A Proposed Keyword-Based Feature Extraction Approach for Labeling and Classifying Egyptian Mobile Apps Arabic Slang User Requirements Reviews

Rabab Emad Saudy[1,2](✉), Alaa El Din El-Ghazaly[2], Eman S. Nasr[3], and Mervat H. Gheith[1]

[1] Faculty of Graduated Studies for Statistical Research, Cairo University, Giza, Egypt
rababsaudy107@gmail.com
[2] Sadat Academy for Management Sciences, Cairo, Egypt
[3] Cairo, Egypt

Abstract. Mobile applications (apps) review feature is supplied by most mobile apps platforms, which authorize users to evaluate, comment, and rate apps after utilizing it. User reviews are identified as an oriental source to enhance mobile applications (apps) and raise the importance for users. With the acute rise in the quantity of reviews, how to functionally and efficiently analyze and mining the user reviews and recognize serious user requirements from them to enhance the mobile apps. In this paper, we suggest an automatic approach for identifying and classifying requirements into Functional Requirements (FR), Non-Functional Requirements (NFR) and Sentimental Requirements (SR) from Egyptian Mobile Apps Arabic Slang Reviews (MASR), utilizing a group of techniques Term Frequency – Inverse Document Frequency (TF-IDF), Bag of Words (BOW) and Natural Language Processing (NLP) techniques with keywords selection. We suggest applying Classifier Chains (CC) approach to convert classifying multi-labeled data problem into one or more problems of single labeling, and utilizing the hybrid stack classification model, which combines Machine Learning (ML) and Deep Learning (DL) approaches consist of Logistic Regression (LR), Random Forest (RF), and Multi-Layer Perceptron Neural Network (MLP-NN). The hybrid stack classification model accomplishes high accuracy results for classifying Egyptian MASR user requirements as follow: (99.7%) for classifying Performance, (99.5%) for classifying Dissatisfied Users, (98.8%) for classifying Others, and (98.1%) for classifying Security, (97.9%) for classifying Usability, and (97.4%) for classifying Feature Requests.

Keywords: Mobile Applications · Mobile Applications Arabic Slang Reviews · User Requirements · Functional Requirements · Non-Functional Requirements · Sentimental Requirements · Term Frequency – Inverse Document Frequency · Bag of Words · Natural Language Processing · Classifier Chains · Machine Learning · Deep Learning · Logistic Regression · Random Forest · Neural Network

E. S. Nasr—Independent Researcher.

R. Hou et al. (Eds.): BDTA 2021/2022, LNICST 480, pp. 24–37, 2023.
https://doi.org/10.1007/978-3-031-33614-0_2

1 Introduction

With the rising ubiquity and income of mobile applications (apps) market, and app stores have turned into the essential hotspot for the division and possession of mobile apps [1]. Mobile applications (apps) review feature is supplied by most mobile apps platforms like (Google, Apple) app store that authorize users to evaluate, comment, and rate apps after utilizing them, which supplies a feedback technique from users to mobile apps developers. Mobile apps reviews are identified as an oriental source to enhance mobile apps and raise the importance for users [2, 3], as the reviews assist mobile apps developers to recognize better user requirements [4] for meet requirement maintenance, development, and evolution tasks [5, 6]. However, present mobile apps platforms support limited assistance for mobile apps developers to systematically assemble, clarify, and classify feedbacks to recognize user requirements [3]. User review and rating information have been explored for business and technical objectives such as mobile app price prediction [7]. Scientists have therefore fostered various approaches to automatically summarize user reviews contents to elicit the most helpful information they convey [8–10]. For instance, few methodologies plan to classify user reviews into various groups (e.g., feature request, bug report, aspect evaluation, etc.) [11, 12], while some studies focus on cluster and prioritize user reviews to distinguish the most significant topics [13]. Chandy et al. [14] suggested an automatic approach to identify spam reviews in iOS App Store. However, there is few works on automatically and systematically recognizing and classifying requirements from mobile apps user reviews, which will seriously enhance requirements elicitation and analysis in mobile app development.

To this end, the contribution for this research can be summarized as follow: we suggest an automatic approach to recognize requirements information from Egyptian Mobile App Arabic Slang Reviews (MASR) and classify them into Functional Requirements (FR), Non-Functional Requirements (NFR), Sentimental Requirements (SR), and Others requirements, which cannot be classified as FR or NFR or SR. This approach focuses on applying recommended Egyptian keywords selection for identifying, and classifying subcategories for main requirements such as: FR contains [feature requests, bugs, and update], NFR contains [security, performance, usability, frequency, and response time], and SR contains [satisfied users, and dissatisfied users].

In the rest of this paper: Sect. 2 discusses related work and compare some of them with our proposed approach. Section 3 suggests our proposed approach and the process of choosing keywords for automatically identify and classify various requirements for Egyptian Mobile Apps Arabic Slang Reviews. Section 4 displays the evaluation results for classifying Egyptian MASR user requirements. Section 5 summarizes this research with proposed future works.

2 Literature Review

This section debates and abbreviates relevant researches containing user reviews mining, filtering, and classification and their relation to our research in this section. Mobile apps store analysis has become a common subject in Software Engineering (SE) especially for requirements evolution. It allocates users suggest their sentiments, and opinions about the mobile apps they utilized in various app platforms.

Various researches [7, 15, 16] that mining mobile app stores concentrates on analyzing the feature information during user reviews, and recognition their inter-relations with different factors, e.g., code, rating, price, and downloads. Our research approach purposes to integrate mining of mobile app store reviews and identifying Requirements Engineering (RE) to assist developers enhance mobile apps to meet requirements evolution process. Many researchers [17–23] have concentrated on mining and analyzing user reviews in order to derive NFR that can assist developers maintaining and enhancing software systems according to requirement evolution. Our research approach aims to identify and classify various requirements categories: FR, NFR, SR, and other.

Finkelstein et al. [7] presented a technique to extract feature which results are also utilized as the input to estimate the prices of mobile apps, while our research concentrates on user requirements extraction from mobile app reviews. Villarroel et al. [24] classified reviews into proposal for new features, bug reports, and other types, in addition clustered together similar user reviews together and advised developers to address the user review cluster in the following release. Guzman et al. [25] utilized a feature and sentiment approach to elicit various opinions regarding utilizing mobile apps user reviews. Al-Subaihin et al. [26] elicited features from textual specification of mobile apps to cluster the mobile apps utilizing hierarchical clustering and generated mobile apps categorization. Panichella et al. [27] proposed merged technique automatically classify reviews into: information giving, information seeking, feature request, and problem discovery. Chen et al. [13] suggested an automatically framework to mine informative user reviews for mobile app developers, and moreover rank informative reviews. Our research purposes to recognize and classify users' reviews to FR, NFR, SR and others. Rastkar et al. [28] created bug report summaries in order to assist developers to find duplicate bug reports more quickly. Oh et al. [29] devised an automatically algorithm that recognizes informative reviews mirroring user interference to decrease developers' information overload. Chandy et al. [14] suggested an automatic approach to recognize spam reviews in mobile app store utilizing a latent class model with an understandable structure and minimal complexity. Lu et al. [18] attempt to apply supervised Machine Learning (ML) techniques to filter out non-informative reviews in addition to classify reviews into different categories (FRs, NFRs, and Others). Sorbo et al. [9] compiled reviews to suggest future software modifications based on user reviews summarization. Yang et al. [30] integrated regular expression and TF-IDF in addition human participation to classify reviews into NFRs and FRs. Hoon et al. [31] developed combined three emotion, quality, and functional ontologies to classify reviews into three groups. Gao et al. [32] created AR-Tracker tool, which automatically gathers user reviews of mobile apps and rates them in order to improve the reviews set exemplification. Tian et al. [33] analyzed the specifications of mobile apps reviews and recognized the factors of high-rated mobile apps. This research presents that analytics tools will assist developers in managing the vast volume of reviews via filtering, classifying them, to determine what requirements to add, change, or remove [34]. Galvis Carreño et al. [35] concentrated on altering requirements and developing new requirements utilizing the topics recognized from reviews, while our research try to recognize and classify various requirements from Arabic Slang mobile apps reviews. Our proposed approach results can consider to be as inputs to be recognized and classified as new requirements topics in requirements evolution. Wei et al. [36] proposed a multi-label classification dataset of mobile apps user reviews. Results show that it's possible to identify automatically new features requests. Our research concentrated on multi-label identification and classification for Arabic Slang Mobile apps reviews dataset which is requirements are categorized into FR, NFR, SR, and others and each main category include many multi-labels topics. Yang et al. [37] develop TOUR tool for detecting app reviews, identifying sentiments, and prioritizing important reviews for facilitating developers' examination. It applies topic modeling approach. It evaluates by a survey that includes various developers, and All of them attest to the features TOUR suggests changing's practical utility. Sany et al. [38] suggest model for predicting covid-19 reviews into depressed or not. The prediction accuracy is 91% for applied ML algorithms, and 79% for DL algorithms.

3 A Proposed Keyword-Based Feature Extraction Approach for Identifying and Classifying of User Requirements from Egyptian Mobile Apps Arabic Slang Reviews (MASR)

We suggest an automated approach with for identifying and classifying requirements into FR, NFR, and SR from mobile apps Arabic slang reviews especially Egyptian reviews. There are four phases in this approach: MASR collecting, preprocessing, keyword extractor, identifying and classifying requirements, categorize requirements.

Figure 1 specifies the sequence workflow of proposed approach, which involves the following many phases will be discuss in details:

3.1 Phase One: Egyptian Mobile Apps Arabic Slang Reviews (MASR) Collection

In this phase, MASR [39] was elicited according to three main steps including: Scraping, Mobile App Categorizing, and Filtering, as shown in Table 1.

Table 1. MASR Collection phases.

Scraping	Scraper Tool	Applying Appbot Scraper Tool
	App Store	Choose specific Play Store (Google Play Store)
Categorizing	App category	Select various 9 categories of mobile apps (Social, Lifetyle, Travel & Locals, Shopping, Tools, Medical, Productivity, Education, Maps & Navigation)
	App Rating	Determine mobile applications were selected based on their rating of more than 4 scores on google play store in order to ensure the quality of the comments that guarantee the needs of the users,

(*continued*)

Table 1. (*continued*)

Filtering	Reviews Filtering	Choose Country/Language, and Date for crawled reviews. In this research we concern on mixing reviews of Egyptian mobile apps, and another Egyptian reviews for non-Egyptian mobile apps such as: Instagram
	MASR Dataset	Save extracted reviews in CSV file which are contained the following columns: app name, store, app id, review id, country, a star rating from 1 to 5, date, the reviewer's username (author), a review data (body), a translated review body, a review polarity (positive, negative, neutral)), language, reply URL, and app category
	Attributes Filtering	select specific attributes for analyze and classify reviews like: app category, app name, review, rating, and review polarity

3.2 Phase Two: Keyword-Based Feature Requirement Identification and Classification Approach (KBRIC) for Egyptian (MASR)

In this research, we focus on extracting three user requirements categories: FR, NFR, and SR. FR specifies "a task that a system must be capable to execute", NFR is limited to a few distinct characteristics other than functionality, and SR is relating to or involving feelings. In this phase, we suggest to perform three steps: Keyword-based Search for Extracting Annotated Reviews, MASR Preprocessing, Keyword Feature Extraction.

Keyword-Based Search for Extracting Annotated Reviews. This paper utilizes a keyword-based search technique to automatically categorize requirements with FR, NFR, and SR. Some reviews include multi-label classifier. We select a limited set of keywords as shown in Table 2. We focus on the sentences including keywords relevant to keyword search requirements types mentioned in Table 2, for example: ((regex = تعطيل/معطل/عطل*), (regex = تامين/امان/امن)). If sentences include any of the keywords: ((تحسين ، تعديل ، اصدار، تحديث), we assume the review is relevant to the Update issue category), ((اضافة ، مكالمة ، فيديو ، صوت ، خاصية ، ميزة ، بلوك، حذف، اخفاء), we assume the review is relevant to the Feature Requests issue category), ((مستاء ، متضرر ، تعيس ، حزين ، سيء), we assume the review is relevant to the Dissatisfied users issue category)). Such a limited set of keywords assist to decrease false positives.

MASR Preprocessing. The primary step is to perform pre-processing so as to develop the performance of classifiers by converting the text into a form as suitable as possible. This approach applies many stages (normalization, tokenization, stop-word removal and stemming) as shown in Table 3.

Keyword Feature Extraction. To estimate classifiers performance, we utilized various variety of features. Those features can be Bag-of-Words (BOW) with Term Frequency Inverse Document Frequency (TF-IDF) as mentioned in Table 4.

Table 2. Keywords for various Requirements Types.

Requirement Category	Requirements Type	Keywords
FR	**Feature Requests**	اضافة ، مكالمة ، فيديو ، صوت ، خاصية ، ميزة ، بلوك ، حذف ، اخفاء
	Bugs	خطأ، عطل ، متوقف ، معطل
	Update	تحسين ، تعديل ، اصدار، تحديث
NFR	**Frequency**	وقت ، مستمر ، مرة ، مرات ، ثانية، دقيقة، ساعة ، معدل ، متوسط ، باستمرار
	Security & Accounts	مستخدم ، سر ، حساب ، شخصى ، معلومة ، خاص ، احتيال ، فيرس ، الكاميرا ،استماع ، مراقبة ، تصنت ، تجسس ، سرقة، صلاحية، خصوصية
	Performance	تهيئة ، كفاءة ، فعالية ، جدارة، حالة ، وضع
	Response Time	سرعة ، سريع ، بطيئة ، بطيء، استجابة ، انتظار ، تاخير ، تحميل ، متجمد ، متوقف ، معطل ، مختصر ، طويل ، دائم ، متقطع
	Usability	سهل ، صعب ، استهلاك ،مرنة ، استخدام ، شاشة، اجتماعى، التعلم ، تشغيل ، تصفح ، صفحة
SR	**Satisfied users**	راضى ، سعيد ، ممتاز ، جيد ، متحمس

3.3 Phase Three: Automated Supervised Single/Multi-label Classification

This paper utilizes the automatically annotated mobile apps reviews from the preceding phase as training reviews for supervised learning of multi-label classification as mentioned in Table 5. Then utilizes different manually annotated dataset as our validation set for evaluating the of multi-label classifier performance. We first tokenize each mobile app review into BOW form, remove the common Arabic stopwords in addition to Egyptian stopword from [40]. Then transform each MASR review into TF-IDF feature vector. Finally, we suggest to employ Classifier Chains (CC) [41] approach to convert classifying multi-labeled data problem into one or more problems of single labeling, and utilizes the hybrid stack classification model [39] which combine Machine Learning (ML) and Deep Learning (DL) approaches which comprised of three classification techniques: Logistic Regression (LR) + Random Forest (RF)+ Multi-layer Perceptron Neural Network (MLP-NN) which accomplish high accuracy results 89.4% on classifying Egyptian MASR polarity sentiments (positive or negative or neutral).

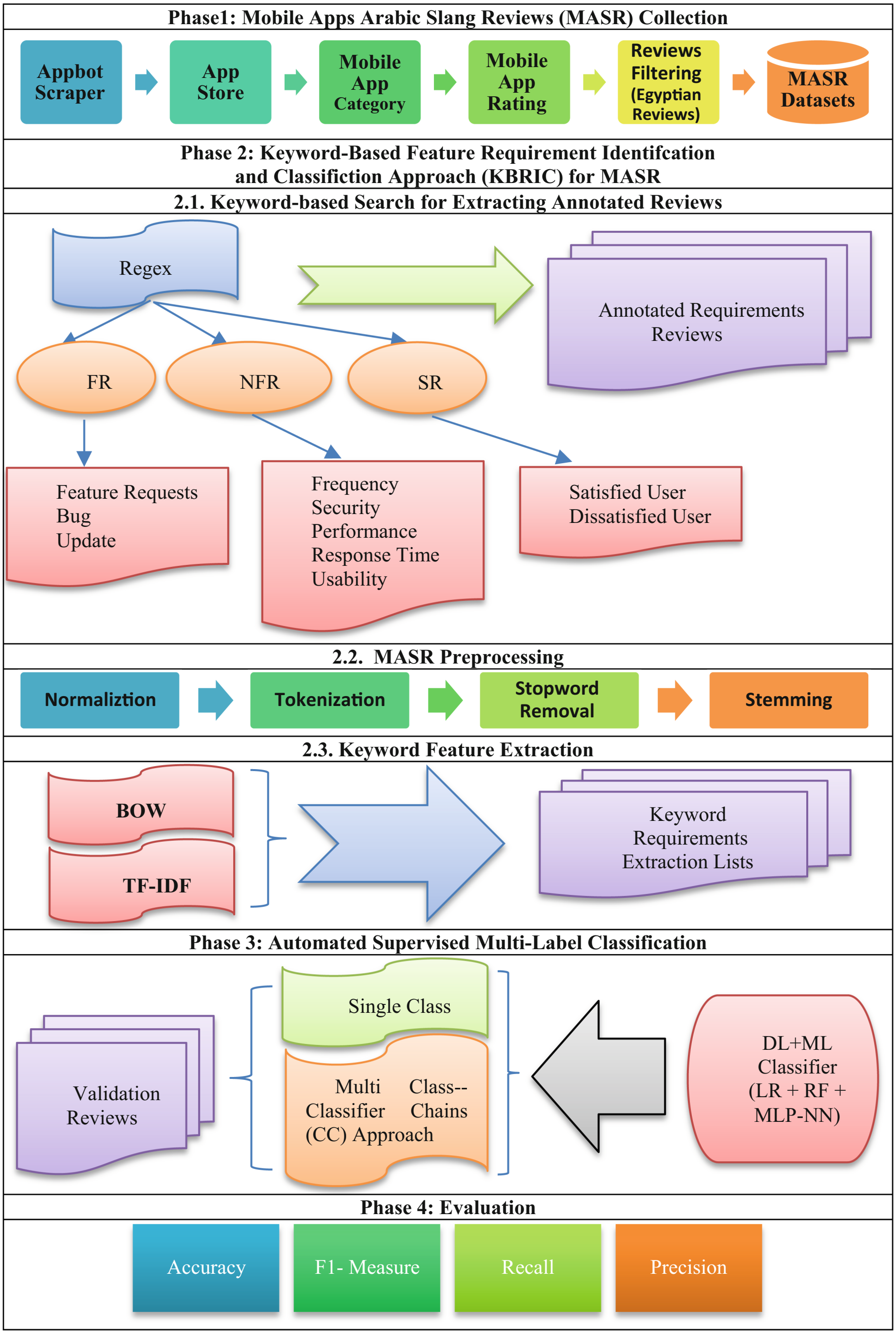

Fig. 1. Keyword-Based Feature Extraction Approach

Table 3. MASR preprocessing phases

Normalization	This step includes applying two tasks: removing and replacing. First: Removing punctuation marks, special characters, all diacritics, digit numbers, and non-Arabic words. Second: Replacing each final letter (ي) with (ى), Replace initial letter alef-hamza (أ، إ،ء،ئ،ؤ، آ) with (ا), and replace each final letter (ة) with (ه).
Tokenization	separate text by word or by sentence. This will allow deal with smaller groups of text that are still relatively significative regular outgoing of the context of the rest of the text.
Stop-word Removal	abstract all stop-words from apps reviews. In this research, Arabic stop word list used from various resources also specific Egyptian stop word [40].
Stemming	technique of reducing a word to its root. In this research, Snowball stemmer performs on MASR dataset.

Table 4. Feature Extraction Techniques

BOW	an operation of extracting features from text for utilize in modeling, such as with ML algorithms. BOW model assigns a corpus with word counts for every document
TF-IDF	a statistical method measure utilized to evaluate the importance of a word to a doc-ument in a corpus. The significance grows proportionally to the frequency of times a word represents in the document

3.4 Phase Four: Evaluation

To evaluate our proposed approach, we suggest to apply those performance evaluation measures [42]: Accuracy (ACC), F1-measure (F1), Precision (PRE), and Recall (REC) as mentioned in Table 6.

Table 5. Requirements Types and Categories for Egyptian MASR

Req.	Req.Type	Egyptian MASR	Translated Review	Pol.
FR	**Feature Re-quests**	حلو بس لو تضيفو مكالمة صوتية بس بدون فيديو	Nice, but if you add an audio call, but without a video	Pos.
	Bugs	لدي مشكلة في قسم فودافون كاش حيث دائمًا ما تعطيني خطأ داخلي في الخادم ، لا أعرف السبب	I have a problem in Vodafone cash section in which it always gives me internal server error, I don't know why	Neg.
	Update	الإصدار الجديد مليء بالأخطاء بحتاج للتعديل...	The new version is full of bugs and need update…	Neg.
NFR	**Usability**	تطبيق سهل فى التعامل وتفعيل والغاء الخدمات ومتابعة الاستهلاك	An easy application in dealing, activating and canceling services, and following up on consumption	Pos.
	Security	الابلكيشن دايما بيدينى خطأ بالرغم انى بدخل ويب من غير مشاكل نهائي.. ارجو افادتي بحل للمشكلة علشان اقدر اعيد التقييم تاني.	The application is always wrong password or user name in my hands, even though I am on a normal web income without a final problem .. Please advise me of a solution to the problem so that I can re-evaluate again	Neg.
	"Frequency", "Performance **[Multi-Label]**	لابد من ادخال تفاصيل بطاقتى فى كل مرة على الرغم من ان التطبيق يقول انه سيحفظ المعلومات. ايضا ، خدمة العملاء الخاصة بك هى الابطأ التى رايتها على الاطلاق. الدردشة المباشرة بطيئة للغاية ، و يستغرق الاشخاص وقت اطول للعودة ، فقط ليقولوا ان الطلب سيكون هنا خلال 10 الى 15 دقيقة فى كل مرة.	I have to enter my card details every time even though the app says it'll save the information. Also, your customer service is the slowest I've ever seen. Live chat is too slow, and your people take too long to get back, only to say the order will be here in 10 to 15 mins. Every. Single. Time.	Neg.
	Response Time	حلو جدا و ثقافي.؛ بس ممكن تخلو سريع	Very sweet and cultural.; You can just give up quickly	Pos.
SR	**Satisfied users**	كنت راضيه كل الرضا عن خدمة خطوطكم متألقين واتمنى لكم كل التوفيق و انصح باستخدام البرنامج	I was completely satisfied with the service of your lines shining and I wish you all the best and I recommend using the program	Pos.
	Dissatisfied Users	خدمة عملاء سيئة للغاية أنا مستاء	very bad customer service I'm upset	Neg.

4 Results and Discussion

For empirical research, we apply a hybrid ML, and DL stack Model [39] for classifying 600 random sample of Egyptian MASR user requirements datasets. Figure 2 displays Egyptian MASR requirements distribution as follow: FR [Feature Requests 7%, Bugs

Table 6. Evaluation Measures

ACC	evaluates the performance measure, and it represents a ratio of correctly predicted examination to the total examinations	$\frac{\sum TP+\sum TN}{\sum TP+FP+TN+FN}$ (1)
F1	a technique that illustrates the incorporated precision, while recall denotes the harmonic medium of recall and precision, and traditional F-measure	$\frac{2*Precision*Recall}{(Precision*Recall)}$ (2)
PRE	evaluates the number of True Positives (TP) divided by the number of True Positives (TP) and False Positives (FP)	$\frac{TP}{TP+FP}$ (3)
REC	refers to positive predictive value (PPV). It evaluates the true positive rate	$\frac{TP}{TP+FN}$ (4)

10%, Update 14%], NFR [Security 5%, Performance 2%, Response Time 6%, Usability 2%, Frequency 5%], SR [Satisfied Users 44%, Dissatisfied Users 3%], and Others 2%. We apply K-Fold Cross Validation (CV) with K = 10. And finally, we implement four evaluation measures: ACC, F1, PRE, RE for evaluating Egyptian MASR user requirements to assist mobile app developers to meet user requirements evolution as shown in Table 7.

Table 7. Evaluation Measures Results for Egyptian MASR User Requirements

		ACC	F1	PR	RE
FR	Feature Requests	97.4%	80%	85.7%	75%
	Bugs	86.9%	53.4%	43%	70.5%
	Update	94.6%	83.2%	76.2%	91.7%
NFR	Security	98.1%	81.4%	**92.3%**	72.7%
	Performance	**99.7%**	35%	33%	28%
	Response Time	94.1%	6%	50%	3%
	Usability	97.9%	13%	33%	9%
	Frequency	94.4%	11%	33%	7%
SR	Satisfied Users	91.5%	**91.4%**	86.1%	**97.4%**
	Dissatisfied Users	99.5%	15%	35%	9%
Others		98.8%	17%	37%	8%

Accuracy. Table 7 shows that the hybrid classification model accomplishes high accuracy (99.7%) for classifying Performance, (99.5%) for classifying Dissatisfied Users,

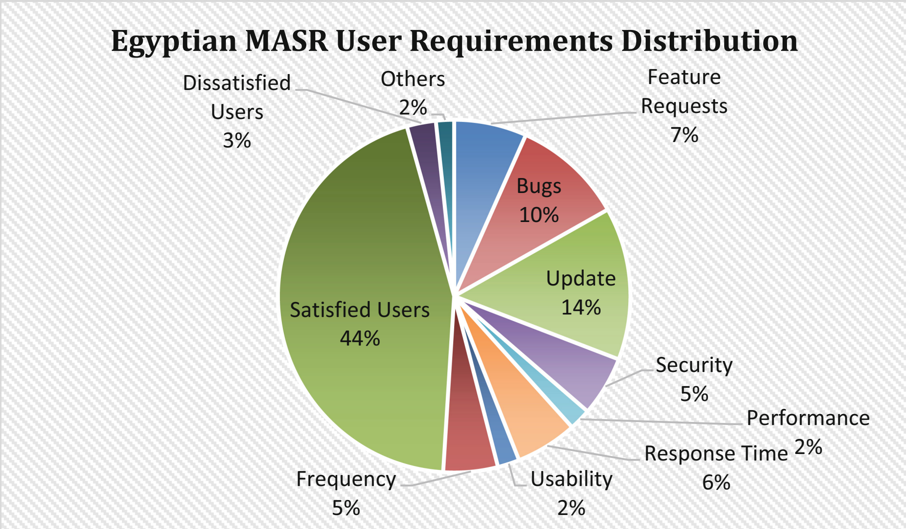

Fig. 2. Distribution of Egyptian MASR User Requirements

(98.8%) for classifying Others, (98.1%) for classifying Security, (97.9%) for classifying Usability, and (97.4%) for classifying Feature Requests.

F1-Measure. Table 7 shows that the hybrid classification model accomplishes high F1-measure (91.4%) for classifying Satisfied Users, (83.2%) for classifying Update, (81.4%) for classifying Security, and (80%) for classifying Feature Requests.

Precision. Table 7 shows that the hybrid classification model accomplishes high precision (92.3%) for classifying Security, (86.1%) for classifying Satisfied Users, (85.7%) for classifying Feature Requests, and (76.2%) for classifying Update.

Recall. Table 7 shows that the hybrid classification model accomplishes high recall (97.4%) for classifying Satisfied Users, (91.7%) for classifying Update, (75%) for classifying Feature Requests, and (72.7%) for classifying Security.

5 Conclusion and Future Work

In this paper, we aim to propose an automatic approach for identifying and classifying user requirements into FR, NFR and SR from Egyptian MASR. TF-IDF, BOW and NLP techniques utilized for extracting keywords. A keyword list was collected for FR consists of (Feature Request, Bug, and Update), NFR consists of (Response Time, Usability, Performance, Security, Frequency), and SR consists of (Satisfied Users, Dissatisfied Users). For multi-labeled problem, we suggest applying the CC approach to classify multi-labeled data problems into one or more problems of single labeling. We also suggest to utilizes the hybrid stack classification model which combines ML and DL approaches consist of LR, RF, and MLP-NN, and also evaluate our hybrid approach using

ACC, F1, PRE, RE measures, which accomplish high accuracy results for classifying Egyptian MASR user requirements as follow: (99.7%) for classifying Performance, (99.5%) for classifying Dissatisfied Users, (98.8%) for classifying Others, and (98.1%) for classifying Security, (97.9%) for classifying Usability, and (97.4%) for classifying Feature Requests.

In the future, we intend to apply different keyword feature extraction methods such as n-grams, word enrichment, and word embedding. We plan to apply Topic Modeling approach for extracting other topics that represent new requirements from Others requirements that cannot be classified as FR, NFR, and SR from Egyptian MASR.

References

1. Holzer, A., Ondrus, J.: Mobile application market: a developer's perspective. Telematics Inform. **28**(1), 22–31 (2011)
2. Bano, M., Zowgh, D.: User involvement in software development and system success: a systematic literature review. In: Proceedings of the 17th International Conference on Evaluation and Assessment in Software Engineering (ESEM) (2013)
3. Abelein, U., Sharp, H., Paech, B.: Involving users in software development really influence system success? IEEE Softw. **30**(6), 17–23 (2013)
4. Liang, P., Avgriou, P., He, K., Xu, L.: From collective knowledge to intelligence: pre-requirements analysis of large and complex systems. In: Proceedings of the 1st Workshop on Web 2.0 for Software Engineering (2010)
5. Saudy, R.E., Nasr, E.S., El-Ghazaly, A.E.D.M., Gheith, M.H.: Use of arabic sentiment analysis for mobile applications' requirements evolution: trends and challenges. In: Hassanien, A.E., Shaalan, K., Gaber, T., Tolba, M.F. (eds.) AISI 2017. AISC, vol. 639, pp. 477–487. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-64861-3_45
6. Francese, R., Gravino, C., Risi, M., Scanniello, G., Tortora, G.: Mobile app development and management: results from a qualitative investigation. In: Proceedings of the 2017 IEEE/ACM 4th International Conference on Mobile Software Engineering and Systems (MOBILESoft) (2017)
7. Finkelstein, A., Harman, M., Jia, Y., Sarro, F., Zhang, Y.: Mining App Stores: Extracting Technical, Business and Customer Rating Information for Analysis and Prediction. Research Note RN/13/21 (2013)
8. Phong, M.V., The Nguyen, T., Viet Pham, H., Thanh Nguyen, T.: Mining user opinions in mobile app reviews: a keyword-based approach. In: Proceedings of the 2015 30th IEEE/ACM International Conference on Automated Software Engineering (ASE) (2015)
9. Di Sorbo, A., et al.: What would users change in my app? summarizing app reviews for recommending software changes. In: Proceedings of the 2016 24th ACM SIGSOFT International Symposium on Foundations of Software Engineering (2016)
10. Gao, C., Wang, B., He, P., Zhu, J., Zhou, Y., Lyu, M.R.: Paid: Prioritizing app issues for developers by tracking user reviews over versions. In: Proceedings of the 2015 IEEE 26th International symposium on software reliability engineering (ISSRE) (2015)
11. Guzman, E., El-Haliby, M., Bruegge, B.: Ensemble methods for app review classification: an approach for software evolution (N). In: Proceedings of the 30th IEEE/ACM International Conference on Automated Software Engineering, ASE'15 (2015)
12. McIlroy, S., Ali, N., Khalid, H., Hassan, A.E.: Analyzing and automatically labelling the types of user issues that are raised in mobile app reviews. Empirical Softw. Eng. **21**(3), 1067–1106 (2015). https://doi.org/10.1007/s10664-015-9375-7

13. Chen, N., Lin, J., Hoi, S.C.H., Xiao, X., Zhang, B.: AR-miner: mining informative reviews for developers from mobile app marketplace. In: Proceedings of the 36th International Conference on Software Engineering (ICSE), Hyderabad, India (2014)
14. Chandy, R., Gu, H.: Identifying spam in the iOS App Store. In: Proceedings of the 2Nd Joint WICOW/AIRWeb Workshop on Web Quality, WebQuality'12. ACM (2012)
15. Harman, M., Jia, Y., Zhang, Y.: App store mining and analysis: MSR for app stores. In: Proceedings of the 2012 9th IEEE Working Conference on Mining Software Repositories (MSR) (2012)
16. Zaidman, A., Van Rompaey, B., Demeyer, S., Van Deursen, A.: Mining software repositories to study co-evolution of production & test code. In: Proceedings of the 1st International Conference on Software Testing, Verification, and Validation (ICST) (2008)
17. Garba, S., Isyaku, B., Abdullahi, M.: A-driven model for non-functional requirements in mobile application development. Int. J. Comput. Sci. Inform. Technol. (IJCSIT) **11**(2), 97–109 (2019)
18. Lu, M., Liang, P.: Automatic classification of non-functional requirements from augmented app user reviews. In: Proceedings of the 21st International Conference on Evaluation and Assessment in Software Engineering (2017)
19. Corbalán, L., et al: A study of non-functional requirements in apps for mobile devices. In: Proceedings of the Conference on Cloud Computing and Big Data, Cham (2019)
20. Tóth, L., Vidács, L.: Study of various classifiers for identification and classification of non-functional requirements. In: Proceedings of the International Conference on Computational Science and Its Applications, Cham (2018)
21. Ahmad, A., Feng, C., Li, K., Asim, S.M., Sun, T.: Toward empirically investigating non-functional requirements of iOS developers on stack overflow. IEEE Access **7**, 61145–61169 (2019)
22. Wang, T., Liang, P., Lu, M.: What aspects do non-functional requirements in app user reviews describe?: an exploratory and comparative study. In: Proceedings of the 25th Asia-Pacific Software Engineering Conference (APSEC) (2018)
23. Saudy, R.E., Nasr, E.S., El-Ghazly, A.E.D.M., Gheith, M.H.: A comparative framework for Arabic sentiment analysis research. In: The 54thAnnual Conference on Statistics, Computer Sciences and Operation Research, Egypt (2019)
24. Villarroel, L., Bavota, G., Russo, B., Oliveto, R., Di Penta, M.: Release planning of mobile apps based on user reviews. In: Procedings of the 2016 IEEE/ACM 38th International Conference on Software Engineering (ICSE) (2016)
25. Guzman, E., Aly, O., Bruegge, B.: Retrieving diverse opinions from app reviews. In: Proceedings of the ACM/IEEE International Symposium on Empirical Software Engineering and Measurement, ESEM'15. IEEE (2015)
26. Al-Subaihin, A.A., Sarro, F., Black, S., Capra, L., Harman, M., Jia, Y., Zhang, Y.: Clustering mobile apps based on mined textual features. In: Proceedings of the 10th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (2016)
27. Panichella, S., Sorbo, A.D., Guzman, E.: How can i improve my app? classifying user reviews for software maintenance and evolution. In: Proceedings of the 31st IEEE International Conference on Software Maintenance and Evolution (2015)
28. Rastkar, S., Murphy, G.C., Murray, G.: Automatic summarization of bug reports. IEEE Trans. Software Eng. **40**(4), 366–380 (2014)
29. Oh, J., Daehoon, K., Lee, U., Lee, J.-G., Song, J.: Facilitating developer-user interactions with mobile app review digests. In: Proceedings of the CHI'13 Extended Abstracts on Human Factors in Computing Systems, CHI EA'13, ACM (2013)
30. Yang, H., Liang, P.: Identification and classification of requirements from app user reviews. In: Proceedings of the 27th International Conference on Software Engineering and Knowledge Engineering (SEKE'15) (2015)

31. Hoon, L., Rodriguez-García, M.A., Vasa, R., Valencia-García, R., Schneider, J.-G.: App reviews: breaking the user and developer language barrier. In: Mejia, J., Munoz, M., Rocha, Á., Calvo-Manzano, J. (eds.) Trends and Applications in Software Engineering, pp. 223–233. Springer International Publishing, Cham (2016). https://doi.org/10.1007/978-3-319-26285-7_19
32. Gao, C., Xu, H., Hu, J., Zhou, Y.: AR-tracker: track the dynamics of mobile apps via user review mining. In: Proceedings of International Workshop on Internet based Virtual Computing Environment (IVCE), San Francisco, USA (2015)
33. Tian, Y., Nagappan, M., Lo, D., Hassan, A.E.: What are the characteristics of high-rated apps? a case study on free android applications. In: Proceedings of the 2015 IEEE international conference on software maintenance and evolution (ICSME) (2015)
34. Maalej, W., Nayebi, M., Johann, T., Ruhe, G.: Toward data-driven requirements engineering. IEEE Softw. **33**(1), 48–54 (2015)
35. Galvis Carreño, L.V., Winbladh, K.: Analysis of user comments: an approach for software requirements evolution. In: Proceedings of the 2013 International Conference on Software Engineering, ICSE'13. IEEE Press (2013)
36. Wei, J., Courbis, A.-L., Lambolais, T., Xu, B., Bernard, P.L., Dray, G.: Towards a data-driven requirements engineering approach: automatic analysis of user reviews. In: Proceedings of the 7th National Conference on Practical Applications of Artificial Intelligence, Saint-Étienne (2022)
37. Yang, T., Gao, C., Zang, J., Lo, D., Lyu, M.R.: TOUR: dynamic topic and sentiment analysis of user reviews for assisting app release. In: Proceedings of the WWW'21: The Web Conference 2021, Ljubljana, Slovenia (2021)
38. Sany, M.M.H., Keya, M., Khushbu, S.A., Rabby, A.S.A., Masum, A.K.M.: An opinion mining of text in COVID-19 Issues along with comparative study in ML, BERT & RNN. In: Proceedings of the International Conference on Deep Learning, Artificial Intelligence and Robotics, Cham (2022)
39. Saudy, R.E., El-Ghazaly, A.E.D.M., Nasr, E.S., Gheith, M.H.: A novel hybrid sentiment analysis classification approach for mobile applications Arabic slang reviews. Int. J. Adv. Comput. Sci. Appl. **13**(8), 423–432 (2022)
40. Medhat, W., Yousef, A., Korashy, H.: Egyptian dialect stopword list generation from social network data. The Egypt. J. Lang. Eng. **2**(1), 43–55 (2015)
41. Read, J., Pfahringer, B., Holmes, G., Frank, E.: Classifier chains for multi-label classification. Mach. Learn. **85**(3), 333–359 (2011)
42. Yamout, B., et al.: Predictors of quality of life among multiple sclerosis patients: a comprehensive analysis. Eur. J. Neurol. **20**(5), 756–764 (2013)