فصلنامه سیستم‌های پردازشی و ارتباطی چندرسانه‌ای هوشمند
Intelligent Multimedia Processing and Communication Systems (IMPCS)

Fall 2022, 3 (3), 19-27
DOR: 20.1001.1.27832570.1401.3.3.3.5

Rcceived: 26 June 2022
Accepted: 9 Aug 2022

مقاله پژوهشی

Software Risk Prediction by Grasshopper Optimization Algorithm and Machine Learning

Bahar Ahmadi Naaghani[1], Hadi Khosravi Faarsaani2*, Taghi Javdani Gandomani[3]

1. MSc. Student, Department of Computer Engineering, Shahrekord University, Shahrekord, Iran.
2. Assistant Professor, Department of Computer Engineering, Shahrekord University, Shahrekord, Iran.
(Corresponding Author) khosravi@eng.sku.ac.ir
3. Assistant Professor, Department of Computer Engineering, Shahrekord University, Shahrekord, Iran. javdani@sku.ac.ir

Abstract

Introduction: Real-time Localization System is one of the several technologies in order to track objects and target locations. Satellite-based positioning systems such as GPS are not reliable due to strong signal attenuation, especially in indoor environments, as well as the multipath effect. Distance-based methods generally include two main parts: 1. Information based on a time interval or TOA and TDOA 2. Positioning based on information received from distance measurement. In this paper, a novel positioning system is represented, which is based on Ultra-Wideband technology. The proposed system is used to utilize the TDOA algorithm.
Method: The UWB positioning system uses the ultra-wideband Gaussian pulse, in which the time range of the pulses is less than 0.5 nanoseconds with very high accuracy. Usually, positioning systems based on ultra-wideband have several source nodes that calculate the reception time by deriving the pulse pattern from the target node. The source node calculates the total average sending and receiving time between the source node and the target node.
Finding: On average, the proposed module has an error of less than 1 meter in about 21 centimeters of distance measurement, which is a very acceptable accuracy compared to GPS-based systems. This error can be solved by creating an offset in the software.
Conclusion: Considering the weaknesses of the current positioning systems, in this paper, a system based on ultra-wideband using a dwm1000 module and ZigBee communication was introduced. The test results report an accuracy of about 10 cm for the module, and with the interpolation and calibration of personalized environments, the amount of error can be greatly reduced. Considering the error of 10 to 100 meters in expensive modules with GPS technology, this accuracy is an acceptable and negligible error. This system can be used for the internal positioning of robots, hospital personnel, and factories.

Keywords: risk management, Possible risks, Locust optimization, Classification methods, Simulated Annealing algorithm, Metahuristical algorithms, Supporf vector machine classification.

فصلنامه سیستم‌های پردازشی و ارتباطی چندرسانه‌ای هوشمند
Intelligent Multimedia Processing and Communication Systems (IMPCS)

پیش‌بینی ریسک پروژه‌های نرم‌افزار توسط الگوریتم بهینه‌سازی ملخ و یادگیری ماشین

تاریخ دریافت: ۱۴۰۱/۰۴/۱۵
تاریخ پذیرش: ۱۴۰۱/۰۵/۱۸

دوره سوم، پاییز ۱۴۰۱
شماره سوم، صص: ۱۹-۲۷

بهار احمدی‌ناغانی[۱]، هادی خسروی‌فارسانی[۲*]، تقی جاودانی‌گندمانی[۳]

۱. دانشجوی ارشد، گروه مهندسی کامپیوتر، دانشگاه شهرکرد، شهرکرد، ایران.
۲. استادیار، گروه مهندسی کامپیوتر، دانشگاه شهرکرد، شهرکرد، ایران. (نویسنده مسئول) khosravi@eng.sku.ac.ir
۳.استادیار، گروه مهندسی کامپیوتر، دانشگاه شهرکرد، شهرکرد، ایران. javdani@sku.ac.ir

چکیده: توسعه نرم‌افزار را می‌توان فعالیتی دانست که از انواع پیشرفت‌های فناورانه استفاده‌می‌کند و نیاز به دانش بالایی دارد. به همین دلیل، هر پروژه توسعه نرم‌افزاری حاوی عناصر عدم قطعیت است که به‌عنوان ریسک پروژه شناخته‌می‌شود. موفقیت یک پروژه توسعه نرم‌افزار به‌شدت به میزان ریسک مربوط به هر فعالیت پروژه بستگی‌دارد. لذا، به‌عنوان یک مدیر پروژه، آگاهی از خطرات کافی نیست. جهت دستیابی به یک نتیجه موفق، یک مدیر پروژه باید بتواند تمام ریسک‌های اصلی را شناسایی، سپس ارزیابی، اولویت‌بندی و درنهایت مدیریت کند. مدیریت ریسک بر شناسایی ریسک‌ها و درمان مناسب با آن‌ها تمرکزدارد. پروژه‌های نرم‌افزاری دارای ریسک‌های فردی یا کلی هستند. برخی از این ریسک‌ها به یک فعالیت خاص و برخی دیگر به پروژه مرتبط است. معمولاً ریسک‌ها ابتدا شناسایی‌شده و با فعالیت‌های پروژه مرتبط می‌شوند. تعیین چگونگی رفتار افراد برای دستیابی به اهداف فعالیت استراتژیک برای شناسایی خطرات است. استفاده از الگوریتم‌ها و فن‌های مختلف برای شناسایی ریسک‌های نرم‌افزاری همواره موردتوجه متخصصین بوده است. هدف این مطالعه، پیش‌بینی ریسک‌های پروژه‌های نرم‌افزاری به کمک الگوریتم بهینه‌سازی ملخ می‌باشد. در این روش انتخاب ویژگی و کاهش آن توسط الگوریتم بهینه‌سازی ملخ انجام‌می‌شود و برای طبقه‌بندی ریسک و ویژگی‌ها از روش‌های طبقه‌بندی ماشین بردار استفاده‌می‌شود.

واژه‌های کلیدی: مدیریت ریسک، توسعه نرم‌افزار، بهینه‌سازی ملخ، روش‌های طبقه‌بندی، الگوریتم‌های فرا ابتکاری، طبقه‌بندی ماشین بردار پشتیبان

۱. مقدمه

در فرآیند توسعه نرم‌افزار درصد زیادی از شکست پروژه‌ها مربوط به مسائل مدیریتی و درصد کمتری از مشکلات فنی است. مسائل مدیریتی شامل مشکلاتی که می‌تواند به ساختار پروژه، منابع پروژه، روش‌های برنامه‌ریزی و خرید مشتری و مدیریت ناکارآمد ریسک مربوط شود. درواقع، مدیریت ریسک فرایندی برای شناسایی، رفع و پیش‌بینی ریسک‌های احتمالی است. به‌عبارت‌دیگر، این فرآیند مربوط به تمام فعالیت‌های است که برای کاهش عدم قطعیت مرتبط با وظایف یا رویدادهای خاصی انجام‌می‌شود.

در مهندسی نرم‌افزار، فرآیند توسعه نرم‌افزار، برای تقسیم‌کار توسعه نرم‌افزار به مراحل یا فرآیندهای فرعی کوچک‌تر، موازی یا متوالی برای بهبود طراحی، مدیریت محصول و همچنین به‌عنوان چرخه عمر توسعه نرم‌افزار (SDLC) شناخته‌می‌شود. این روش ممکن است شامل از پیش تعریف محصولات و مصنوعات خاص باشد که توسط یک گروه پروژه برای توسعه یا حفظ یک برنامه کاربردی ایجاد و تکمیل می‌شود [۱].

هنگام شناسایی خطرات یک سیستم نرم‌افزاری، مهم است که همه خطرات احتمالی، میزان شدت هر ریسک و همه پیامدهای بالقوه هر یک را بدانیم. مراحل اقدام برای کاهش یا کنترل هر ریسک بر اساس آگاهی کامل از همه ریسک‌ها تعیین‌می‌شود. این رویکرد پیشگیرانه برای مدیریت ریسک به توسعه‌دهندگان نرم‌افزار اجازه‌می‌دهد تا پروژه‌ها را در بازه‌های زمانی و بودجه مورد انتظار خود به پایان برسانند [۲].

خطرات را می‌توان در تمام بخش‌های یک فرآیند شناسایی‌کرد. چندین مدل ریسک برای کمک به شناسایی ریسک و طبقه‌بندی وجوددارد. این مدل‌ها ممکن است بر مدیریت ریسک‌های تجاری تمرکزکنند تا ریسک-های محصول. بااین‌حال، محصولات نرم‌افزاری بسیار مهم هستند و بنابراین توسعه‌دهندگان نیز باید از روش‌های مورداستفاده برای تعیین خطرات ایمنی و یکپارچگی محصولات خودآگاه باشند [۴].

۱,۱. انگیزه و هدف

پیش‌بینی ریسک‌های احتمالی می‌تواند به گروه‌های نرم‌افزاری کمک نماید تا قبل از وقوع این موارد، چاره مناسبی برای پیش‌گیری از رخداد ریسک‌های احتمالی اندیشیده و هزینه‌های چالش‌های احتمالی را کاهش‌دهند. علی‌رغم کارهای متعدد انجام‌شده در خصوص مدیریت ریسک پروژه‌های نرم‌افزاری، به‌نظرمی‌رسد که بتوان با کمک الگوریتم‌های نوین و بهره‌گیری از فن‌های یادگیری ماشین فرایند شناسایی یا پیش‌بینی ریسک‌های احتمالی را بهبودبخشید. در این راستا، شناسایی مدل نرم‌افزاری مناسب با درنظرگرفتن عوامل خطر ضروری است؛ بنابراین، در این مطالعه، مدل‌های پیش‌بینی ریسک نرم‌افزار مورداستفاده در صنعت را از طریق تعامل با متخصصان توسعه نرم‌افزار موردبررسی قراردادند.

بررسی کارهای انجام‌گرفته در حوزه الگوریتم‌های بهینه‌سازی از جمله الگوریتم ژنتیک، الگوریتم بهینه‌سازی ذرات و .. نشان‌داده‌است که نسخه‌های ارائه‌شده الگوریتم‌ها جوابگوی تمامی مسائل بهینه‌سازی نبوده و در برخی از اوقات با گرفتارشدن در بهینه‌های محلی، راه‌حل مطلوب را تولید نمی‌کنند. برای غلبه بر این محدودیت پژوهشگران از رویکردهای مختلفی استفاده‌می‌کنند. برخی الگوریتم‌های بهینه‌سازی را با یکدیگر ترکیب می‌کنند تا با تغییر در ساختار الگوریتم پایه از محدودیت موجود پیشگیری‌نمایند؛ اما این امر نیز نمی‌تواند به‌صورت کامل محدودیت موجود را برطرف‌کند. این در حالی است که پیاده‌سازی چنین رویکردی سختی خاص خود را داشته و پیچیدگی الگوریتم را نیز بالامی‌برد.

از طرفی استفاده از الگوریتم‌های شبکه عصبی پیچشی و دیگر شبکه‌های عصبی ممکن است با افزایش ابعاد داده‌ها منجر به کاهش دقت شوند. در این روش از الگوریتم بهینه‌سازی ملخ که یک الگوریتم فراابتکاری است جهت دسته‌بندی داده‌ها استفاده و پرداخته‌می‌شود تا هم ابعاد داده‌ها در صورت امکان کاهش یابد و از طرفی با انتخاب ویژگی‌ها مناسب به افزایش دقت طبقه‌بندی کمک‌شود.

الگوریتم مورد استفاده در این مقاله، الگوریتم بهینه سازی‌ملخ می باشد که یک الگوریتم فراابتکاری است و از طبعیت ایده‌گرفته‌شده است.این الگوریتم با بهرگیری از رفتار یار ملخ‌ها در طبیعت به جستجوی غذا می‌پردازد. این الگوریتم در سال ۲۰۱۷ توسط دو دانشجوی ایرانی در استرالیا به عنوان یک مسئله بهینه‌سازی معرفی‌شد[۶].

درروش پیشنهادی، با رویکرد بهبود روش‌های پیشین هدف پیش‌بینی ریسک پروژه‌های نرم‌افزاری می‌باشد که با کمک الگوریتم بهینه‌سازی ملخ و طبقه‌بند ماشین بردار پشتیبان انجام‌می‌شود.

این الگوریتم که از بهینگی محلی رنج نمی‌برد، به انتخاب ویژگی¬های مهم پرداخته‌است. دیانت‌های این ویژگی‌ها توسط الگوریتم طبقه‌بند ماشین بردار پشتیبان طبقه‌بندی‌می‌شوند.

در این مطالعه از الگوریتم بهینه‌سازی ملخ برای بهبود صحت طبقه‌بندی و افزایش دقت و نیز کاهش ویژگی استفاده‌شده است. نتایج به‌دست‌آمده نشان‌داده‌است که این امر کاربردی و قوی بودن الگوریتم بهینه‌سازی ملخ را برای حل مسائل با فضاهای جستجوی ناشناخته و واقعی را ثابت‌می‌کند.

در این روش با ترکیب الگوریتم پیشنهادی با الگوریتم ماشین بردار پشتیبانی برای به‌دست‌آوردن نتایج بهتر و قابل‌قبول‌تری درداده‌های با ابعاد زیاد در جهت کاهش بعد و انتخاب ویژگی استفاده‌می‌شود که توجه افراد بسیار از محققین و پژوهشگران را به خود جلب‌کرده‌است.

با مقایسه این روش یعنی الگوریتم بهینه‌سازی ملخ با ترکیب خروجی دیگر روش‌های فرا ابتکاری همانند الگوریتم ازدحام ذرات، الگوریتم ژنتیک و همچنین الگوریتم تحلیل مؤلفه برای کاهش ابعاد پرداخته می‌شود که در هر دو مجموعه داده روش پیشنهادی دارای صحت ۸۸ درصد و ۹۹,۳۶ درصد به ترتیب بوده‌است.

ادامه مقاله به شرح زیر سازماندهی می‌شود: بخش دوم به بررسی پیشینه پژوهش می‌پردازد. بخش سوم بیان‌کننده راه‌حل پیشنهادی را شرح‌می‌دهد. بخش چهارم شامل نتایج حاصل از تست داده‌ها و بخش پنجم نتیجه‌گیری نهایی را بیان می‌کند.

۲. پیشینه پژوهش

در این بخش به مروری بر مفاهیم مشکلات و روش‌های شناسایی ریسک-های نرم‌افزاری توسط الگوریتم‌های مختلف پرداخته‌می‌شود که برخی از این روش‌ها عبارتند از:

Gouthaman P و همکاران [۱]، یک چارچوب برای مجموعه داده دریافتی از طریق پرسشنامه ارائه‌می‌دهند که در آن طبقه‌بندی کننده‌های یادگیری ماشین و پیش‌بینی خطر برای هر یک از مدل‌های نرم‌افزاری شناسایی‌شده اعمال‌می‌شود که با نتایج به‌دست‌آمده مدیران محصولات نرم‌افزاری می‌توان محصولات مناسب نیاز خود را با درصد پیش‌بینی خطر شناسایی‌کنند.

N. Paltrinieri و همکاران [۷] یک روش ارزیابی ریسک بر اساس یادگیری ماشین که درواقع یک مدل شبکه عصبی عمیق توسعه داده‌شده و برای سناریوی دورزدن دکل حفاری نفت و گاز آزمایش‌شده است. نتایج دقت معقولی را برای پیش‌بینی‌های DNN[۱] و مناسب بودن برای غلبه بر چالش‌های ارزیابی ریسک نشان‌می‌دهد مدل شبکه عمیق عصبی سطوح بالایی از دقت را گزارش‌می‌کند.

L. Bai et و همکاران [۸] ابتدا عوامل خطرساز منابع پروژه را با درنظرگرفتن وابستگی متقابل پروژه مشخص‌می‌شوند و سپس مدل شبکه بیزی برای دستیابی به یک نتیجه مطلوب باید تمام ریسک‌های اصلی پروژه را شناسایی، ارزیابی، اولویت‌بندی و مدیریت‌کرد.

L. Radu [۹] مدلی را برای توسعه نرم‌افزار چابک که پیش‌بینی پروژه با استفاده از شبکه‌های بیزین انجام‌می‌شود. در این روش دودسته عوامل اصلی که بر تلاش موردنیاز تأثیرمی‌گذارد، مشخص‌می‌شود یکی کیفیت کار گروهی و دیگری ویژگی‌های داستان‌های کاربر است این مدل می‌تواند به تیم‌های چابک برای برآورد بهتر تلاش نرم‌افزاری کمک‌کند. این روش مدلی از BN را برای کمک به تیم‌های چابک برای تخمین تلاش موردنیاز در مورد پروژه‌های نرم‌افزاری پیشنهادکرد. برای ساختن مدل، مروری بر ادبیات انجام‌شد و با کارشناسان برای شناسایی عوامل اصلی مؤثر بر تلاش موردنیاز بحث ردند.

L. Pan و همکاران [۱۰]، یک مدل بهبودیافته شبکه عصبی کانولوشن را برای پیش‌بینی نقص درون پروژه پیشنهادشد و نتایج خود را با نتایج موجود CNN[۲] و یک مطالعه تجربی مقایسه‌کرد. آزمایش بر اساس مدل ارزش‌گذاری یا اعتبارسنجی K-fold10 بود نتایج تجربی نشان‌داد که مدل بهبودیافته CNN با مدل CNN موجود قابل‌مقایسه است و از مدل‌های پیشرفته یادگیری ماشین به‌طور قابل‌توجهی برای WPDP[۳] پیشی‌گرفت.

Sai Mohan و همکاران [۱۱] ارائه یک نمای کلی از مدل‌های برآورد هزینه نرم‌افزار و خلاصه‌کردن نقاط قوت، ضعف، دقت، میزان داده‌های موردنیاز و فن‌های اعتبارسنجی مورداستفاده آن‌ها است. یافته ها نشان‌می‌دهد، به‌طورکلی، مدل‌های مبتنی بر شبکه عصبی از سایر فن‌های برآورد هزینه بهتر عمل‌می‌کند. در بررسی محققان از مدل‌های مختلف، به این نتیجه رسیدند که روش شبکه عصبی عملکرد بسیار قوی‌تری نسبت به سایر روش‌ها دارند و باعث تخمین دقت بالاتری است.

بررسی کارهای انجام‌گرفته در حوزه الگوریتم‌های بهینه‌سازی نشان داده است که نسخه‌های ارائه‌شده الگوریتم‌ها جوابگوی تمامی مسائل بهینه‌سازی نبوده و در برخی از اوقات با گرفتار شدن در بهینه‌های محلی، راه‌حل مطلوب را تولیدنمی‌کنند. برای غلبه بر این محدودیت پژوهشگران از رویکردهای مختلفی استفاده می‌کنند. برخی الگوریتم‌های بهینه‌سازی را با یکدیگر ترکیب‌می‌کنند تا با تغییر در ساختار الگوریتم پایه از محدودیت موجود پیشگیری نمایند؛ اما این امر نیز نمی‌تواند به‌صورت کامل محدودیت موصوف را برطرف‌کند. این در حالی است که پیاده‌سازی چنین رویکردی سختی خاص خود را داشته و پیچیدگی الگوریتم را نیز بالا می‌برد. از طرفی استفاده از الگوریتم‌های شبکه عصبی کانولوشن و دیگر شبکه‌های عصبی ممکن است با افزایش ابعاد داده‌ها منجر به کاهش دقت شوند. در این روش از الگوریتم بهینه‌سازی ملخ که یک الگوریتم فرا ابتکاری است جهت دسته‌بندی داده‌ها استفاده و پرداخته می‌شود تا هم ابعاد داده‌ها در صورت امکان کاهش یابد و از طرفی با انتخاب ویژگی‌ها مناسب به افزایش دقت طبقه‌بندی کمک شود.

۳. راه‌حل پیشنهادی

مجموعه داده اول در روشی برای یک مدل یادگیری ماشین برای پیش بینی ریسک پروژه است که هدف از این کار طراحی یک مدل یادگیری ماشینی برای پیش‌بینی احتمال داشتن یک پروژه دارای مسائلی است که ارزش آن را در گزارش ریسک مدیریت پروژه نشان‌می‌دهد. تدوین گزارش ریسک پروژه به تلاش قابل‌توجهی نیاز دارد، زیرا تحلیلگران باید بسیاری از گزارش‌ها و اسناد مرتبط را بررسی‌کنند تا تعیین‌کنند که آیا یک پروژه دارای ریسک بالایی است یا خیر. بنابراین، با آموزش یک مدل متمایز، می‌توانیم پروژه‌هایی را که از نظر آماری دارای ریسک بالایی هستند اولویت‌بندی‌کرد و هزینه تهیه گزارش را کاهش‌داد. همچنین مهم‌ترین عواملی را که به ریسک پروژه کمک‌می‌کنند، مانند مدیران، محدوده، فصلی بودن و غیره بررسی‌کرد. مجموعه داده دریافتی در این روش استفاده‌شده‌است. درروش پیشنهادی ابتدا با دریافت داده‌ها از سایت اینترنتی، مجموعه پروژه‌های استخراج‌شده از Microsoft project server می‌باشد که در روشی به صورتی که پروژه‌های شرکتی ذخیره

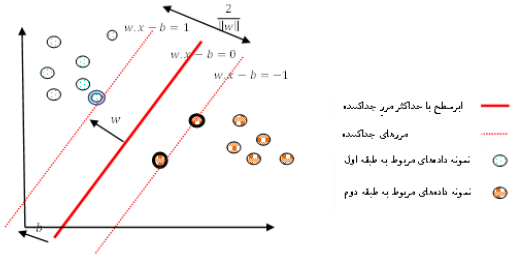
می‌شوند، استفاده‌می‌کند. به‌طور خاص، داده‌های پروژه‌های فناوری اطلاعات از مارس ۲۰۱۵ تا آگوست ۲۰۱۶ تجزیه‌وتحلیل شده‌است. روش پیشنهادی بر روی مجموعه داده دریافتی دیگر[1] که از یک سایت اینترنتی به‌دست‌آمده و نشان‌دهنده مدلی برای پیش‌بینی خطر یک نیاز نرم‌افزاری بر اساس مجموعه داده منبع باز می باشد که به‌پیش پردازش داده‌ها پرداخته‌می‌شود جهت حذف داده‌های تکراری و نویز و خطا و نامعتبر (اعداد منفی و صفر) سپس کاهش ویژگی توسط الگوریتم بهینه‌سازی ملخ صورت‌می‌گیرد؛ و نیز از ماشین بردار پشتیبان جهت طبقه‌بندی داده‌ها استفاده‌می‌کنیم. همان‌طور که گفته‌شد، در روش پیشنهادی، با رویکرد بهبود روش‌های پیشین هدف پیش‌بینی ریسک پروژه‌های نرم‌افزاری می‌باشد که با کمک الگوریتم بهینه‌سازی ملخ و طبقه‌بند ماشین بردار پشتیبان انجام‌می‌شود.

الگوریتم بهینه‌سازی ملخ به‌صورت تصادفی ویژگی‌های را جهت کاهش ابعاد انتخاب‌می‌کند. درواقع از بین ویژگی‌های موجود ویژگی را به‌عنوان ویژگی اصلی انتخاب‌می‌کند. برخلاف روش زیرمجموعه‌ای ویژگی‌های دیگر را ذخیره‌می‌کند تا درصورتی‌که ویژگی انتخابی نیاز ما را برطرف نکرد از بقیه ویژگی‌ها جهت کاهش ابعاد استفاده‌کند. تابع مورداستفاده برای افزایش صحت طبقه‌بندی و کاهش ویژگی از تابع برازندگی می‌باشد. درواقع محاسبه هر راه‌حل طبق تابع برازندگی است که به طبقه‌بندی ماشین بردار پشتیبان و صحت و تعداد ویژگی‌های انتخاب‌شده وابسته است. انتخاب ویژگی به‌عنوان یک مشکل بهینه‌سازی چندهدفی است که دو هدف متضاد را بیان‌می‌کند. حداقل تعداد ویژگی‌ها انتخاب‌شده و صحت طبقه‌بندی بالاتر. هرچه تعداد ویژگی کمتر باشد، صحت طبقه‌بندی بالاتر است و راه‌حل بهینه‌تر خواهدبود. هر راه‌حل بر اساس تابع برازندگی محاسبه خواهدشد که وابسته به صحت و تعداد ویژگی‌های انتخاب‌شده و طبقه بند ماشین بردار پشتیبان است. به‌منظور تعادل بین تعداد ویژگی‌های انتخاب‌شده در هر راه‌حل (حداقل) و صحت طبقه‌بندی (حداکثر)، عملکرد برازندگی برای جست‌وجوی عوامل در الگوریتم ملخ با استفاده از فرمول زیر محاسبه می‌شود:

$$fitness = \alpha\gamma_R(D) + \beta\frac{|R|}{|N|} \qquad (۱)$$

که در آن $\gamma_R(D)$، نرخ خطای طبقه‌بندی از طبقه‌بند داده‌شده (در اینجا از طبقه بند K نزدیک‌ترین همسایه) استفاده‌شده‌است، $|R|$ زیرمجموعه‌های قدرتمند ویژگی‌های انتخاب‌شده و $|N|$ مجموع ویژگی‌های انتخاب‌شده در دیتاست است. دو پارامتر α و β دو پارامتر مربوط به اهمیت کیفیت طبقه‌بندی و طول زیرمجموعه هستند که به‌صورت $\alpha \in [0و1]$ و $\beta = [1-\alpha]$ تعریف‌می‌شوند.

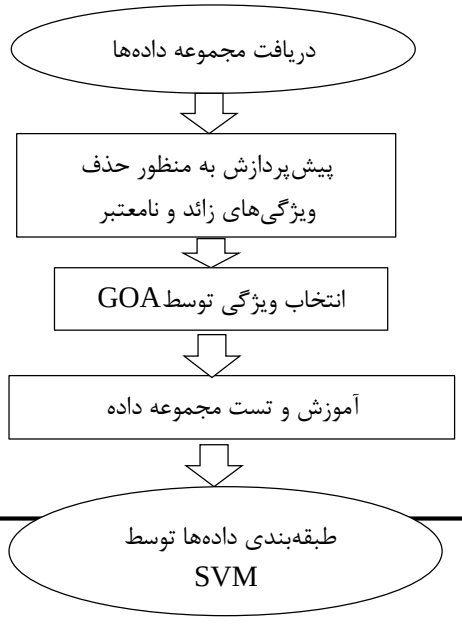
طبقه‌بندی داده‌ها توسط روش ماشین بردار پشتیبان است که این روش‌های طبقه‌بندی خطی، سعی‌دارند که با ساختن یک ابر سطح (که عبارت است از یک معادله خطی)، داده‌ها را از هم تفکیک‌کنند. روش طبقه‌بندی ماشین بردار پشتیبان که یکی از روش‌های طبقه‌بندی خطی است، بهترین ابر سطحی را پیدامی‌کند که با حداکثر فاصله، داده‌های مربوط به دوطبقه را از هم تفکیک‌کند. شکل زیر تصویری از یک مجموعه داده متعلق به دو کلاس است که روش ماشین بردار پشتیبان بهترین ابر سطح را برای جداسازی آن‌ها انتخاب‌می‌کند.



شکل ۲: ابر سطحی با بیشترین مرزِ جداکننده همراه با مرزهای جداکننده دیگر برای طبقه‌بندی نمونه داده‌های دو طبقه متفاوت است. نمونه‌های روی مرزها بردارهای پشتیبان نام دارند.

در این شکل داده‌ها دوبعدی هستند یعنی هر داده تنها از دو متغیر تشکیل‌شده‌است. دسته‌بند به ماهیت داده بستگی دارد، اگر امکان جداسازی خطی یا تقریباً خطی داده‌ها وجودداشته‌باشد، انتخاب باید روی دسته‌بند با حداکثر حاشیه و یا طبقه ند بردار پشتیبانی قرارگیرد. در عمل بدون توجه به ماهیت داده‌ها از توابع هسته استفاده‌می‌گردد. در مواردی که امکان جداسازی داده‌ها به‌صورت خطی یا تقریباً خطی فراهم‌باشد، تابع هسته خطی به‌کارگرفته‌شده و متناسب با آن تنظیم-می‌شود؛ بنابراین، برای طبقه‌بندی ماشین بردار پشتیبان با هسته RBF، از ۷۰ درصد داده‌های برای آموزش و ۳۰ درصد داده‌ها برای تست استفاده می‌شود. به‌صورت کلی می‌توان‌گفت که برای پیش‌بینی ریسک نرم‌افزاری از کاهش‌دهنده الگوریتم بهینه‌سازی ملخ استفاده می‌شود و از طبقه‌بند ماشین بردار پشتیبان به سبب مزایای ذکرشده برای طبقه‌بندی ویژگی‌ها استفاده‌شده است، به این صورت که پس‌ازآن که کاهش ویژگی توسط الگوریتم بهینه‌سازی ملخ صورت‌گرفت یعنی الگوریتم تعدادی ویژگی کمتر و بهتر را به‌صورت تصادفی انتخاب و آن‌ها را با برچسب واقعی جهت طبقه‌بندی به طبقه‌بند ماشین بردار پشتیبان واگذارمی‌کند که ماشین بردار بر اساس اعتبارسنجی K-fold10 داده‌ها را طبقه‌بندی‌می‌کند.

فلوچارت روش پیشنهادی در شکل(۱) نشان داده‌شده است.



شکل ۱: ساختار کلی روش پیشنهادی

۴. نتایج

همان‌طور که در شکل ۲ قابل‌مشاهده است، مجموعه داده‌ها در ستون دوم با نام risk و دارای دو برچسب بدون ریسک و با ریسک با مقدار صفر و یک هستند.

isk	status	compliance	report_count	has_schedule	scope	office	month	year	day	risk_previous1	risk_previous2	risk_previous3	proje
.0	tramitando para contratação	0.9879	32	1	Corporativo	Corporativo	3	2015	17.13932980599647	0	0	0	0.014
.0	tramitando para contratação	0.9879	32	1	Corporativo	Corporativo	4	2015	6.0	1	0	0	0.024
.0	tramitando para contratação	0.9879	32	1	Corporativo	Corporativo	5	2015	11.0	1	1	0	0.030
.0	em andamento	0.9879	32	1	Corporativo	Corporativo	6	2015	29.0	1	1	1	0.036
.0	em andamento	0.9879	32	1	Corporativo	Corporativo	6	2015	1.0	1	1	1	0.036
.0	não iniciado	1.0	8	1	Estruturante	CENIN	6	2015	17.13932980599647	0	0	0	0.009
.0	em andamento	0.9879	32	1	Corporativo	Corporativo	7	2015	31.0	0	0	0	0.036
.0	não iniciado	1.0	8	1	Estruturante	CENIN	7	2015	17.13932980599647	0	0	0	0.009
.0	em andamento	0.9879	32	1	Corporativo	Corporativo	8	2015	31.0	1	0	0	0.041
.0	em andamento	1.0	8	1	Estruturante	CENIN	8	2015	31.0	0	0	0	0.008

شکل ۳: بخشی از مجموعه داده اول

علاوه بر مجموعه فوق، روش پیشنهادی بر روی مجموعه داده دیگری نیز تست و ارزیابی‌شده‌است که از سایت اینترنتی دانلود شده-است. شکل ۴ بیانگر بخشی از مجموعه داده دوم را نشان‌می‌دهد.

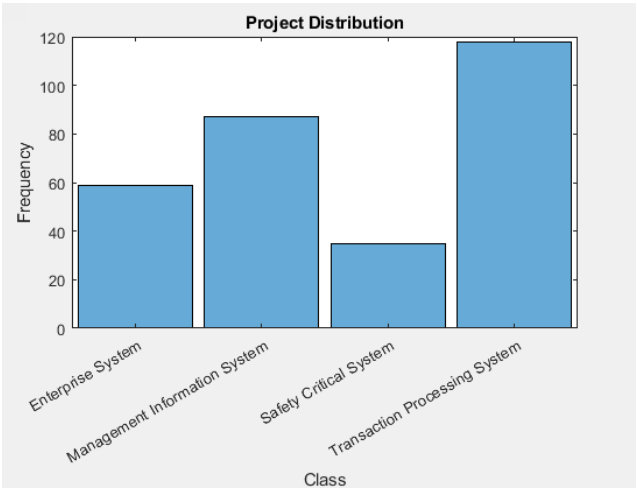
requirement_text	project	requirement_category	risk_type	probability	risk_magnitude	risk_impact	risk_dimension	requirement_affected	fixing	fix	fix_priority	r
The system shall display all the products that...	Transaction Processing System	Functional	Budget	10	Negligible	high	Requirements	9	1	10	95.715385	
The system shall allow user to select the prod...	Transaction Processing System	Functional	Quality	22	Very Low	catastrophic	Requirements	7	2	11	35.900000	
The system shall display all the available com...	Transaction Processing System	Functional	Schedule	33	Low	high	User	5	1	3	35.692308	
The system shall enable user to add one or mor...	Transaction Processing System	Functional	Personal	40	Medium	moderate	Project complexity	D	3	2	44.100000	
The system shall notify the user about any con...	Transaction Processing System	Functional	Budget	100	High	Low	planning and control	4	2	1	85.776923	

شکل ۴: بخشی از مجموعه داده دوم

این مجموعه داده برخی ستون‌ها شامل متن و اعداد هستند. در ستون آخر نیز بانام req_aff_fix_affects است که شامل ۵ دسته است که از ۱ تا ۵ شماره‌گذاری شده‌است.

ماتریکس مورداستفاده درزمینه یادگیری ماشینی و به‌طور خاص مسئله مربوط به طبقه‌بندی به‌صورت آماری، زمانی که یک ماتریکس درهم‌ریختگی که تحت عنوان ماتریس خطا نیز معرفی‌می‌گردد که شامل یک طرحی از جدول خاصی می‌باشد و این امکان را می‌دهد که با تجسم کارکرد یک الگوریتم، معمولاً یک الگوریتم تحت نظارت (در یادگیری بدون نظارت) که معمولاً ماتریس تطبیق نیز نامیده‌می‌شود، هر ردیف از ماتریس نمونه‌های موجود در یک کلاس واقعی را نشان-می‌دهد درحالی‌که هر ستون نمونه‌های یک کلاس پیش‌بینی‌شده را نشان‌می‌دهد.

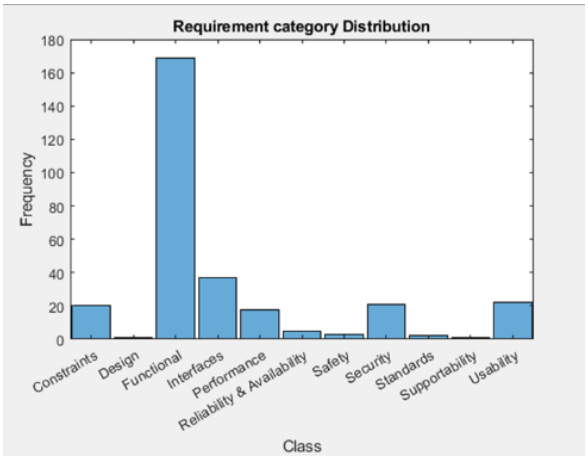
شکل ۵ بیانگر نمایش و نحوه توزیع داده‌ها در مجموعه داده دوم می‌باشد.



شکل ۵: توزیع انواع پروژه‌ها

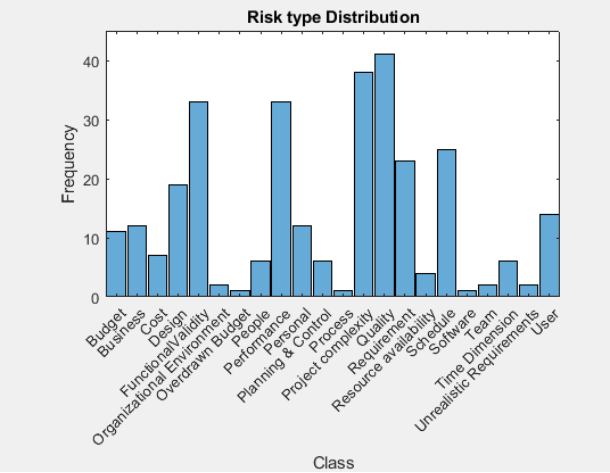
توزیع پروژه در انواع مختلف: سیستم سرمایه‌گذاری، سیستم اطلاعاتی مدیریت، سیستم بحرانی، سیستم پردازش تراکنشی توزیع‌شده است.

توزیع گروه‌بندی موردنیاز نیز در شکل ۶ نشان داده‌شده است:



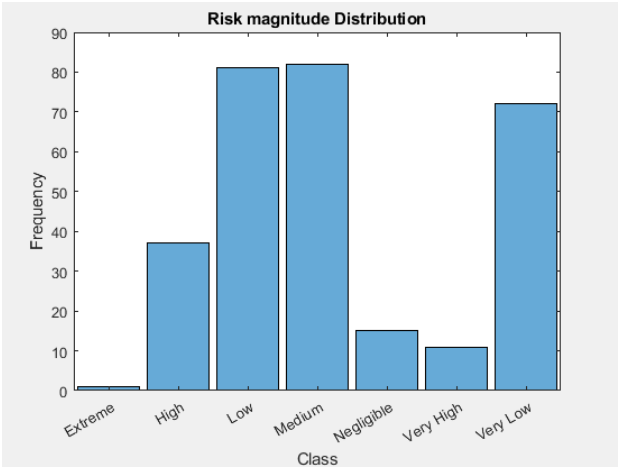
شکل ۶: توزیع پراکندگی نیازمندی‌ها

شکل ۷ توزیع مقدار ریسک در حالت‌های بالا، پایین، متوسط و... را نشان‌می‌دهد.



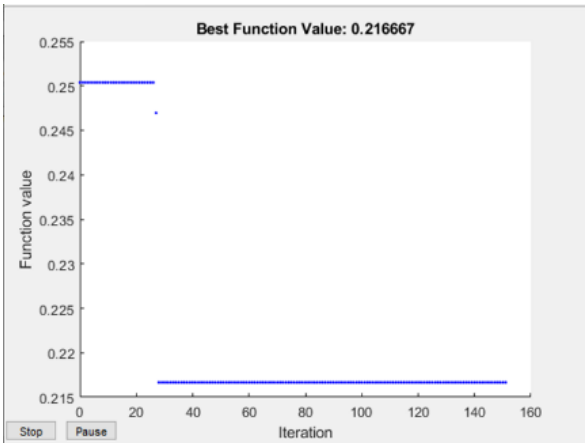
شکل۷: توزیع مقدار ریسک

حال پس از مشاهده توزیع مقدار ریسک در شکل ۷، توزیع مقدار تأثیر ریسک نیز در شکل ۸ قابل‌مشاهده است.



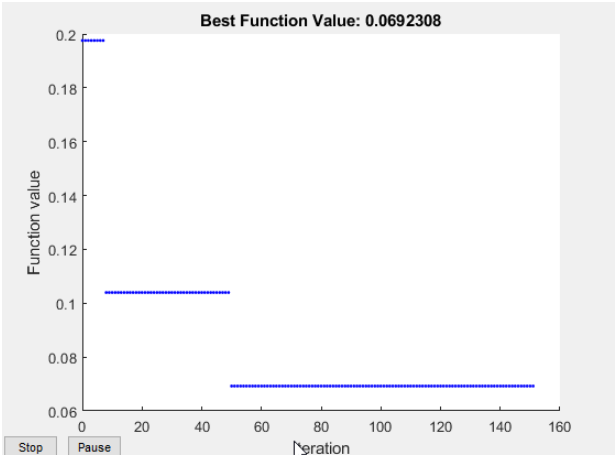
شکل۸: توزیع مقدار تأثیر ریسک

در ادامه نتایج شبیه‌سازی با استفاده از الگوریتم بهینه‌سازی ملخ بررسی‌می‌شود. بدین‌صورت که پس پیش‌پردازش و نرمال‌سازی به کاهش ابعاد پرداخته‌شد. لذا، پس از تحلیل و نمایش توزیع داده‌ها در شکل ۹ و ۱۰ کاهش ابعاد توسط مجموعه اول و دوم پرداخته می‌شود.



شکل ۹: همگرایی مجموعه داده اول

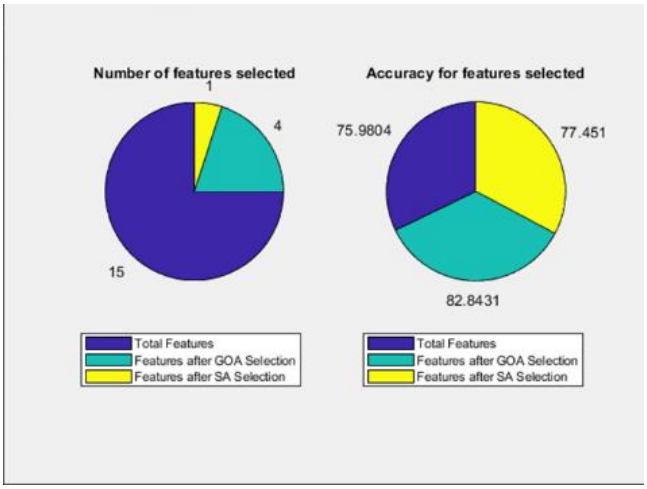
در شکل ۹ روش پیشنهادی پس از ۱۶۰ بار تکرار به میزان همگرایی مناسبی می‌رسد که نشان از عملکرد خوب الگوریتم بهینه‌سازی ملخ و یافتن مقادیر بهینه و کاهش ابعاد این مجموعه داده‌ها است.



شکل ۱۰: همگرایی مجموعه داده دوم

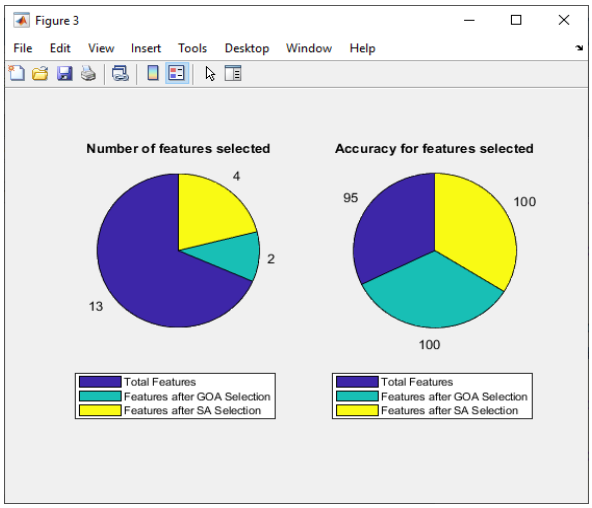
در شکل ۱۰ نیز می‌توان دریافت، همگرایی در این مجموعه داده دوم به مقدار بسیار قابل‌توجهی رسیده‌است به‌گونه‌ای که می‌توان به بهترین دقت رسید.

حال با توجه به کاهش بعد، در شکل ۱۱ و ۱۲ روش پیشنهادی الگوریتم ملخ با کمک روش الگوریتم تبرید مقایسه‌شده‌است به ترتیب نشان داده‌شده‌است.



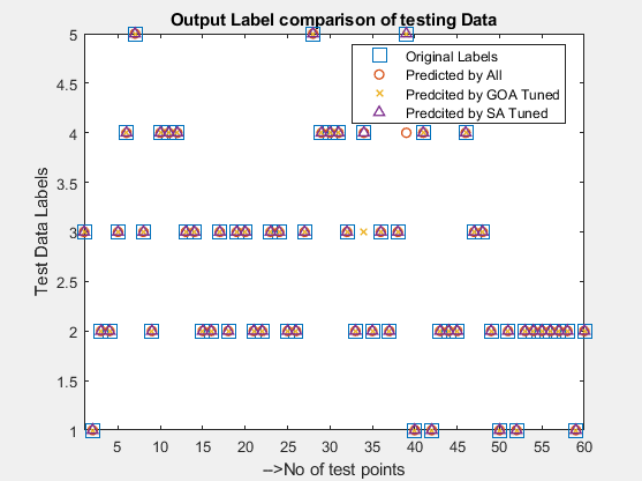
شکل ۱۱: صحت به‌دست‌آمده توسط کاهش ابعاد با مجموعه داده اول

همان‌طور که از شکل (۱۱) قابل‌مشاهده است، روش الگوریتم شبیه‌سازی تبرید بدون کاهش ابعاد به صحت ۷۷,۴۵ درصد رسیده است و این در حالی است که با کمک الگوریتم بهینه‌سازی ملخ به صحت ۸۲,۸۴ درصد رسیده است. در این مجموعه ۴ ویژگی risk_previous 2,risk_previous 1, day, mount هستند که به‌عنوان ویژگی‌های بهینه درنظرگرفته‌شده‌اند که توسط تابع برازندگی الگوریتم بهینه‌سازی انتخاب‌شده‌اند. بدین‌صورت داده‌هایی بیشتر مساوی ۰,۵ به‌عنوان ویژگی انتخاب‌شده و کوچک‌تر از آن به معنای عدم انتخاب ویژگی‌ها است.



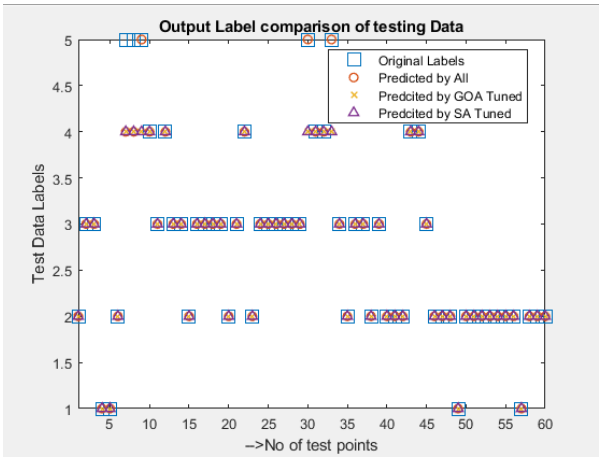
شکل ۱۲: صحت به‌دست‌آمده توسط کاهش ابعاد با مجموعه داده دوم

در شکل ۱۲ با دو ویژگی عددی به صحت ۱۰۰ درصد رسیده‌است که البته با میانگین از اجراهای مختلف به‌دقت ۹۹,۳۶ رسیده‌است. این در حالی است که بدون کاهش ابعاد به صحت ۹۵ درصد رسیده‌است. در این مجموعه دو ویژگی probability و requirement_affected به‌عنوان ویژگی بهینه به‌دست‌آمد. در ادامه و در شکل ۱۴ و ۱۵ حالت بصری مقادیر پیش‌بینی توسط روش GOA و SA نشان داده‌شده‌است.



شکل ۱۴: حالت بصری مقادیر پیش‌بینی‌شده با مجموعه داده اول

همان‌طور که در شکل ۱۴ نشان‌داده‌شده‌است، قرارگرفتن ضربدر به‌عنوان پیش‌بینی برچسب درست توسط الگوریتم بهینه‌سازی ملخ علامت مثلث پیش‌بینی توسط تبرید و علامت مربع به معنای پیشگویی همه داده‌ها است.

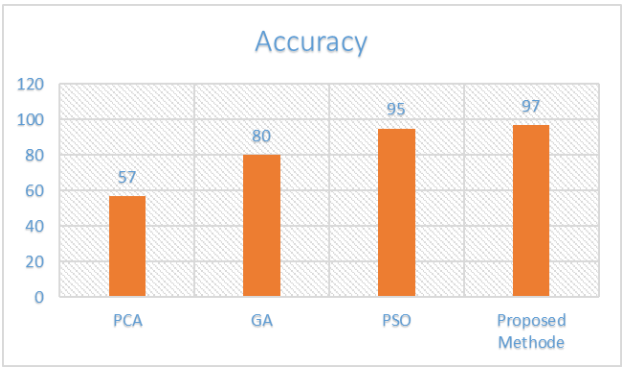


شکل ۱۵: حالت بصری مقادیر پیش‌بینی‌شده با مجموعه داده دوم

با توجه به شکل‌های ۱۴ و ۱۵ مقادیر پیش‌بینی‌شده توسط هر دو مجموعه به‌خوبی توسط دو الگوریتم بهینه‌سازی ملخ و الگوریتم تبرید نشان داده‌شده‌است که در برچسب‌های ۴ الگوریتم SA دارای مشکلاتی به دلیل پیش‌بینی اشتباه است.

هرگاه علامت ضربدر برای الگوریتم بهینه‌سازی ملخ درون مستطیل قرار گیرد، بدان معنا است که برچسب به‌خوبی پیشگویی شده‌است. همین امر برای مثلث نیز صادق است.

مقایسه روش پیشنهادی با سایر الگوریتم‌ها کاهش ویژگی توسط الگوریتم بهینه‌سازی ملخ که در بین دیگر روش‌ها بهترین صحت و نتیجه را دارد که روش آنالیز مؤلفه اصلی در بین سایر الگوریتم‌ها دارای بیشترین تشخیص می‌باشد. الگوریتم بهینه‌سازی ازدحام ذرات و ژنتیک به ترتیب در ردیف‌های بعدی قرارمی‌گیرند. شکل ۱۶



شکل ۱۶: نتایج حاصل از کاهش ویژگی توسط روش‌های مختلف

با توجه به شکل ۱۶ می‌توان‌دریافت که کاهش ویژگی توسط الگوریتم بهینه‌سازی ملخ که روش پیشنهادی بود در بین دیگر روش‌ها بهترین صحت و نتیجه را دارد، درصورتی‌که روش آنالیز مؤلفه اصلی در بین سایر الگوریتم‌ها دارای بیشترین تشخیص می‌باشد. الگوریتم بهینه‌سازی ازدحام ذرات و ژنتیک به ترتیب در ردیف‌های بعدی قرارمی‌گیرند.

جدول ۱۷: نتایج حاصل از روش‌های مختلف توسط مجموعه اول

روش‌ها	Accuracy

PCA	٪۵۷٫۱۴
GA	٪۸۰٫۰۰
PSO	٪۹۵
روش پیشنهادی	٪۸۸٫۰۰

جدول ۱۸: نتایج حاصل از روش‌های مختلف توسط مجموعه دوم

روش‌ها	Accuracy
PCA	٪۵۷٫۱۴
GA	٪۸۰٫۰۰
PSO	٪۹۵
روش پیشنهادی	٪۹۶٫۳۶

همان‌طور که از جدول ۱۸ واضح است، روش پیشنهادی در مجموعه داده اول دارای عملکرد بهتری نسبت به دیگر روش‌های فراابتکاری است.

۵. نتیجه‌گیری

مدیریت مهندسی نرم‌افزار شامل برنامه‌ریزی است. برنامه‌ریزی پروژه نرم‌افزاری شامل پیش‌بینی نرم‌افزار است و متداول‌ترین متغیرهای پیش‌بینی‌شده اندازه بوده‌است. روش پیشنهادی با ترکیب روش داده‌کاوی و طبقه‌بندی ریسک‌های نرم‌افزاری به‌دنبال بهبود نتیجه نهایی است. در بخش داده‌کاوی به کاهش ویژگی پرداخته‌ایم که از تکنیک الگوریتم بهینه‌سازی ملخ GOA استفاده‌شده‌است. با استفاده از الگوریتم فراابتکاری ویژگی‌ها اعتبارسنجی می‌شوند و با استفاده از روش ترکیبی در داده‌های با ابعاد بالا نتایج قابل‌قبول و بهتری در انتخاب ویژگی و کاهش بعد داشته‌است. لذا با مقایسه این روش با ترکیب خروجی دیگر روش‌های فراابتکاری همانند PSO، GA و کاهش ابعاد به کمک PCA پرداخته‌شد که در هر دو مجموعه داده روش پیشنهادی دارای صحت ۸۸ درصد و ۹۹,۳۶ درصد به ترتیب است.

References

[1] GouthamanP, “Prediction of Risk Percentage in Software Projects by Training Machine Learning Classifiers Computers & Electrical Engineering”, Volume 94, September 2021.

[3] RIZWAN M et al, “Analyses of Classifier’s Performance Measures Used in Software Fault Prediction Studies”. IEEE Access 2019, 7:82764–75.

[4] MattosDFM et al, “An aglie effective network functionvir tualization in infrastucture for the Internet of Things”.J Internet, volume 10, Article number:6 (2019).

[5] Dam HK et al, “Towards Effective AI-Powered Agile Project Management”, 2019 IEEE/ACM 41st International Conference on Software Engineering: New Ideas and Emerging Result (ICSE-NIER). Pages 41–44

[6] SH.Saremi, Mirjalali, A.Lewi (2017), “GrasshopperOptimis ationAlgorithm”: Theory and application, Advances in Engi neering Software, Volume 105, Pages 30-47,2017.

[7] N.Paltrinieri et al,”Learning about risk:Machine learning for risk assessment”,Safety Science,Vol 118,Pages 475-486,October 2019.

[8] L.Bai et al,”Project Portfolio Resource Risk Assessment considering Project Interdependency by the Fuzzy Bayesian Network”, Complexity, vol. 2020, Article ID 5410978,21 pages, 2020.

[9] L.Radu,”Effort Prediction in Agile Software Development with Bayesian,Networks”, Proceedings of the International Conference on Software Technologies (ICSOFT 2019). Pages 238–245.

[10] L. Pan et al, “An Improved CNN Model for Within-Project Software Delect Prediction”, Journals Applied Vol 9, Applied Sciences 9(10):2138,2019.

[11] Sai Mohan et al.”A Survey on Software Cost Estimation Techniques”,School of Electrical Engineering & Computer Science,University of North Dakota,Grand Forks,USA Vol.12 No.6, Pages 23 June 2019.7

[12] A. Puri et al,”Risk Management in Software Engineering Using Big Data”,International Conference on Intelligent Engineering and Management, 17-19 June 2020.

[13] M. Trzeciak et al,”Sustainable Risk Management in IT Enterprises,Advances in Sutainable Rick Management”, Jour nals Risks Vol 9,2021.

[14] Aru,O, et al: “OPTIMIZATION OF SOFTWARE RISK ASSESSMENT MODEL USING GENETICAL ALGORITHM”,117 Umudike Journal of Engineering and Technology(UJET);Vol.5, No 1,June 2019, pp.117 123;Michael Okpara University of Agriculture, Umudike,PrintISSN:2536-7404,Electronic ISSN: 2545-525 7;https://ujetmouau. net; DOI:https://doi.org/10.33922/j.ujet_ v5i1 13.

[15] M. Alanis et al,”Particle Swarm Optimization for Predicting the Development Effort of Software Projects” ,Mathematics2020,8,1819;doi:10.3390/math81018 19.

[16] Ms. Manju Vyas et al,”Effect of Dimensionality Reduction on Prediction Accuracy of Effort of Agile Projects Using Principal Component Analysis”, IOP Conf. Series: Materials Science and Engineering,2021.

[17] Juan Murillo-Morera, Christian Quesada-López,”A genetic algorithm based framework for software efforT prediction”, Journal of Software Engineering Research and Development 5, Article number: 4 (2017)

[18] Mariana Dayanara Alanis, “Particle Swarm Optimization for Predicting the Development Effort of Software Projects”, Mathematics 2020, 8(10).

[19] Pospieszny, P.Czarnacka-Chrobot, B.Kobylinski,A. “An effective approach for software project effort and duration estimation with machine learning algorithms”. J. Syst. Softw.2018, 137, 184–196

[20] Li, Z. Jing, X.Y.” Zhu, X.Progress on approaches to software defect prediction”.IET Softw. 2018, 12, 161–175.

پی‌نوشت

[1] deep neural network (DNN)

[2] Convolutional neural networks (CNN)

[3] within-project defect prediction (WPDP)