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

**Summer 2024, 5 (2), 23-31**
**DOI: 10.71856/impcs.2024.903599**

**Received: 6 Apr 2024**
**Accepted: 25 May 2024**

**مقاله پژوهشی**

# An Algorithm to Measure the Attention Rate of Mobile Application Users using App Reviews

**Mehrdad Razavi Dehkordi[1], Hamid Rastegari[2*], Akbar Nabiollahi Najafabadi[3], Taghi Javdani Gandomani[4]**

1. Department of Computer Engineering, Najafabad Branch, Islamic Azad University, Najafabad, Iran. mehrdad.razavi68@sco.iaun.ac.ir
2. Assistant Professor, Department of Computer Engineering, Najafabad Branch, Islamic Azad University, Najafabad, Iran. **Corresponding Author*, rastegari@iaun.ac.ir
3. Assistant Professor, Department of Computer Engineering, Najafabad Branch, Islamic Azad University, Najafabad, Iran. a.nabi@pco.iaun.ac.ir
4. Assistant Professor, Department of Computer Science, Shahrekord University, Shahrekord, Iran. javdani@sku.ac.ir

## Abstract

**Introduction:** Since the emergence of mobile apps, user reviews have been of great importance for app developers. as they contain users' sentiment, bugs and new requests, user feedback has been valuable to app developers. Due to the large number of reviews, analysing them is a difficult, time-consuming and error-prone task. So far, many works have been done in the field of classifying reviews but in none of them, it is not clear which category the users paid more attention to after categorization.

**Method:** In this article, an algorithm called UAR(User Attention Rate) is presented based on the date of reviews and their number to measure the level of attention of users to the category of bug report or feature request. To test the proposed algorithm, the reviews of WhatsApp, NextCloud, Mozilla Firefox and VLC Media Player apps were extracted by the web crawler for 30 days, and then the reviews were given to 4 software experts after classification. In the following, after analyzing the reviews by 4 experts, the level of attention of users to the reviews was calculated by them. Then, using the proposed method, amount of users' attention to reviews was measured.

**Results:** It was found that the proposed algorithm was able to have acceptable results in calculating the amount of users' attention to reviews compared to the values calculated by experts. So that the difference in the attention rate calculated by experts according to the proposed algorithm was 72.2%.

**Discussion:** In this article, an algorithm is presented to calculate the users' attention rate to the categories of feature requests and bug reports. By using the relevant algorithm, the development team will understand that the user wants to fix the bugs related to the application or add new features to the application.

**Keywords:** Mobile application, user attention rate, google play store analysis, user reviews.

سامانه‌های پردازشی و ارتباطی چندرسانه‌ای هوشمند
Intelligent Multimedia Processing and Communication Systems (IMPCS)
انجمن علمی تجارت الکترونیکی ایران
واحد زنجان

# ارائه الگوریتمی جهت اندازه‌گیری میزان توجه کاربران برنامه‌های موبایل با استفاده از نظرات مربوط به برنامه‌ها

تاریخ دریافت: ۱۸/ ۰۱/ ۱۴۰۳
تاریخ پذیرش:۰۵/ ۰۳/ ۱۴۰۳
دورهٔ پنجم، تابستان ۱۴۰۳
شمارهٔ دوم، صص: ۲۳-۳۱

**مهرداد رضوی‌دهکردی[۱]، حمید رستگاری[۲*]، اکبر نبی‌اللهی نجف‌آبادی[۳]**، تقی جاودانی‌گندمانی[۴]

۱. گروه مهندسی کامپیوتر، واحد نجف‌آباد، دانشگاه آزاد اسلامی، نجف‌آباد، ایران. mehrdad.razavi68@sco.iaun.ac.ir
۲. استادیار، گروه مهندسی کامپیوتر، واحد نجف‌آباد، دانشگاه آزاد اسلامی، نجف آباد، ایران. (نویسنده مسئول) rastegari@iaun.ac.ir
۳. استادیار، گروه مهندسی کامپیوتر، واحد نجف‌آباد، دانشگاه آزاد اسلامی، نجف‌آباد، ایران. a.nabi@pco.iaun.ac.ir
۴. استادیار، گروه علوم کامپیوتر، دانشگاه شهرکرد، شهرکرد، ایران. javdani@sku.ac.ir

**چکیده:** از زمان پیدایش برنامه‌های موبایل، نظرات کاربران برای توسعه‌دهندگان برنامه بسیار ارزشمند بوده‌است. به دلیل حجم بالای نظرات، تحلیل آن‌ها کار بسیار مشکلی است. تاکنون کارهای بسیاری در زمینه دسته‌بندی نظرات انجام‌شده‌است، اما در هیچ‌کدام از آن‌ها مشخص‌نشده‌است که پس از دسته‌بندی، توجه کاربران بیشتر به کدام دسته بوده‌است. در این مقاله الگوریتمی بانام UAR بر اساس تاریخ ثبت نظرات و تعداد آن‌ها برای اندازه‌گیری میزان توجه کاربران به دسته گزارش مشکل یا درخواست ویژگی ارائه‌شده‌است. برای آزمایش الگوریتم مورد نظر، نظرات برنامه‌های WhatsApp، NextCloud، Mozilla Firefoxو VLC Media Player توسط خزنده وب به مدت ۳۰ روز استخراج‌شده و نظرات پس از دسته‌بندی به ۴ متخصص نرم‌افزار داده‌شد. در ادامه پس از تحلیل نظرات توسط ۴ متخصص میزان توجه کاربران به نظرات توسط آن‌ها مورد محاسبه قرارگرفت. سپس با استفاده از روش پیشنهادی میزان توجه کاربران به نظرات اندازه‌گیری‌شد. در نهایت مشخص‌شد که روش پیشنهادی توانسته نتایج قابل‌قبولی در محاسبه میزان توجه کاربران به نظرات نسبت به مقادیر محاسبه‌شده توسط متخصصان داشته‌باشد طوری که تفاوت میزان توجه محاسبه‌شده توسط متخصصان با میزان توجه محاسبه‌شده توسط الگوریتم ارائه‌شده ۷۲/۲٪ بوده‌است

**واژه‌های کلیدی:** برنامه‌های موبایل، میزان توجه، آنالیز فروشگاه گوگل، میزان توجه کاربران.

## ۱. مقدمه

امروزه طبق آمارهای رسمی سایت IDC تا نیمه چهارم سال ۲۰۲۲ حدود ۳۰۰/۳ میلیون تلفن همراه هوشمند تولیدشده‌است که ۷۵/۸٪ از آن به گوشی‌های هوشمند با سیستم‌عامل اندروید تعلق‌دارد. سیستم‌عامل آندروید برای خود فروشگاهی با نام Google PlayStore دارد که تمامی توسعه‌دهندگان نرم‌افزار های تولیدشده خود را برروی آن قرارمی‌دهند[2][1]. برنامه‌های موجود در فروشگاه توسط کاربران بسیار دانلودشده و کاربران فروشگاه گوگل امکان ثبت نظر برای برنامه مورد نظر خود را دارند. مطالعات نشان‌داده‌است که نظرات کاربران حاوی اطلاعات مهمی شامل گزارش مشکل، درخواست ویژگی و تجربه کاربران در کار با برنامه است[5][4][3]. مطالعات پیشین نشان‌می‌دهد که نظرات ثبت‌شده توسط کاربران می توانند فرآیند توسعه نرم‌افزار را هدایت‌کرده و باعث بهبود نسخه‌های آینده نرم‌افزار شوند[7][6]. به‌علاوه نظرات شامل اطلاعات بسیار مهمی برای آنالیزکننده‌ها و طراحان برنامه نیز هستند[4][8][5]. به‌علاوه اولویت‌بندی و دسته‌بندی به‌موقع نظرات نیز باعث‌می‌شود تا تیم توسعه‌دهنده در نگهداری نرم‌افزار و به‌روزرسانی‌های به‌موقع موفق‌تر باشد[11][10][9].

در زمینه دسته‌بندی نظرات تاکنون کارهای بسیاری انجام شده‌است؛ اما در هیچ‌کدام از این روش‌ها یا مدل‌ها بیان‌نشده که پس از دسته‌بندی یا اولویت‌بندی نظرات کاربران، توجه کاربران بیشتر به دسته گزارش مشکل بوده‌است یا درخواست ویژگی. به‌بیان‌دیگر می‌توان‌گفت در هیچ‌کدام از روش‌ها، گزارش‌نشده که پس از اولویت‌بندی یا دسته‌بندی نظرات، کاربر برنامه از توسعه‌دهنده خواسته تا بیشتر مشکلات برنامه را رفع‌کند یا بیشتر درخواست اضافه‌کردن ویژگی‌های جدید را به توسعه‌دهنده داده‌است؛ به‌عبارت‌دیگر توجه کاربر برنامه بیشتر به سمت گزارش مشکلات بوده‌است یا درخواست ویژگی‌های جدید. باید توجه‌داشت که تاکنون هیچ کاری درزمینه اندازه‌گیری میزان توجه کاربران انجام‌نشده‌است. با استفاده از الگوریتم پیشنهادی تیم توسعه‌دهنده خواهدتوانست میزان توجه کاربر به دسته‌های نظرات را اندازه‌گیری‌کرده و در به‌روزرسانی بعدی سعی در پاسخ‌دادن به آن‌ها داشته‌باشد. اگر اشکالات گزارش‌شده توسط کاربران یا ویژگی‌های مورد نیاز آن‌ها به‌موقع پاسخ‌داده‌شوند، برنامه نه‌تنها موفق‌تر است بلکه نزد کاربر نیز دارای محبوبیت بیشتری خواهدبود[11][10].

هدف از این تحقیق ارائه الگوریتمی جهت محاسبه میزان توجه کاربران برنامه‌های موبایل به دسته‌های درخواست ویژگی یا گزارش مشکل با استفاده از تاریخ ثبت نظرات و تعداد آن‌ها است. برای مشاهده و اندازه‌گیری کارایی روش پیشنهادی با استفاده از خزنده وب، نظرات مربوط به چهار برنامه WhatsApp، Next Cloud، Mozilla Firefox و VLC Media Player به مدت ۳۰ روز استخراج‌شده و نظرات پس از دسته‌بندی به دسته‌های درخواست ویژگی و گزارش مشکل به ۴ متخصص در حوزه نرم‌افزار بوده‌اند، داده‌شد. در ادامه میزان توجه کاربران برای هرکدام از برنامه‌ها توسط متخصصین به‌طور میانگین اندازه‌گیری‌شده و با مقدار محاسبه‌شده توسط الگوریتم پیشنهادی مقایسه و در نهایت مشخص شده‌است که الگوریتم پیشنهادی نتایج قابل‌قبولی در محاسبه نرخ توجه کاربران به دسته‌های نظرات دارد.

ادامه این مقاله به صورت زیر سامان‌دهی شده‌است: در بخش دوم کارهای انجام‌شده در دسته‌بندی نظرات و نیازمندی‌ها بررسی‌شده و به چالش‌های آن‌ها پرداخته شده‌است، در بخش سوم الگوریتم پیشنهادی برای محاسبه میزان توجه کاربران به دسته‌های گزارش مشکل یا درخواست ویژگی آورده شده‌است، در بخش چهارم در مورد روش انجام آزمایش‌ها و معیارهای مقایسه صحبت‌شده، در بخش پنجم نتایج آزمایش‌ها عنوان‌شده و در نهایت در بخش ششم نتیجه‌گیری و پیشنهادهایی برای کارهای آینده آمده‌است.

## ۲. کارهای مرتبط و چالش‌های آن‌ها

در دسته‌بندی نظرات عنوان‌شده توسط کاربران برنامه‌های موبایل کارهای زیادی انجام‌شده‌است که از مهم‌ترین آن‌ها می‌توان به موارد زیر اشاره‌کرد:

چن و همکاران در سال ۲۰۱۴، ابزاری با نام AR-Miner برای تشخیص نقدهای دارای اطلاعات ضروری و غیرضروری برای تیم توسعه‌دهنده ارائه‌دادند، مدل ارائه‌شده برای آموزش، از نظرات مربوط به دو برنامه SwiftKey و Facebook و دو بازی Temple Run و Tapfish، با استفاده از الگوریتم EMNB و Top Modeling استفاده‌کرد و توانست با Hit rate با مقدار ۰/۷ نظرات سودمند را از نظرات غیر سودمند مجزا سازد[9].

مالج و نبیل در سال ۲۰۱۶، مطالعه‌ای را برای شناخت بهترین روش برای دسته‌بندی نظرات کاربران در فروشگاه Play بر اساس دسته‌های درخواست ویژگی، گزارش مشکل، تجربه کاربران و امتیازدهی انجام‌دادند. آن‌ها پایگاه داده‌ای از ۱۱۰۰ برنامه با ۱/۱ میلیون نظر ساختند و در نهایت تئوری بیز را به‌عنوان بهترین روش دسته‌بندی با precision با مقدار ۰/۸۲ معرفی‌کردند[12].

مقدم در سال ۲۰۱۵، با بررسی ۵۰۰۰۰ نظر از فروشگاه برنامه‌های ebay، توانست الگوهای موجود در نظرات کاربران را با استفاده از برچسب زدن گفتار شناسایی‌کند. سپس با استفاده از تحلیل احساسات و روش بردار پشتیبان و روش کاهش ویژگی LDA برای دسته‌بندی، نظرات کاربران را به دسته‌های درخواست ویژگی، گزارش اشکال یا گزارش کار با برنامه دسته‌بندی‌کرد. در نهایت مشخص شد که روش مبتنی بر الگوها توانسته در دسته‌بندی نظرات Precision با مقدار ۸۸% داشته‌باشد[13].

دجوهان و همکاران، ابزاری با نام SAFE برای استخراج ویژگی‌های نهفته در نظران کاربران در فروشگاه Play ارائه‌دادند. آن‌ها به روش دستی، ۱۸ الگوی POS و ۵ الگوی مربوط به جملات را از نظرات استخراج‌کردند. سپس برای آزمایش از نظرات مربوط به ۱۰

برنامه استفاده و مشخص شد ابزار ارائه شده می‌تواند ویژگی‌های موجود در نظرات کاربران را با Precision با مقدار ۰/۵۵ و recall با مقدار ۰/۴۳ تشخیص‌دهد[14].

اسکالابرینو و همکاران در سال ۲۰۱۹، از روش‌های یادگیری ماشین استفاده‌کرده و ابزاری با عنوان CLAP را برای دسته‌بندی، خو شه‌بندی نظرات و اولویت‌بندی خو شه‌های که باید در آینده تو سط تیم تو سعه‌دهنده پیاده سازی شوند، ارائه‌دادند. د سته‌های آن‌ها شامل گزارش مشکل عملیاتی، پیشنهاد برای ویژگی جدید و نیازمندی‌های غیرعملکردی بود. در نهایت آن‌ها با استفاده از الگوریتم‌های درخت تصادفی و DBScan و استفاده از پایگاه داده ابزار Ar-Miner[9] به precision با مقدار ۰/۸۷ در دسته‌بندی نظرات دست‌یافتند[15].

مارتنز و همکار در سال ۲۰۱۹، با تحقیق بر روی نظرات جعلی در فروشگاه Apple App Store و مشورت با ۴۳ فرد تولیدکننده نظرات جعلی تواستند پایگاه داده‌ای شامل ۶۰۰۰۰ نظر جعلی و ۶۲ میلیون نظر تولیدکنند. سپس پایگاه داده جمع‌آوری‌شده را به ۷ ابزار دسته‌بندی دادند و درنهایت مشخص‌شد که ابزار درخت تصادفی توانسته با ۹۸٪ مقدار Recall نظرات جعلی را از نظرات واقعی دسته‌بندی کند[16].

اسلم و همکاران در سال ۲۰۲۰ با استفاده از پایگاه داده ارائه شده در مطالعه مالج و همکاران در سال ۲۰۱۶[12] و استخراج ۴۴۰۰ نظر از نظرات این پایگاه داده، سیستمی را با استفاده از شبکه‌های عصبی کونولوشنی (CNN) و یادگیری عمیق برای دسته‌بندی نظرات ارائه‌دادند. سیستم پیشنهادی توانست F-Measure با مقدار ۹۴/۷٪ در دسته‌بندی نظرات داشته‌باشد[17].

کیائو و وانگ در سال ۲۰۲۰، مدلی را بر پایه یادگیری عمیق با معماری LSTM برای دسته‌بندی نظرات کاربران در دسته بازی‌های موجود در فرو شگاه Apple App Store ارائه‌دادند. مدل آن‌ها با برر سی 261,51518, نظر برای 4,602 بازی ساخته شد. سپس از ۲۶ دانشجوی کار شنا سی برای برچسب زدن د ستی نظرات به سه د سته درخوا ست ویژگی، گزارش مشکل و سایر استفاده و درنهایت مشخص شد که ابزار پیشنهادی با استفاده از یادگیری عمیق دارای F-Measure با مقدار ۰/۷۶۶، دارای Precision با مقدار ۰/۷۶۹ و دارای Recall با مقدار ۰/۷۶۸ در دسته‌بندی نظرات بوده‌است[18].

در جدول ۱ مهم‌ترین کار ها در دسته‌بندی نظرات با هم مقایسه‌شده‌اند.

**جدول ۱: مهمترین کارها در زمینه دسته بندی نظرات کاربران**

| ارائه‌دهنده / سال نشر | دستاورد | روش مورد استفاده | نتایج و معیار ارزیابی |
|---|---|---|---|
| ۲۰۱۴ [9] چن و همکاران | ار AR-Miner | EMNB و Top Modeling و LDA | F-Mesaure با مقدار ۰/۷ در دسته‌بندی نظرات سودمند از غیر سودمند |
| ۲۰۱۵ [13] مقدم | شناسایی الگوهای موجود در نظرات کاربران | POS, تحلیل احساسات و SVM | Precision با مقدار ۰/۸۸ در شناسایی نظرات کاربران |
| ۲۰۱۶ [14] مالج و همکار | شناسایی بهترین الگوریتم در دسته بندی نظرات کاربران | ات | تئوری بیز با precision با مقدار ۰/۸۲ بهترین روش در دسته‌بندی نظرات است |
| ۲۰۱۷ [15] جوهان و همکاران | ار SAFE | Pos Tag & Pattern Recognition | precision با میزان ۰/۵۵ و recall با میزان ۰/۴۳ در شناسایی ویژگی‌های مورد درخواست کاربران |
| ۲۰۱۹ [15] اسکالابرینو و همکاران | ار CLAP | Random Forest و DBSCAN | Precision با مقدار ۰/۸۷ در دسته‌بندی و اولویت‌بندی نظرات کاربران |
| ۲۰۲۰ [17] اسلم و همکاران | اه Play | NLP و | F-Measure با میزان ۹۴/۷ در دسته‌بندی نظرات کاربران |
| ۲۰۲۰ [18] کیائو و همکار | اه Apple App Store | CNN و LSTM | ار ۰/۷۶۶، دار۰/۷۶۹ و رات |

طبق جدول ۱ کارهای انجام‌شده دیگران دارای چالش‌ها و مشکلات زیر است:

- در ابزار ارائه‌شده امکان استخراج نظرات به‌روز برای برنامه، دسته‌بندی، خلاصه‌سازی، تعیین قطبیت و اولویت‌بندی آن‌ها در هرلحظه و با اراده تیم توسعه‌دهنده وجودندارد؛ بلکه پایگاه داده‌ای ساخته‌شده سیستم با آن آموزش‌دیده و عمل دسته‌بندی، خوشه‌بندی و اولویت‌بندی روی آن پایگاه داده آزمایش‌شده‌است
- برای یادگیری سیستم دسته‌بندی آن‌ها از روش‌های قدیمی یادگیری ماشین استفاده‌شده‌است.
- در دسته‌بندی و اولویت‌بندی از ویژگی‌هایی استفاده‌شده که اکنون دیگر توسط فروشگاه گوگل ارائه‌نمی‌شوند و اکنون امکان استفاده از این ویژگی‌های نمی‌باشد.

- برای ساخت پایگاه داده برای آموزش مدل ارائه‌شده، ویژگی‌ها به‌صورت دستی توسط عوامل انسانی از نظرات استخراج‌شده‌اند؛ اما امروزه با استفاده از یادگیری عمیق دیگر عمل استخراج ویژگی کاربرد آن‌چنانی نخواهدداشت.

به‌علاوه با توجه به مشکلات مورد اشاره موجود در روش‌های دیگران در هیچ‌کدام از ابزارها مشخص نشده‌است که پس از دسته‌بندی یا اولویت‌بندی نظرات، کاربران برنامه‌های موبایل از تیم توسعه‌دهنده بیشتر درخواست اضافه‌کردن ویژگی‌های جدید را داشته یا بیشتر مشکلات مربوط به برنامه را گزارش‌داده و خواستار رفع آن‌ها بوده‌اند.

با توجه به چالش‌های مذکور، ارائه روش یا الگوریتمی برای محاسبه میزان درخواست کاربران یا میزان توجه آن‌ها مورد نیاز است و پرسش زیر در تحقیق مطرح‌می‌شود:

آیا روش پیشنهادی می‌تواند به‌خوبی روش‌های دستی (روش بررسی نظرات توسط متخصصین) میزان توجه کاربران به دسته‌های گزارش مشکل یا درخواست ویژگی را محاسبه‌کند؟

## ۳. روش محاسبه میزان توجه کاربران

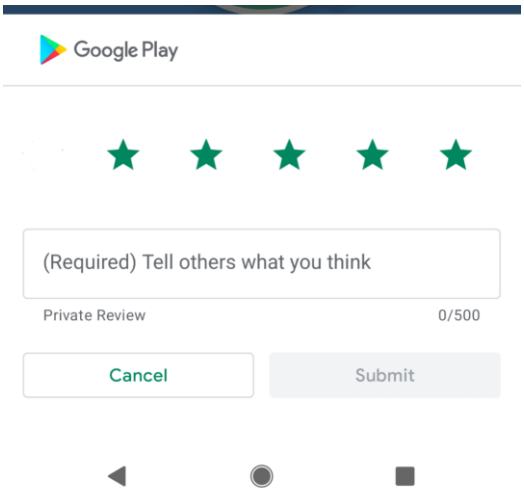
روزانه نظرات بسیاری برای برنامه‌ها ثبت‌می‌شود. هدف ما محاسبه میزان توجه کاربران به نظرات است. در این تحقیق بر پایه ویژگی‌هایی که در زیر بیان‌می‌شود الگوریتمی برای محاسبه نرخ توجه کاربران به دسته‌های گزارش مشکل و درخواست ویژگی با عنوان *UAR (User Attention Rate)* ارائه‌شده‌است. باید توجه‌داشت که نظرات برنامه به‌وسیله خزنده‌های تحت‌وب پیاده‌سازی شده با استفاده از زبان برنامه‌سازی پایتون با دادن نام Package برنامه در فروشگاه Play قابل-استخراج هستند. برای محاسبه نرخ توجه از ویژگی‌های تعداد نظرات و تاریخ ثبت نظرات استفاده‌شده‌است. در ادامه به بررسی هر کدام از ویژگی‌ها به‌طور جداگانه پرداخته‌شده‌است.

### ۱،۳. تاریخ ثبت نظرات

هنگامی‌که کاربر در فروشگاه Play اقدام به ثبت نظر کند، گزینه‌ای برای امتیازدهی نمایش داده‌می‌شود که می‌تواند امتیازی بین ۱ تا ۵ برحسب رضایت به برنامه بدهد. به‌علاوه در زیر فیلد مربوط به امتیاز، محلی برای ثبت نظر تعبیه‌شده‌است (شکل ۱).

**شکل ۱: ثبت نظر و امتیاز مربوط به برنامه در فروشگاه Play**

مطابق شـــکل ۱ ، هنگام کلیک بر روی گزینه Write a review پنجره‌ای برای نوشتن نظر به کاربر نشان‌داده‌می‌شود. در شکل ۲ نحوه امتیازدهی و ثبت نظر کاربر آمده‌است.

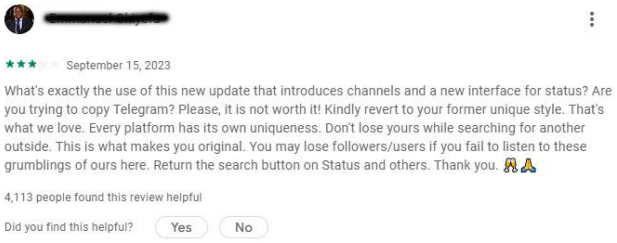



**شکل ۲: نحوه ثبت امتیاز و نظر برای برنامه در فروشگاه Play**

هنگام مراجعه به صفحه مربوط به برنامه مشاهده‌می‌شود که تعدادی از نظرات تاریخ ثبت قدیمی‌تری نسبت به نظرات جدید دارند اما در بالای نظرات جدید نمایش‌داده‌می‌شوند. به این نظرات، بسیار مرتبط ( Most Relevant) گفته‌می‌شود. فروشگاه گوگل با استفاده از فاکتورهایی مانند الف- طول نظر ثبت‌شده، ب- اشاره نظر ثبت‌شده به یک ویژگی یا قسمت خاص برنامه، ج- نظر ثبت‌شده توسط منتقد خاص یا مورد تأیید Google، د- استفاده از کلمات خاص، ه- استفاده از عکس در ثبت نظرات، و-نظرات دارای احساس منفی، نظرات بسیار مرتبط را از دیگر نظرات متمایزکرده و به کاربران نشان‌می‌دهد[20][19]. ممکن است تعداد نظرات بسیار مرتبط نسبت به سایر نظرات کمتر باشند اما در بسیاری از موارد این نظرات دغدغه‌های دیگر کاربران استفاده‌کننده از برنامه هستند.

هنگام ثبت نظر تاریخ آن نیز توسط فروشگاه ثبت‌می‌شود. زمانی که با استفاده از خزنده وب نظرات مربوط به برنامه خاص استخراج‌می‌شود، تاریخ ثبت نظر و نوع نظر شامل جدید یا بسیار مرتبط بودن آن نیز توسط خزنده استخراج‌می‌شود.

در نتیجه کاربران هنگام ثبت نظر که شامل گزارش مشکل یا درخواست ویژگی یا شرح کارکرد در کار با برنامه است، در صورت مشاهده دغدغه خود که توسط کاربر دیگر مطرح‌شده، دیگر آن را ثبت نکرده و برای تصدیق و تأیید نظر دیگران از دکمه ThumbsUp با علامت 👍 استفاده‌می‌کنند. در بسیاری از موارد نیز ممکن است با پیغام Did you find this Helpful با دکمه‌های Yes و No مواجه‌شوند. در نتیجه تعدادی از نظرات در مورد برنامه دارای محبوبیت بسیار از طرف کاربران وجوددارند.




**شکل ۳: نظرات ثبت شده در فروشگاه Play برای نرم‌افزار WhatsApp**

همان‌طورکه در شکل ۳ مشاهده‌می‌شود در ابتدا نام ثبت‌کننده نظر به همراه عکس پروفایل، در زیر آن تاریخ ثبت نظر مشاهده‌می‌شود. تاریخ ثبت نظر مربوط به چند روز پیش است اما به‌دلیل Thumbsup بالا (4113 People found this review helpful) نظر در بالای دیگر نظرات نشان داده‌شده‌است.

برای محاسبه تاریخ ثبت نظرات، پس‌ازآن که نظرات در روز با استفاده از خزنده استخراج‌شدند، تاریخ ثبت اولین نظر با تاریخ ثبت آخرین نظر، مقایسه‌شده و فاصله آن‌ها برحسب روز طبق رابطه ۱ به‌دست‌می‌آید.

$$Dates = LastDate - FirstDate \qquad (1)$$

در رابطه ۱، Dates تفاوت تاریخ ثبت اولین نظر با آخرین نظر است. درحقیقت، می‌توان‌گفت که با استفاده از رابطه بالا می‌خواهیم محاسبه‌کنیم که نظرات موجود در این دسته‌بندی (دسته‌بندی درخواست ویژگی یا گزارش مشکل) در چند روز توسط کاربران ثبت شده‌اند زیرا هرچه این تعداد کوچک‌تر باشد نشان‌دهنده میزان اهمیت کاربر به دسته‌بندی مربوطه است.

### ۲،۳. تعداد نظرات

برای محاسبه تعداد نظرات موجود در هر دسته‌بندی، هنگام استفاده از خزنده جهت استخراج نظرات و پس از دسته‌بندی آن‌ها توسط یکی ابزارها مورد اشاره در قسمت ۲، تعداد نظرات را با $|R|$ نشان‌می‌دهیم که *بیانگر* تعداد نظرات ثبت شده توسط کاربر در آن دسته‌بندی خاص می‌باشد.

### ۳،۳. الگوریتم پیشنهادی محاسبه میزان توجه کاربران (UAR)

برای محاسبه میزان توجه کاربران به دسته‌بندی‌های انجام شده توسط ابزار دسته‌بندی، از رابطه ۲ استفاده‌می‌شود.

$$AR_r \leftarrow \frac{|R|}{Dates} \qquad (2)$$

در رابطه ۲ $|R|$ نشان‌دهنده تعداد نظرات، $Dates$ برابر تعداد روزی است که در طی آن نظرات توسط کاربران ثبت‌شده‌اند و $AR_r$ برابر میزان یا نرخ توجه کاربران به دسته‌بندی مورد نظر است.

در ادامه از رابطه ۱ و ۲ برای ارائه الگوریتم محاسبه میزان توجه استفاده‌خواهدشد.

| Algorithm : Calculate Attention Rate |
|---|
| Input: Sets of Reviews with their Metadata (R)<br>Output: $AR_r$ (Attention Rate R) |
| 1. $R_d \leftarrow Sort\ by\ Date\ Descending(r)$<br>2. $Calculate\ Dates\ from\ R_d$<br>3. $AR_r \leftarrow \frac{\lvert R\rvert}{Dates}$<br>4. $Return\ AR_r$<br>5. $End$ |
| |

**الگوریتم1 : الگوریتم محاسبه نرخ توجه به دسته خاص از نظرات**

همان‌طورکه در الگوریتم ۱ مشاهده‌می‌شود، ورودی الگوریتم دسته نظرات به‌صورت خام (استخراج‌شده توسط خزنده) است. در مرحله ۱ نظرات به‌صورت نزولی بر اساس تاریخ مرتب‌می‌شوند تا به‌راحتی بتوان تفاوت تاریخ ثبت اولین نظر تا آخرین نظر دارای ThumbsUp یا محبوبیت بالا را مشخص‌کرد. در مرحله ۲ میزان تفاوت تاریخ محاسبه‌شده، در مرحله ۳ با استفاده از رابطه ۲ نرخ توجه کاربر به دسته نظرات محاسبه‌می‌شود. سپس در مرحله ۴ نرخ توجه به هر دو دسته‌بندی گزارش مشکل یا درخواست ویژگی به‌عنوان خروجی منتشرمی‌شود. نرخ توجه عددی مثبت با مقدار حداقل صفر و به‌صورت اعشاری می‌باشد.

در ادامه می‌توان برای دسته‌بندی شامل نظراتی که توجه کاربر به آن‌ها بیشتر است، اولویت‌بندی را انجام‌داد. با استفاده از این الگوریتم وقت تیم توسعه‌دهنده کمتر تلف‌شده و تیم توسعه‌دهنده مستقیماً به سراغ اولویت‌بندی نظراتی خواهدرفت که بیشتر مورد توجه کاربر هستند.

## ۴. ارزیابی تجربی

در این قسمت به توضیح در مورد محیط انجام آزمایش‌ها، نحوه انجام آن‌ها، معیارهای مقایسه و نتایج آزمایش پرداخته‌می‌شود.

### ۱،۴. محیط آزمایش

برای آموزش مدل از زبان برنامه‌نویسی پایتون نسخه 3.10.11 و محیط برنامه‌نویسی Visual Studio Code نسخه 1.78.2 استفاده‌شده‌است. دلیل استفاده از این محیط راحتی Debug و هم‌شکل بودن با Microsoft Visual Studio است.

برای ارزیابی نتایج از کامپیوتر با پردازنده Intel Xeon E5-2650 v4، 16 گیگابایت Ram از نوع DDR4، بدون کارت گرافیک و ویندوز 10 استفاده‌شده‌است.

### ۲،۴. برنامه ها و نظرات مورد استفاده

برای آزمایش الگوریتم پیشنهادی در محاسبه میزان توجه کاربران، از تاریخ October 3,2023 تا تاریخ November 2,2023 نظرات بسیار

مرتبط* مربوط به برنامه‌های WhatsApp، Firefox،NextCloud و VLC Media Player توسط خزنده وب طی ۳۰ روز استخراج‌شد. برای مجموع نظرات استخراج‌شده، ویژگی‌های نام ثبت‌کننده نظر، عکس مربوط به او در صورت وجود، نام کاربری وی، امتیاز او به برنامه (عدد بین ۱ تا ۵)، متن نظر، ThumbsUp مربوط به آن، نسخه برنامه، تاریخ ثبت نظر و نوع نظر (Most Relevant یا New) موجود است. برای هر برنامه نظرات با ویژگی‌های یادشده استخراج‌شد.

**جدول ۲: تعداد نظرات استخراج شده برای هر برنامه طی ۳۰ روز**

| برنامه | تعداد نظرات در ۳۰ روز |
|---|---|
| WhatsApp | ۲۹۱۰ |
| NextCloud | ۹۸۱ |
| Mozilla Firefox | ۱۸۱۵ |
| VLC Media Player | ۲۸۶۵ |

طبق جدول ۲، تعداد نظراتی که طی ۳۰ روز برای هریک از برنامه‌ها جمع‌آوری‌شده‌اند، متفاوت است؛ به این دلیل که هر برنامه ممکن است دارای میزان محبوبیت بالا یا محبوبیت پایین با توجه به کاربران آن باشد.

به‌علاوه باید درنظرداشت که در مدت ۳۰ روز جمع‌آوری نظرات، نظرات تکراری حذف‌شده و تنها نظرات متمایز و غیرتکراری به همراه ویژگی‌ها ثبت شده و به کارشناس نرم‌افزار داده‌شده‌اند.

### 3.4. نتایج ارزیابی

در ادامه برای پاسخ به پرسش تحقیق، برای هر برنامه مجموع نظرات استخراج‌شده به ابزار دسته‌بندی اسلم و همکاران [17] -به‌دلیل داشتن بالاترین دقت- داده‌شد.

پس از دسته‌بندی نظرات با توجه به روش ارائه‌شده در [21]، از ۴ متخصص حوزه نرم‌افزار دارای حداقل ۳ سال سابقه کاری استفاده‌شد. متخصصان سوابق بسیاری در پروژه‌های تجاری داشته و به‌عنوان افراد مجرب فنی شناخته‌می‌شوند.

نظرات به متخصصین برای تحلیل داده شد و در مدت زمان ۷ روز هر ۴ متخصص جداگانه نظرات را بررسی و مشخص‌کردند که توجه کاربر چند درصد به گزارش مشکل و چند درصد به درخواست ویژگی در هر نرم‌افزار بوده‌است. میانگین نتایج بررسی در جدول ۳ آمده‌است.

**جدول ۳ : نتایج محاسبه میانگین درصد توجه کاربران به هرکدام از دسته‌ها**

| نرم‌افزار | میانگین درصد توجه کاربران استخراج شده توسط ۴ متخصص |
|---|---|
| WhatsApp | ۷۸/٪۶ گزارش مشکل<br>۲۱/٪۴ درخواست ویژگی |
| NextCloud | ۸۶/٪۹ درخواست ویژگی<br>۱۳/٪۱ گزارش مشکل |
| Mozilla Firefox | ۹۳/٪۴ گزارش مشکل |
| | ۶/٪۶ درخواست ویژگی |
| VLC Media Player | ۹۶/٪۹ گزارش مشکل<br>۳/٪۱ درخواست ویژگی |

در جدول ۳ میانگین نتایج درصد توجه کاربران به دسته‌های درخواست مشکل یا گزارش مشکل که توسط متخصصان اندازه‌گیری شده، آمده‌است. با توجه به اینکه امکان اختلاف نظر بین متخصصان اختلاف‌نظر نیز وجود داشت، در مجموع رأی اکثریت مورد قبول واقع‌شد.

در ادامه برای بررسی توانایی روش پیشنهادی UAR در محاسبه میزان توجه کاربران به دسته‌های نظرات، در ابتدا برای هر دسته‌بندی با استفاده از الگوریتم ۱ ارائه‌شده و سپس نرخ یا میزان توجه کاربران به دسته‌های نظرات محاسبه‌شده‌است و نتایج در جدول ۴ آورده‌شده‌است.

**جدول ۴ :محاسبه میزان یا نرخ توجه کاربران به دسته‌های نظرات محاسبه‌شده توسط االگوریتم UAR**

| نرم‌افزار | نرخ توجه محاسبه شده توسط الگوریتم پیشنهادی | درصد توجه کاربران محاسبه شده توسط الگوریتم پیشنهادی UAR | تفاوت نرخ توجه محاسبه شده توسط متخصصین و الگوریتم پیشنهادی |
|---|---|---|---|
| WhatsApp | ۷۶/۲۳ گزارش مشکل<br>۲۰/۷۶ درخواست ویژگی | ۷۲/٪۷۶ گزارش مشکل<br>۲۷/٪۲۴ درخواست ویژگی | ٪ ۵/۸۴ گزارش مشکل<br>٪ ۵/۸۴ درخواست ویژگی |
| NextCloud | ۲۸/۴ درخواست ویژگی<br>۴/۳ گزارش مشکل | ۸۴درخواست /٪۸۵ ویژگی<br>۱۵/٪۱۵ گزارش مشکل | ٪ ۲/۰۵ درخواست ویژگی<br>٪ ۲/۰۵ گزارش مشکل |
| Mozilla Firefox | ۵۶/۵ گزارش مشکل<br>۴ درخواست ویژگی | ۹۲/٪۹ گزارش مشکل<br>۷/٪۰۱ درخواست ویژگی | ٪ ۰/۵گزارش مشکل<br>٪ ۰/۴۱ درخواست ویژگی |
| VLC Media Player | ۹۲/۵ گزارش مشکل<br>۵/۲۳ درخواست ویژگی | ۹۴/٪۳۴ گزارش مشکل<br>۵/٪۶۶ درخواست ویژگی | ٪ ۲/۵۶ گزارش مشکل<br>٪ ۲/۵۶ درخواست ویژگی |

همان‌طور که در جدول ۴ مشاهده‌می‌شود، میزان توجه نظرات با توجه تعداد نظرات موجود در دسته‌بندی تقسیم بر تعداد روزی که نظرات در آن وارد شده‌است، مشاهده‌می‌شود. سپس در ستون مقابل میزان توجه به درصد برای فهم بهتر تبدیل‌شده‌است. به‌علاوه برای فهم بهتر میزان تفاوت نرخ توجه محاسبه‌شده توسط الگوریتم و نرخ توجه محاسبه‌شده نیز در کنار آن‌ها آمده‌است.

- الگوریتم UAR پیشنهادی در مقایسه با روش دستی محاسبه میزان توجه کاربران در دسته گزارش مشکل توانسته به‌طور میانگین با اختلاف **۲/۷۳** ٪ میزان توجه کاربران را محاسبه‌کند.
- به‌علاوه الگوریتم UAR پیشنهادی توانسته در مقایسه با روش دستی، به‌طور میانگین با اختلاف **۲/۷۱**٪ میزان توجه نظرات کاربران به دسته درخواست ویژگی را محاسبه‌کند.
- باید توجه‌داشت که تاکنون هیچ روش یا ابزار یا مدلی برای محاسبه میزان توجه کاربران و گرایش آن‌ها به دسته‌های خاص نظرات ارائه نشده‌است. درنتیجه مقایسه الگوریتم ارائه‌شده با دیگر روش‌ها اشتباه است.

برای مقایسه روش پیشنهادی با روش دستی تنها راه، استفاده از ۴ کارشناس خبره در زمینه برنامه‌نویسی نرم‌افزار بود. زیرا هیچ ابزار یا روشی برای مقایسه با روش پیشنهادی وجودنداشت.

## ۵. نتیجه‌گیری و کارهای آینده

روزانه نظرات بسیاری توسط کاربران برای برنامه‌های موبایل ثبت‌می‌شود. با توجه به آن‌که حجم نظرات ثبت‌شده بسیار بالا بوده و تحلیل آن‌ها برای تیم توسعه‌دهنده امری دشوار است، ابزار یا روشی مورد نیاز است تا میزان یا نرخ توجه کاربران به نظرات برنامه مورد نظر با توجه به دسته‌های گزارش مشکل یا درخواست ویژگی بررسی‌کند. در صورت داشتن چنین ابزاری و اندازه‌گیری میزان توجه کاربران، تیم توسعه‌دهنده مستقیماً و بدون اتلاف وقت به دسته موردتوجه کاربر مراجعه‌کرده و علاوه‌بر عدم اتلاف زمان، به‌راحتی ویژگی‌های مورد درخواست کاربر یا مشکلات گزارش‌شده توسط او را در نسخه‌های بعدی پیاده‌سازی‌کرده یا پاسخ‌می‌دهد. بدین‌ترتیب، نرم‌افزار در بین رقبا محبوبیت خود را حفظ-کرده و همیشه به‌روز خواهدبود.

در این تحقیق الگوریتمی با عنوان UAR (User Attention Rate) برای محاسبه میزان توجه کاربران به دسته‌های گزارش مشکل یا درخواست ویژگی ارائه‌شده‌است. برای آزمایش الگوریتم یا روش پیشنهادی از ۴ متخصص کمک گرفته‌شد و از آن‌ها خواسته‌شد تا میزان توجه کاربران به نظرات مربوط به ۴ برنامه از فروشگاه Play را بررسی-کنند. درنهایت با مقایسه میزان توجه تعیین‌شده توسط متخصصین و میزان یا نرخ توجه محاسبه‌شده توسط روش پیشنهادی، مشخص‌شد که الگوریتم پیشنهادی توانسته با اختلاف بسیار کم میزان توجه کاربران به دسته‌های گزارش مشکل یا درخواست ویژگی یا تشخیص‌دهد.

با استفاده از ویژگی‌های بیشتر، می توان در کارهای آینده، میزان توجه کاربران به دسته‌بندی‌های بیشتر را بررسی‌کرد.

## مراجع


[1] M. R. Dehkordi, H. Seifzadeh, G. Beydoun, and M. H. Nadimi-Shahraki, “Success prediction of android applications in a novel repository using neural networks,” *Complex Intell. Syst.*, vol. 6, no. 3, pp. 573–590, 2020, doi: 10.1007/s40747-020-00154-3.

[2] W. Martin, F. Sarro, Y. Jia, Y. Zhang, and M. Harman, “A survey of app store analysis for software engineering,” *IEEE Trans. Softw. Eng.*, vol. 43, no. 9, pp. 817–847, 2017, doi: 10.1109/TSE.2016.2630689.

[3] E. Guzman and W. Maalej, “How Do Users Like This Feature? A Fine Grained Sentiment Analysis of App Reviews,” in *2014 IEEE 22nd International Requirements Engineering Conference (RE)*, Aug. 2014, pp. 153–162, doi: 10.1109/RE.2014.6912257.

[4] D. Pagano and W. Maalej, “User feedback in the appstore: An empirical study,” in *2013 21st IEEE International Requirements Engineering Conference (RE)*, Jul. 2013, pp. 125–134, doi: 10.1109/RE.2013.6636712.

[5] L. V. G. Carreno and K. Winbladh, “Analysis of user comments: An approach for software requirements evolution,” *Proc. - Int. Conf. Softw. Eng.*, pp. 582–591, 2013, doi: 10.1109/ICSE.2013.6606604.

[6] W. Maalej and D. Pagano, “On the socialness of software,” *Proc. - IEEE 9th Int. Conf. Dependable, Auton. Secur. Comput. DASC 2011*, pp. 864–871, 2011, doi: 10.1109/DASC.2011.146.

[7] N. Seyff, F. Graf, and N. Maiden, “Using mobile RE tools to give end-users their own voice,” *Proc. 2010 18th IEEE Int. Requir. Eng. Conf. RE2010*, pp. 37–46, 2010, doi: 10.1109/RE.2010.15.

[8] A. Al-Subaihin *et al.*, “App store mining and analysis,” in *Proceedings of the 3rd International Workshop on Software Development Lifecycle for Mobile*, Aug. 2015, pp. 1–2, doi: 10.1145/2804345.2804346.

[9] N. Chen, J. Lin, S. C. H. Hoi, X. Xiao, and B. Zhang, “AR-miner: Mining informative reviews for developers from mobile app marketplace,” in *Proceedings - International Conference on Software Engineering*, May 2014, no. 1, pp. 767–778, doi: 10.1145/2568225.2568263.

[10] E. C. Groen, J. Doerr, and S. Adam, “Towards Crowd-Based Requirements Engineering A Research Preview,” in *Requirements Engineering: Foundation for Software Quality*, 2015, pp. 247–253.

[11] S. A. Licorish, B. T. R. Savarimuthu, and S. Keertipati, “Attributes that predict which features to fix: Lessons for app store mining,” *ACM Int. Conf. Proceeding Ser.*, vol. Part F1286, pp. 108–117, 2017, doi: 10.1145/3084226.3084246.

[12] W. Maalej, Z. Kurtanović, H. Nabil, and C. Stanik, “On the automatic classification of app reviews,” *Requir. Eng.*, vol. 21, no. 3, pp. 311–331, Sep. 2016, doi: 10.1007/s00766-016-0251-9.

[13] S. Moghaddam, “Beyond Sentiment Analysis: Mining Defects and Improvements from Customer Feedback,” 2015, pp. 400–410.

[14] T. Johann, C. Stanik, A. M. B. Alizadeh, and W. Maalej, “SAFE: A Simple Approach for Feature Extraction from App Descriptions and App Reviews,” in *Proceedings - 2017 IEEE 25th International Requirements Engineering Conference, RE 2017*, Sep. 2017, pp. 21–30, doi: 10.1109/RE.2017.71.

[15] S. Scalabrino, G. Bavota, B. Russo, M. Di Penta, and R. Oliveto, “Listening to the Crowd for the Release Planning of Mobile Apps,” *IEEE Trans. Softw. Eng.*, vol. 45, no. 1, pp. 68–86, 2019, doi: 10.1109/TSE.2017.2759112.

[16] D. Martens and W. Maalej, “Towards understanding and detecting fake reviews in app stores,” *Empir. Softw. Eng.*, vol. 24, no. 6, pp. 3316–3355, Dec. 2019,

doi: 10.1007/s10664-019-09706-9.
[17] N. Aslam, W. Y. Ramay, K. Xia, and N. Sarwar, “Convolutional neural network based classification of app reviews,” *IEEE Access*, vol. 8. Institute of Electrical and Electronics Engineers Inc., pp. 185619–185628, 2020, doi: 10.1109/ACCESS.2020.3029634.
[18] Z. Qiao, A. Wang, A. Abrahams, and W. Fan, “Deep Learning-Based User Feedback Classification in Mobile App Reviews,” 2020. [Online]. Available: https://aisel.aisnet.org/sigdsa2020.
[19] “What Criteria are Used by Google to Rank Reviews Most Relevant? | LaPraim Insights.” https://lapraim.com/insights/criteria-for-relevant-google-reviews (accessed Sep. 18, 2023).
[20] “How Google Decides What Reviews Are Most Relevant.” https://www.widewail.com/blog/how-google-decides-what-reviews-are-most-relevant (accessed Sep. 18, 2023).
[21] Z. Wan, X. Xia, D. Lo, and G. C. Murphy, “How does machine learning change software development practices?,” *IEEE Trans. Softw. Eng.*, vol. 47, no. 9, pp. 1857–1871, 2021, doi: 10.1109/TSE.2019.2937083.