترین مقدار بدست آمده برای پارامتر MdMRE مدل پیشنهادی CBR+FA با مقدار ۰/۲۲ و مدل پیشنهادی CBR+PSO با مقدار ۰/۳۰ در رتبه دوم قرار می‌گیرد، و بدترین مقدار آن مربوط به مدل GA با مقدار ۰/۶۰ می‌باشد بهترین مقدار بدست آمده برای پارامتر PRED نیز مربوط به مدل پیشنهادی CBR+FA با مقدار ۰/۶۰ و مدل پیشنهادی CBR+PSO با مقدار ۰/۴۶ در رتبه دوم قرار می‌گیرد. بدترین مقدار آن مربوط به مدل‌های MLFE و LSE با مقدار ۲۷,۰ می‌باشد.



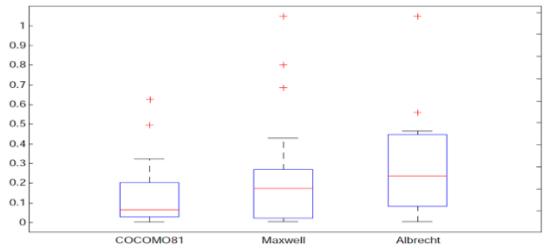
نمودار۳: نتایج بدست آمده توسط مدل‌های مختلف بر روی مجموعه داده ماکسول

نتایج نشان می‌دهد که مدل پیشنهادی بر روی این مجموعه داده بسیار خوب عمل کرده است و نتایج دقیق بدست آمده از مدل پیشنهادی بر روی این مجموعه داده مطلوب است. در نمودار ۳ این نتایج را به تصویر کشیده است.

۷-۴- تحلیل معیارهای عملکردی

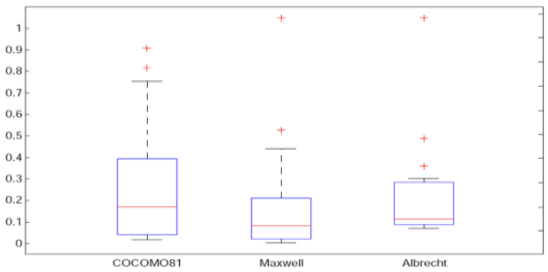
هدف اصلی مدل پیشنهادی بررسی پارامترهای مهم ارزیابی بر روی مجموعه داده‌های بکار رفته می‌باشد. در این بخش مقایسه اجمالی بر روی سه مجموعه داده انجام می‌شود:

آ- یکی از پارامترهای محاسباتی در فرآیند تخمین توسعه نرم‌افزار پارامتر خطای نسبی می‌باشد. این پارامتر خطای نسبی همه پروژه‌ها را نشان می‌دهد که با نمودار جعبه‌ای نشان داده می‌شود. نمودار ۴ میزان MRE بدست آمده توسط مدل پیشنهادی CBR+FA بر روی سه مجموعه داده استفاده شده در این پایان‌نامه را نشان می‌دهد.



نمودار ۴: نمودار جعبه ای MRE مدل پیشنهادی CBR+FA بر روی سه مجموعه داده

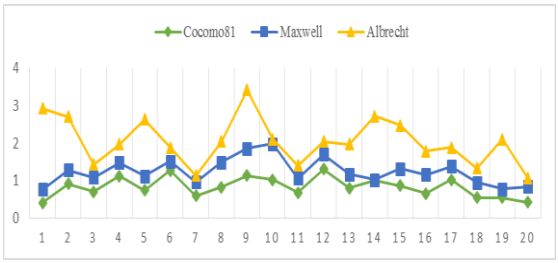
در نمودار ۴ مشاهده می‌شود که مدل پیشنهادی عملکرد قابل قبولی داشته است. مقدار MRE برای پروژه‌های مختلف پراکندگی نسبتاً بالایی دارد و میانه MREها نیز عدد قابل قبولی است، میانه MREهای موجود کمتر از ۰/۵ می باشد که مناسب می باشد. نمودار ۵ میزان MRE بدست آمده توسط مدل پیشنهادی CBR+PSO بر روی سه مجموعه داده استفاده شده در این پایان نامه را نشان می‌دهد.



نمودار ۵: نمودار جعبه ای MRE مدل پیشنهادی CBR+PSO بر روی سه مجموعه داده

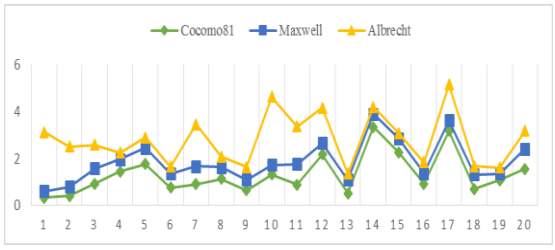
در نمودار ۵ مشاهده می‌شود که مدل پیشنهادی عملکرد قابل قبولی داشته است. مقدار MRE برای پروژه‌های مختلف پراکندگی کمی دارد و میانه MREها نیز عدد قابل قبولی است (کمتر از۰/۵).

ب- از پارامترهای محاسباتی دیگر در فرآیند تخمین توسعه نرم‌افزار پارامتر میانگین خطای نسبی می‌باشد. این پارامتر میانگین خطای نسبی همه پروژه‌ها را نشان می‌دهد، نمودار ۶ میزان MMRE بدست آمده توسط مدل پیشنهادی CBR+FA بر روی سه مجموعه داده استفاده شده در این پایان نامه را نشان می‌دهد.



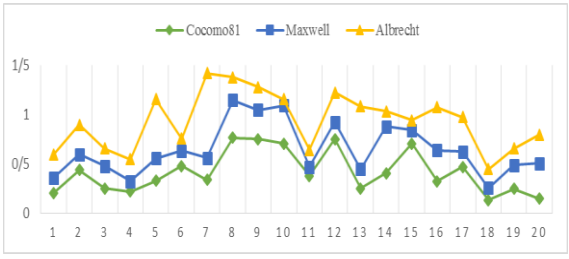
نمودار۶: نتایج MMRE بدست آمده بر روی سه مجموعه داده توسط مدل CBR+FA

نمودار ۷ میزان MMRE بدست آمده توسط مدل پیشنهادی CBR+PSO بر روی سه مجموعه داده استفاده شده در این پایان نامه را نشان می‌دهد.

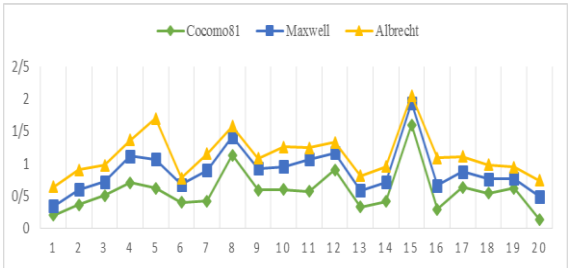


نمودار۷: نتایج MMRE بدست آمده بر روی سه مجموعه داده توسط مدل CBR+PSO

از دیگر پارامترهای مهم در تخمین تلاش نرم افزار پارامتر MdMRE می‌باشد. مدل‌های ابداع شده سعی در کمینه کردن این پارامتر مهم دارند. نتایج بدست آمده بر روی مدل‌های CBR+FA و CBR+PSO را با توجه به نمودارهای ۸ و ۹ مورد بررسی قرار گرفته است.



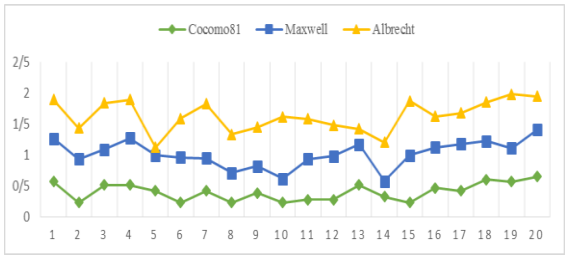
نمودار۸: نتایج MdMRE بدست آمده بر روی سه مجموعه داده توسط مدل CBR+FA



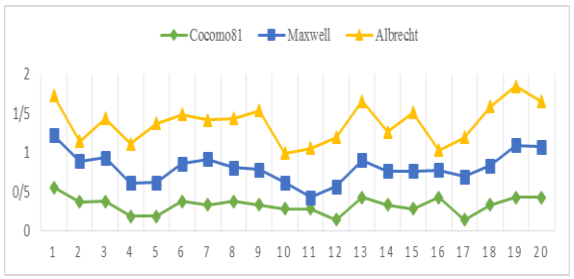
نمودار۹: نتایج MdMRE بدست آمده بر روی سه مجموعه داده توسط مدل CBR+PSO

با توجه به نتایج بدست آمده در شکل (۱۵)مشخص است که پارامتر MdMRE در پایگاه داده کوکومو بهتر نشده است، و عملکرد دو الگوریتم بسیار نزدیک به هم می‌باشد.

پ- پارامتر PRED (۰/۲۵) درصد پیش بینی خطا را بیان می‌کند. این پارامتر از مهمترین پارامترهایی است که در فرآیند تخمین توسعه نرم افزار است، که باید بیشینه شود. نتایج بدست آمده بر روی مدل‌های CBR+FA و CBR+PSO را با توجه به نمودارهای ۱۰ و ۱۱ مورد بررسی قرار گرفته است.



نمودار۱۰: نتایج PRED(۰,۲۵) بدست آمده بر روی سه مجموعه داده توسط مدل CBR+FA



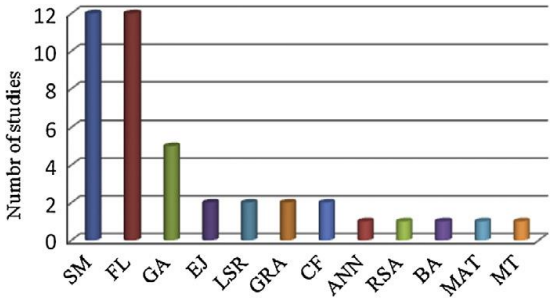
نمودار۱۱: نتایج PRED(۰,۲۵) بدست آمده بر روی سه مجموعه داده توسط مدل CBR+PSO

با توجه به نتایج بدست آمده مدل پیشنهادی ما نتوانست در مجموعه داده کوکومو خوب عمل کند. عملکرد مدل پیشنهادی CBR+FA بهتر عمل کرده است.

۸- مقایسه تحلیلی :

مدل CBR احتمال زیاد بهترین گزینه برای برآورد تلاش نرم‌افزار است. احتمال دارد راه‌حل تجویز شده برای مشکل با عملکرد پروژه نرم افزاری مغایرت داشته باشد و باعث بروز مشکلاتی شود [۴۲]. الگوهای مختلف در ترکیب با تکنیک CBR اغلب برای غلبه برا چالش‌های متعدد مربوط به ویژگی‌های مختلف و انتخاب معیارهای شباهت و سازگاری آنها با استراتژی مورد استفاده در پروژه نرم افزاری است [۴۱]. نمودار ۱۲ نشان می‌دهد که اغلب روش‌های آماری

SM[1] و منطق فازی FL[2] (هر یک با احتمال ۱۸٪)، در ترکیب با روش قیاس ، بعد از این دو الگوریتم ژنتیک GA[3] (با احتمال ۸٪)، و روش‌های دیگر کمتر مثلاً مدل های LSR[4] و غیره (هر یک با احتمال ۳٪) استفاده می‌شوند.



نمودار۱۲ : توزیع تکنیکهای در ترکیب شده با روش CBR [۴۰]

۹- بحث

در این مقاله یک مدل ترکیبی جهت بهبود دقت تخمین تلاش پروژه‌های نرم افزاری ارائه شده است. در مدل پیشنهادی روش استدلال مبتنی بر رویداد با الگوریتم بهینه سازی کرم شب تاب ترکیب شده است. الگوریتم FA در مرحله اول ویژگی های یک پروژه نرم افزاری را به کمک تابع تشابه در مدل CBR به بهترین نحو وزن دهی می‌کند. سپس این وزن‌ها برای ارزیابی عملکرد خدمات جدید مورد آزمون قرار می‌گیرند. الگوریتم FA پیشنهاد وزن مناسب به شبیه ترین ویژگی‌ها و تابع راه حل وزن های بهینه سازی شده و تعداد نزدیکترین همسایگان را در مدل CBR جهت بهبود دقت برآورد تلاش انتخاب می‌کند. محققان نتایج را با بهترین روش‌های تخمین چند سال اخیر از جمله الگوریتم GA و مدل CBR به تنهایی مقایسه کرده‌اند. از آنجایی که الگوریتم ژنتیک برای تنظیم اختلاف بین یک پروژه جدید و پروژه‌های قبلی با توجه به تمام ویژگی‌های پروژه مورد استفاده قرار گرفت، هدف اصلی استفاده از الگوریتم GA بهینه سازی ضریب وزن هر یک از ویژگی‌های و اندازه گیری تفاوت ویژگی‌ها با استفاده از یک تابع هدف می‌باشد. مدل LSE در تلاش است که شباهت بین دو پروژه را بر اساس پروژه‌هایی قبلی اندازه گیری کند، عملکرد خوبی در مقایسه با الگوریتم EBA با مدل (EBA0) نشان داده است، اما این روش بر روی مجموعه‌های بسیار محدود قابل قبول است. مدل RTM فرض می‌کند که پروژه‌هایی با ساختار مشابه باید ویژگی‌های پروژه کارایی مشابهی داشته باشند. نتایج به دست آمده نشان می‌دهد که روش پیشنهادی جدید بهتر از مدل‌های قبلی تخمین تلاش پروژه‌های نرم افزاری را انجام می دهد. به عنوان مثال مدل پیشنهادی در مقایسه با روش درخت تصمیم‌گیری، روش-های رگرسیون، روش‌های مبتنی بر آنالوگ و الگوریتم‌های نرم محاسباتی عملکرد بهتری نشان داده است. علاوه بر این، برای تعیین دقت مدل پیشنهادی میانگین خطایی نسبی و درصد پیش بینی خطا مورد استفاده قرار گرفته است. برای ارزیابی نتایج در مدل پیشنهادی از سه مجموعه استفاده شده است. مدل پیشنهادی نمی‌تواند نتایج مناسبی در هر یک از سه پارامتر عملکردی ارائه دهد. به دلیل اینکه انواع مختلفی از پروژه‌ها در مجموعه داده Cocomo وجود دارد به نحوی که هزینه ۱۵ محرک این مجموعه داده به تدریج در دامنه محدود و محدودتر می‌شود. به عبارت دیگر، پارامترهای تابع شباهت اثر قابل توجهی بر دقت برآوردها ندارد، بنابراین کشف پروژه‌های مشابه نمی‌تواند یک راه حل مناسب در این مجموعه داده باشد. مدل پیشنهادی بر روی داده‌های آلبرشت عملکرد عملکرد خوبی نشان داد، وزن-دهی مناسب و انتخاب دقیق توابع راه حل و شباهت عملکرد مدل پیشنهادی به طور قابل ملاحظه‌ای بهبود یافته است. مجموعه داده آلبرشت نسبتاً استاندارد می‌باشد. این واقعیت که دقت مدل پیشنهادی بهبود یافته است از نتایج بدست آمده بر روی پایگاه داده ماکسول مشخص می شود. نتایج نشان می‌دهد که عملکرد مدل پیشنهادی بر روی مجموعه داده ماکسول به خوبی بهبود یافته است. موفقیت این مدل در همبستگی قوی بین اندازه و تلاش و در دسترس بودن اندازه گیری یک ویژگی مناسب می‌باشد. بر اساس تحقیقات، نتایج مناسب در پایگاه داده های آلبرشت و ماکسول به دست آمد که نشان دهنده دقت مناسب مدل پیشنهادی می‌باشد. این مدل قابلیت انعطاف پذیری بالایی دارد، زیرا هیچ پیش شرطی و فرضی برای ساخت و استفاده از آن لازم نیست. مدل پیشهادی می‌تواند در بهبود عملکرد روش‌های تخمین تلاش مفید باشد. در زیر یافته مدل پیشنهادی به صورت کلیدی بیان شده است:

- مدل CBR به تنهایی عملکرد خوبی ندارد. بنابراین، برای بهبود کارایی باید با یک الگوریتم

[1] Statistical Methods
[2] Fuzzy Logic
[3] Genetic Algorithms
[4] Least Squares Regression

تکاملی مانند GA, PSO و یا FA ترکیب شود.

- مدل پیشنهادی نسبت به مدل‌های قبلی عملکرد بهتری داشته است.
- برای ارزیابی بهتر مدل پیشنهادی از سه مجموعه داده با تنوع استفاده شده است.
- میزان بهبود عملکرد مدل پیشنهادی در جداول نشان داده شده است. اما متوسط نرخ عملکرد ٪۳۵ می‌باشد. این مقدار دقت و صحت فرآیند تخمین تلاش را نشان می‌دهد.
- در نهایت می‌توان گفت مقاله ما می‌تواند برای محققان دیگر منبع خوبی برای بهبود عملکرد مدل‌های جدید باشد.

۱۰- نتیجه گیری و کارهای آینده

مدیریت موفق پروژه نقش موثری در بهره‌وری سازمان ایفا می‌نماید. بنابراین، استفاده از روش‌های موثر تخمین هزینه و تلاش، تاثیر قابل توجهی را در مدیریت پروژه دارد. تخمین تلاش مورد نیاز برای ساخت سیستم اطلاعاتی، یکی از دغدغه‌های مهم مدیریت پروژه تلقی می‌شود. الگوهای تخمین هزینه‌ای که در مراحل اولیه ساخت پروژه، با حداقل اطلاعات موجود از پروژه، هزینه ساخت سیستم را تخمین می‌زنند، سودمند و مورد نیاز هستند. روش‌های تخمین هزینه مناسب، امکان کنترل موثر زمان و هزینه ساخت سیستم را فراهم می‌نماید. در نتیجه الگوهای متعددی برای تخمین تلاش سیستم مطرح شده است. در این مقاله سعی شد که با ارائه یک مدل ترکیبی میزان دقت تخمین تلاش تا حدودی بهبود پیدا کند. در مدل پیشنهادی از ترکیب الگوریتم بهینه سازی کرم شب تاب و روش برآورد استدلال مبتنی بر رویداد استفاده شده است. در مدل پیشنهادی الگوریتم بهینه سازی کرم شب تاب با پیشنهاد مناسب ترین تابع شباهت و تابع راه حل و همچنین بهینه‌ترین وزن‌ها و انتخاب تعداد نزدیکترین همسایه‌ها به روش برآورد استدلال مبتنی بر رویداد سعی در افزایش میزان دقت تخمین تلاش را دارد. همچنین از دو معیار میانگین خطای نسبی و درصد پیش بینی جهت تعیین میزان دقت مدل پیشنهادی استفاده شده است. به منظور ارزیابی نتایج حاصله از مدل پیشنهادی سه پایگاه داده بکار گرفته شده است. میزان دقت مدل پیشنهادی طبق بررسی های صورت گرفته با سایر مدل‌های مذکور در هر پایگاه داده نشان می دهد که در پایگاه داده آلبرشت و ماکسول نتایج خوبی بدست آمده است. همچنین مدل پیشنهادی در پایگاه داده کوکومو نتایج مطلوبی بدست نیاورده است. دلیل آن می‌تواند تنوع کم مقادیر ویژگیها در پایگاه داده کوکومو باشد که فرآیند وزن-دهی را دچار مشکل می‌کند. روش پیشنهادی می تواند در بهبود عملکرد روش‌های برآورد تلاش مفید واقع شود

مراجع

[1] W. L. Du, D. Ho, and L.F. Capretz, "Improving software effort estimation using neuro-fuzzy model with SEER-SEM", arXiv preprint arXiv:1507.06917, 2015.

[2] A. Bawa M.R. Chawla, and D. Karnal, "Experimental analysis of effort estimation using artificial neural network", International Journal of Electrical and Computer Science Engineering, Vol. 1(3), 2012, pp. 1817-1824.

[3] J. Popović, and D. Bojić, "A comparative evaluation of effort estimation methods in the software life cycle", Computer Science and Information Systems. Vol. 9(1), 2012, pp.456- 484.

[4] S. Kad, and V. Chopra, "Software development effort estimation using soft computing", International Journal of Machine Learning and Computing, Vol. 2(5), 2012, pp. 437-439.

[5] D. Kashyap, and A. K. Misra, "An approach for software effort estimation using fuzzy numbers and genetic algorithm to deal with uncertaint", Computer Science and Information Technology (CS & IT), 2013, pp. 57-66.

[6] A. Sheta, D. Rine, and A. Ayesh, "Development of software effort and schedule estimation models using soft computing techniques", IEEE Transaction, 2008, pp. 978-1.

[7] R. K. Clemmons, "Project estimation with use case points", The Journal of Defense Software Engineering, 2006, pp. 18-22 .

[8] B. Peschi, "Recommending effort estimate method for software project management", Web Intelligence and Intelligent Agent Technologies, 2009, pp. 77-80.

[9] N. AZ Adem, and Z. M Kasirun, "Automating function point analysis based on the functional and the non-functional requirement text", Computer and Automation Engineering, 5, 2010, pp.664-669.

[10] S. Kumari, and SH. Pushkar, "Performance analysis of the software cost estimation methods: A review", International Journal of Advanced Research in Computer Science and Software Engineering, Vol.3(7), 2013, pp. 229-238.

[11] P. K. Suri, "Comparative analysis of software effort estimation techniques", International Journal of Computer Application, Vol.48(21), 2012, pp. 1-8.

[12] J. Parkash, "Cocomo ii model parameter optimization using pso and effort estimation", Journal of Information Technology & Mechanical Engineering, Vol. 1(4), 2014, pp.1-11.

[13] A. K. Bardsiri, and S.M. Hashemi, "Software Effort Estimation: A Survey of Well-known Approaches", International Journal of Computer Science Engineering (IJCSE), Vol.3(1), 2014, pp. 46-50.

[14] V. Khatibi, and D. N. Jawawi, "Software Cost Estimation Methods: A Review", Journal of Emerging Trends in Computing and Information Sciences, Vol.1(2), 2010, pp.21-29.

[15] S. Chattopadhyay, S. Banerjee, F. A. Rabhi, and U. R. Acharya, "A Case-Based Reasoning system for complex medical diagnosis", Expert Systems, Vol.30(1), 2013, pp. 12-20.

[16] M. Azzeh, "Adjusted case-based software effort estimation using bees optimization algorithm", International Conference on Knowledge-Based and Intelligent Information and Engineering Systems. Springer, Vol.6882, 2011, pp. 315-324.

[17] G. R. Finnie, G. E. Wittig, and A. J. M. Desharnais, "comparison of software effort estimation techniques: using function points with neural networks, case-based reasoning and regression models", Journal of Systems and Software; Vol. 39(3), 1997, pp.281-289.

[18] M. J. sanjari, , "A new support vector model-based imperialist competitive algorithm for time estimation in new product development projects", Robotics and Computer-Integrated Manufacturing, Vol. 29(1), 2013, pp. 157-168.

[19] V. Khatibi, and D. A. N. jawawi, "Software cost estimation methods: a review", Journal of Emerging Trends in Computing and Information Sciences, Vol. 2(1), 2011, pp. 21-29.

[20] A. Khatibi Bardsiri, and S.M. Hashemi, "A differential evolution-based model to estimate the software services development effort", Journal of Software: Evolution and Process, Vol. 28(1), 2016, pp. 57-77.

[21] J. Cherng Lin and H. Y. Tzeng, "Applying particle swarm optimization to estimate software effort by multiple factors software project clustering", IEEE, Computer Symposium International, 2010, pp.1039–1044.

[22] Z. Ashegi Dizaji, R. Ahmadi, H.Gholizadeh and F. Soleimanian Gharehchopogh, "A bee colony optimization algorithm approach for software cost estimation", International Journal of Computer Applications, Vol. 104(12), 2014, pp.41-44.

[23] Ph. D. Živadinović, Z. Dragan Maksimović, A. Damnjanović, S. Vujčić and Z. Serbia, "Methods of effort estimation in software engineering", International Symposium Engineering Management And Competitiveness, 2011, pp.417-422.

[24] F. Soleimanian Gharehchopogh and Z. Asheghi Dizaji, "A new approach in software cost estimation with hybrid of bee colony and chaos optimizations algorithms", Magnt Research Report, Vol. 2(6), 2014, pp. 1263-1271.

[25] V. Khatibi, D. A.N. Jawawi, S.Z. Mohd Hashim and E. Khatibi, "A pso-based model to increase the accuracy of software development effort estimation", Software Qual J , Vol. 21, 2012, pp.501-526.

[26] B. Nasiri and M. Meybodi, "Speciation based firefly algorithm for optimization in dynamic environments", International Journal of Artificial Intelligence. Vol. 8(S12), 2012, pp. 118-132.

[27] J. Kwiecień, and B. Filipowicz, "Firefly algorithm in optimization of queueing systems", Bulletin of the Polish Academy of Sciences: Technical Sciences, Vol. 60(2), 2012, pp. 363-368.

[28] I. Attarzadeh, and S. Hock Ow, "Proposing a new high performance model for software cost estimation", International Conference on Computer and Electrical Engineering, Vol. 2, 2009, pp. 112-119.

[29] R. Ketata, H.Bellaaj, M.Chtourou and M.B.Amer, "Adjustment of membership functions, generation and reduction of fuzzy rule base from numerical data", Malaysian Journal of Computer Science, Vol. 20(2), 2007, pp. 147-169.

[30] A. Hamdy, "Fuzzy logic for enhancing the sensitivity of cocomo cost model", Journal of Emerging Trends in Computing and Information Sciences, Vol. 3(9), 2012, pp.1292-1297.

[31] K. Krishnanand, and D. Ghose, "Glowworm swarm optimization for simultaneous capture of multiple local optima of multimodal functions", Swarm intelligence, Vol. 3(2), 2009, pp. 87-124.

[32] X. S. Yang, "Engineering optimization: an introduction with metaheuristic applications", John Wiley and Sons,INC, Publication, 2010.

[33] L. I. O. Adriano, "GA-based method for feature selection and parameters optimization for machine learning regression applied to software effort estimation", Information and Software Technology, Vol. 52(11), 2010, pp.1155-1168.

[34] Z. Dan, "Improving the accuracy in software effort estimation: Using artificial neural network model based on particle swarm optimization", Service Operations and Logistics, and Informatics (SOLI) , IEEE International Conference on, 2013, pp.180-185.

[35] A. Khatibi Bardsiri, S.M. Hashemi, and M. Razzazi, "GVSEE: a new global model to estimate software services development effort", Journal of the Chinese Institute of Engineers: Vol. 39 , 2016, pp. 1-12.

[36] T. Madhusudan, J.L. Zhao, and B. Marshall, "A case-based reasoning framework for workflow model management", Data and Knowledge Engineering, Vol.50(1), 2004, pp. 87-115.

[37] E. Kocaguneli, T. Menzies, A. Bener, and J. W. Keung, "Exploiting the essential assumptions of analogy-based effort estimation", IEEE Transactions on Software Engineering, Vol. 38(2), 2012, pp. 425-438.

[38] A. Idri, F. azzahra Amazal, and A.Abran, "Analogy-based software development effort estimation: A systematic mapping and review", Information and Software Technology; Vol. 58, 2015, pp. 206–230.

[39] A. K. Bardsiri, S.M. Hashemi, and M. Razzazi, " Statistical Analysis of The Most Popular Software Service Effort Estimation Datasets", Journal of Telecommunication, Electronic and Computer Engineering (JTEC), Vol. 7(1), 2015, pp.8.96-7.

[40] y. Elsheikh, M. Alseid, and M. Azzeh, "An optimized analogy-based project effort estimation", International Journal of Advanced Computer Science and Applications, Vol. 5(4), 2014, pp. 6-11.

[41] S. Meysam Mousavi, R. Tavakkoli-Moghaddam, B. Vahdani, H. Hashemi, and M.J.sanjari, "A new support vector model-based imperialist competitive algorithm for time estimation in new product development projects", Robotics and Computer-Integrated Manufacturing, Vol.29(1), 2013, pp.157-168.

[۴۲] ع. خطیبی بردسیری، س.م. هاشمی و م. رزازی، " ارائه یک مدل جدید جهت تخمین تلاش لازم برای توسعه سرویس های نرم افزاری "، مدل سازی در مهندسی, ۱۵(۴۹)، صفحه ۲۰-۲۲، ۲۰۱۷