Fine-Grained Categorization of Mobile Applications Through Semantic Similarity Techniques for Apps Classification

Elena Flondor[1(✉)] and Marc Frincu[2]

[1] Faculty of Mathematics and Computer Science, West University of Timisoara, bv. Vasile Parvan, Timisoara, Romania
elena.flondor97@e-uvt.ro

[2] School of Science and Technology, Nottingham Trent University, Cliftom Campus, Nottingham, UK
marc.frincu@ntu.ac.uk

Abstract. The number of Android apps is constantly on the rise. Existing stores allow selecting apps from general named categories. To prevent miscategorization and facilitate user selection of the appropriate app, a closer examination of the categories' content is required to discover hidden subcategories of apps. Recent work focuses on exploring the granularity of the categories, but a validation of the categories' content against miscategorized apps is missing. In this research, we apply semantic similarity to apps' descriptions to uncover similarity and hierarchical clustering to search for misclassified apps. Furthermore, we apply Latent Dirichlet Allocation (LDA) algorithm to explore the existence of possible subcategories and to classify apps. Our empirical research is conducted using two data sets: 9,265 apps from Google Play Store, and 300 apps from App Store. Results confirm the existence of misclassified apps on markets and suggest the existence of multiple fine-grained categories. Our experiments outperform other LDA-based classification approaches achieving 0.61 precision. Moreover, the analysis hints the presence of misclassified apps might decrease the performance of existing classifiers.

Keywords: Semantic Similarity · Hierarchical Clustering · Latent Dirichlet Allocation · Mobile Applications

1 Introduction

The ongoing evolution of mobile devices in recent years has significantly impacted our lives. Developers of mobile apps share their products in the rapidly expanding markets of Google Play Store (GPS) [2] for Android and App Store (AS) [1] for iOS. On these platforms, developers must select a relevant category to help users find suitable apps easily. However, with thousands of apps in each category, finding apps that match consumer interests can be challenging. Moreover, some may be misclassified, making their discovery even more difficult.

O. Pedreira and V. Estivill-Castro (Eds.): SISAP 2023, LNCS 14289, pp. 79–87, 2023.
https://doi.org/10.1007/978-3-031-46994-7_7

Existing work is focused to find subcategories of apps which are hidden in the main markets' categories. Supervised and Unsupervised Machine Learning classification techniques combined with Natural Language Processing (NLP) techniques were applied to solve this challenging task. However, few of researchers mention misclassified apps on the markets [11].

We focus on validating categories' content and automating the process of identification of misclassified apps for prevention, and finding possible subcategories of apps on the markets to facilitate appropriate user app selection. We apply *semantic similarity* [10] and *hierarchical clustering* [15] to determine misclassifications, and *Latent Dirichlet Allocation* (LDA) [6] to find apps subcategories. A text-to-text *semantic similarity metric* is applied, as it outperforms traditional text similarity metrics [10]. Hierarchical clustering is applied for its capacity to represent data and to identify clusters that deviate significantly from the rest of the data. Lastly, LDA is applied to find mobile apps *subcategories* for its capacity to resume content, and to find relevant groups of words (called *topics*) which can provide insights of the main data categories. Specifically, we aim to:

- Propose an automated method for detection of miscategorized apps based on apps description using semantic similarity, hierarchical clustering, and markets classification recommendations [18,19];
- Discover hidden subcategories of apps by applying LDA on a well-defined and processed data set containing apps descriptions;
- Apply LDA to classify apps in proposed subcategories;

2 Literature Survey

This section explores existing methods of mobile apps classification based on their description corpus and reviews applied NLP techniques.

Several works are focused on exploiting the content of existing categories to extend current classification on the markets. For instance, [16,20] exploited the capacity of LDA and discovered multiple topics as an extension to the initial classification. Al-Subaihin et al. [3], extracted features from apps descriptions using the algorithm proposed in [12]. Then, a greedy hierarchical clustering algorithm was applied and the clusters, interpreted by humans, denoted that the initial categories can be extended. Ebrahimi et. al [11] encoded apps descriptions using different word embeddings and achieved the best classification results with GloVe and Support Vector Machines. Apps from Education and Health&Medical categories were manually labeled by researchers to find subcategories of apps.

The researchers applied NLP techniques to encode the descriptions: stop words removal and tokenization [11,13,20], part of speech tagging (identification of nouns, verbs, etc.) and stemming [13] or lemmatization [3,11], and adding n-grams [3]. All of these can help in boosting the performance of the classification techniques if they are combined in a proper manner.

Compared to the preceding works, our initial aim is to validate the content of the categories based on the similarity between the apps descriptions and

the markets recommendations. For data processing, we combine various NLP techniques compared to previous researchers (Sect. 3.2). Then, we apply the method proposed by [10] to compute semantic similarity between apps' descriptions and fed the similarity distance matrix into hierarchical clustering algorithm [15]. Based on the distribution of the apps into clusters, we propose a new method to establish if a category presents misclassified apps and we search for subcategories of apps. Due to its promising results [20], we apply LDA to determine subcategories of apps in each category. We analyse the granularity of the categories and perform human interpretation [9] of the generated topics to validate our findings.

3 Methodology

The work discussed in Sect. 2 was adapted by combining more NLP techniques to reduce the noise of the data and to strengthen its semantic quality (Sect. 3.2). We searched for misclassified apps using *semantic similarity measures* and *hierarchical clustering* and we used LDA to find subcategories of apps included in each main category. Moreover, contrary our predecessors, we validate the content of the categories before investigating the fine-grained categories.

Data Set Gathering (Sect. 3.1) and **Data Set Processing** (Sect. 3.2) are the first two phases in our research. In the third phase we propose our method for **Categories' Content Validation** (Sect. 3.3). In the fourth phase we apply LDA in **Mining the Categories' Content** (Sect. 3.4) to discover the fine-grained categorization of the initial categories. The **Evaluation** (Sect. 3.5) phase describes the steps applied to validate our approach.

3.1 Data Set Gathering

We use a set of 9,265 Android apps to analyse fine-grained categories of GPS. It consists in English descriptions of apps from GPS between 2019-2023. They are classified in 32 categories based on the short outlines provided by the market [19]. We excluded the *Game* category as it already encompasses numerous fine-grained subcategories. The apps' descriptions were processed and LDA was applied to determine topics for subcategories (Sect. 4). To evaluate the performance of LDA in topics identification we used the data set proposed in [11]. It consists in apps crawled from AS. We selected educational apps and compared the ground-truths set by prior work (Sect. 4) with our finding.

3.2 Data Set Processing

We applied various *NLP techniques* to process the descriptions of the apps. We prepared the data to minimize the noise and to enable the algorithms to uncover latent relationships between words. We applied the following steps: uppercase characters were lowered; special characters, stop words, words with less than three characters, URLs, emojis, and numerical values were removed. Thus, we

kept the most appropriate words and reduced the computational time. Then, we applied tokenization and each obtained term was labeled corresponding to its part of speech (e.g., noun, verb) to compute *semantic similarity scores* (Sect. 3.3). For subcategories discovering phase (Sect. 3.3), the remaining terms were converted to their lemma equivalent to maintain the inherent nature of words and we improved the description corpus with co-occurring word pairs of length 2, called *bi-grams* [14], as they were proven to bring improvements in topics discovering [7]. Finally, we filtered out words occurring in less than 10% and in more than 50% of the description corpus to allow LDA to build stronger relationships between the most relevant words.

3.3 Categories' Content Validation

This section proposes a method to validate the content of the categories by applying *semantic similarity measures* and *hierarchical clustering* algorithm.

Semantic Similarity Measure. To preserve the semantic similarity relations between the words, we applied the method for text similarity proposed in [10] on the apps descriptions level. The algorithm was applied to each pair of descriptions from the same category: for each part of speech from one description, we identified the one with the highest similarity (max_sim) to the other. For nouns and verbs the similarity is computed based on *Wu and Palmer similarity* [21], as it computes the degree of similarity between word senses and where the rings of synonyms occur relative to each other. As we removed cardinals (Sect. 3.2), we computed lexical similarity only for adjectives and adverbs. Therefore, a directional similarity score is given by: $sim(D_1, D_2)_{D_1} = \frac{\sum_{w_i \epsilon D_1} max\_sim(w_i, D_2) \times idf(w_i)}{\sum_{w_i \epsilon D_1} idf(w_i)}$, where D_1, D_2 are descriptions, w_i is the i^{th} word in a description, and $idf(w_i)$ is the Inverse Document Frequency [17]. Both directional scores can be combined into a bidirectional similarity function: $sim(D1, D2) = \frac{sim(D1,D2)_{D_1} + sim(D1,D2)_{D_2}}{2}$. As its value ranges from 0 to 1, it allows the conversion into a normalized distance function: $dist\_sem(D_1, D_2) = 1 - sim(D_1, D_2)$ [5].

Hierarchical Clustering. We used *hierarchical clustering* [15] to investigate the granularity levels from the categories of mobile apps. Using the *similarity distance function* ($dist\_sem$), we computed the *semantic similarity distance matrix*. The shape of the matrix is given by the number of apps from a category. Let $E(D_i, D_j)$ the value of a matrix entry, where (D_i, D_j) is a pair of apps descriptions, and i, j general notations for line and column indices in the matrix. If $i = j$, we set $E(D_i, D_j) = 0$, else $E(D_i, D_j) = dist\_sem(D_i, D_j)$. We fed the similarity distance matrix obtained into the *complete linkage clustering* [15] to determine the clusters based on the maximum distance between the data points.

Possible Outliers Identification. *We define an outlier as a misclassified app* and analyse the distribution of the apps in the formed clusters to discover possible outliers. We consider the first two formed clusters to establish the existence

of possible outliers. If they are highly unbalanced (e.g., a cluster contains only one or two descriptions), we compute the *semantic similarity* between the possible outlier app's description and each one of the categories recommendations proposed by the market source [18,19]. If the maximum similarity is obtained for its main category, then it is not an outlier. The identified outliers were removed.

3.4 Mining the Categories' Content with LDA

The process continues with the discovery of different subcategories of apps within a category. We chose LDA [6], as it uses statistical models to infer topics in text data [4]. The main objective of LDA is to discover the *document topics*, within a text document. This produces a *topic-keyword* matrix. In our research, LDA is applied to the description corpus of the Android apps and the resulting topics are assumed to represent mixtures of a basic set of keywords that may describe a subcategory of apps. The number of the LDA topics, was determined based on *the highest coherence score*, as it measures the interpretability and semantic coherence of the generated topics.

3.5 Evaluation Framework

For evaluation we used *topic labeling* with *human interpretation* [9]. In the case of the second data set, we measured the performance of LDA to group mobile apps in the same category. Given the ground truth labels, we evaluated the performance of LDA by: $F2Score = \frac{(\beta^2+1)\times Precision\times Recall}{\beta^2\times Precision+Recall}$ ($\beta = 2$), $Precision = \frac{TP}{TP+FP}$, and $Recall = \frac{TP}{TP+FN}$. We define: TP (true positive) as the number of the apps whose label of the assigned topic matches the initial category label; TN (true negative) as the number of apps correctly classified as not belonging to inappropriate topics; FP (false positive) as the number of apps incorrectly assigned; FN (false negative) as the number of apps incorrectly excluded from appropriate topics.

4 Experiments and Results

In this section we analyse the results of outlier identification (Sect. 4.1), and the fine-grained categorization resulted using LDA (Sects. 4.2 and 4.3).

4.1 Outliers Identification Analysis

We investigated the feasibility of our approach to identify misclassified apps among the GPS data set (Sect. 3.3) and discovered two categories with outliers: *Medical* and *Weather*. Hierarchical clustering was applied for two clusters, grouping 370 samples in cluster 1 and leaving 1 sample in cluster 2 for *Medical* category. For *Weather* category, it grouped 293 in the first cluster, and left 1 in the second one. We noticed that the outlier of the *Medical* category could better fit in the *Education* category. Moreover, the maximum semantic similarity score (Sect. 3.3) between the *Medical* outlier and the categorization recommended by

GPS [19] was *0.40* for *Education*. For the *Weather* category outlier, we analysed the app and decided that it should be classified as a Game. We did not applied the same validation as in case of the other outlier, as GPS market only provides a list of games subcategories.

4.2 Google Play Store-Fine-Grained Categorization

To obtain consistent categories, we first searched for and removed possible outliers. We then applied LDA to explore the fine-grained categorization of mobile apps. For an *optimal* number of topics we investigated the behavior of *coherence scores* in relation to changes in granularity within clustering apps, which proved beneficial in practice [9]. We provided LDA with granularity levels ranging from 2 topics per category to the maximum based on apps count. We evaluated the coherence scores for each category cluster, selecting the topic count that maximized coherence scores, as topic words might define a subcategory. Given the GPS data set, the number of obtained topics sums up to 236 and hints the existence of more than 200 possible subcategories (Sect. 4.3). The subcategories coherence scores range from 0.37 (*Libraries & Demo*) to 0.58 (*Video Players & Editors*), and their average is 0.48. As the score typically ranges from 0 to 1 (Sect. 3.4), these can be promising results. The number of determined topics, might correspond to the number of existing subcategories. It ranges from 2 to 20 and does not depend on the number of the apps in each category, but on the logical relations among the keywords that exhibit higher probabilities (Sect. 3.4).

4.3 App Store Fine-Grained Categorization

To assess the ability of LDA to generate fine-grained app categories, we used the data set from [11]. We considered the *300 educational apps*, which were manually labeled with specific subcategories. We took into consideration this category as it contains misfit apps. The researchers [11] labeled them into: *Skill-based* apps, *Content-based* apps, *Function-based* apps, *Games*, and *Misfits*.

We evaluated LDA on: classification into all five types of apps (Scenario 1); classification into *Skill-based*, *Content-based*, *Function-based*, and *Games* - to analyse the effect of removing misclassified apps (Scenario 2). Furthermore, we analyse the semantic similarity between the misfits and *Education* category content recommendations and analyse their inclusion in this category. We processed the apps descriptions (Sect. 3.2) and used LDA to generate a number of topics equal to the number of subcategories. To compare our results to those obtained in [11], we labeled the topics based on contained words (Table 1) and through human interpretation: *1 - Skill-based*, *2 - Function-based*, *3 - Content-based*, *4 - Games*. We assigned topics to each description and applied evaluation metrics (Sect. 3.5). Scenario 1 achieved the lowest performance: 0.36 *precision*, 0.38 *recall* and *f2-measure*; compared to Scenario 2: 0.614 *precision*, 0.582 *recall*,

and 0.588 for *f2-measure*. However, Scenario 1 overachieved in terms of precision several classification algorithms applied in [11] on LDA vectorized descriptions (Naïve Bayes - 0.27, SVM - 0.35, Logistic Regression - 0.35). Our results show that removing misclassified apps from the categories can increase the performance of a classifier by approximately 25%. Therefore, we obtained better results compared to [11] after *Misfits* removal. We overachieved results from [11] when description was encoded using VSM (Naïve Bayes, AdaBoost, Decision Trees) and BM25 (Naïve Bayes, AdaBoost, Random Forest, KNN, SVM, etc.). For *Misfits* we applied the *bidirectional semantic similarity measure* (Sect. 3.3) between each app description and AS categories recommendations [8] to discover their most suitable main categories. E.g., we found out that: *A&A Days* is more suitable for *Productivity*, *Vuga Conf* should correspond to *Social Networking*, *Weather Saying* to *Weather*, *AR Paintings* to *Graphics & Design*, etc.

Table 1. Word topics used to establish subcategories for the markets

ID	Google Play's Education Topics	App Store's Education Topics
1	study, help, course, solution, practice, function, math, problem, data, skill	math, test, question, learn, practice, help, number, skill, exam, lesson,
2	question, test, student, class, exam, school, information, quiz, book, homework	school, feature, student, parent, information, work, access, mobile, course, service,
3	language, lesson, vocabulary, word, speak, pronunciation, course, sentence, conversation, grammar	book, study, view, available, share, version, story, record, search, experience,
4	word, feature, use, dictionary, search, time, photo, easy, share, translation, text	word, learn, game, kid, vocabulary, letter, color, picture, animal, sound,
5.	video, child, plant, world, book, play, song, educational, design, parent	–

5 Conclusion and Future Work

This paper proposed the identification of misclassified mobile apps in the markets and evaluated the performance of LDA in discovering subcategories of apps based on their descriptions. The process was done by applying NLP techniques (Sect. 3.2) and the experimental setup was based on averaged semantic similarity and hierarchical clustering to determine misclassified apps on GPS. Apps proposed in [11] were used to evaluate LDA in finding topics. We labeled the identified topics corresponding to [11] through human interpretation and evaluated LDA

in classification based on the labeled descriptions. Our method for identifying misclassified apps is promising, as it found misclassified apps in the Weather and Medical categories. Moreover, our LDA based approach determined 236 possible subcategories of apps on GPS for a total of 9,265 apps. Our proposed LDA classifier out-performed LDA based classifiers applied in existing works (Sect. 4.3). Moreover, removing misclassified apps from the proposed data set, LDA can substantially improve classification process achieving a precision of 0.614.

For future work, a closer examination of the formed LDA topics is necessary to observe the fine-grained categorization of apps published on the markets. This could suggest appropriate subcategories of apps, and markets can use this to improve existing classification recommendations to avoid misclassification.

References

1. Apple app store. www.apple.com/app-store/. Accessed 18 Jun 2023
2. Google play store. www.play.google.com/store. Accessed 18 Jun 2023
3. Al-Subaihin, A.A., et al.: Clustering mobile apps based on mined textual features. In: Proceedings of the 10th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement. ESEM 2016, Association for Computing Machinery, New York, NY, USA (2016). https://doi.org/10.1145/2961111.2962600
4. Al-Subaihin, A., Sarro, F., Black, S., et al.: Empirical comparison of text-based mobile apps similarity measurement techniques. Empir. Softw. Eng. **24**(6), 3290–3315 (2019). https://doi.org/10.1007/s10664-019-09726-5
5. Alcic, S., Conrad, S.: Page segmentation by web content clustering. In: Proceedings of the International Conference on Web Intelligence, Mining and Semantics. WIMS 2011, Association for Computing Machinery, New York, NY, USA (2011). https://doi.org/10.1145/1988688.1988717
6. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. J. Mach. Learn. Res. **3**, 993–1022 (2003)
7. Bunyamin, H., Sulistiani, L.: Automatic topic clustering using latent dirichlet allocation with skip-gram model on final project abstracts. In: 2017 21st International Computer Science and Engineering Conference (ICSEC), pp. 1–5 (2017). https://doi.org/10.1109/ICSEC.2017.8443795
8. Ceci, L.: Number of available apps in the apple app store from 2008 to July 2022 (2023). www.statista.com/statistics/268251/number-of-apps-in-the-itunes-app-store-since-2008/. Accessed 18 Jun 2023
9. Chang, J., Boyd-Graber, J., Gerrish, S., Wang, C., Blei, D.: Reading tea leaves: How humans interpret topic models. In: Advances in Neural Information Processing Systems 22 (NIPS 2009), vol. 32, pp. 288–296 (2009)
10. Corley, C., Mihalcea, R.: Measuring the semantic similarity of texts. In: Proceedings of the ACL Workshop on Empirical Modeling of Semantic Equivalence and Entailment, pp. 13–18. Association for Computational Linguistics, Ann Arbor, Michigan (2005). www.aclanthology.org/W05-1203
11. Ebrahimi, F., Tushev, M., Mahmoud, A.: Classifying mobile applications using word embeddings. ACM Trans. Softw. Eng. Methodol. **31**, 1–30 (2021). https://doi.org/10.1145/3474827
12. Harman, M., Jia, Y., Zhang, Y.: App store mining and analysis: MSR for app stores. In: 2012 9th IEEE Working Conference on Mining Software Repositories (MSR), pp. 108–111 (2012). https://doi.org/10.1109/MSR.2012.6224306

13. Lavid Ben Lulu, D., Kuflik, T.: Functionality-based clustering using short textual description: helping users to find apps installed on their mobile device. In: Proceedings of the 2013 International Conference on Intelligent User Interfaces, pp. 297–306. IUI 2013, Association for Computing Machinery, New York, NY, USA (2013).https://doi.org/10.1145/2449396.2449434
14. Mikolov, T., Sutskever, I., Chen, K., Corrado, G., Dean, J.: Distributed representations of words and phrases and their compositionality (2013). https://doi.org/10.48550/ARXIV.1310.4546
15. Müllner, D.: Modern hierarchical, agglomerative clustering algorithms (2011)
16. Mokarizadeh, S., Rahman, M., Matskin, M.: Mining and analysis of apps in google play. In: Proceedings of the 9th International Conference on Web Information Systems and Technologies (BA-2013), pp. 527–535 (2013)
17. Sparck Jones, K.: A Statistical Interpretation of Term Specificity and Its Application in Retrieval, pp. 132–142. Taylor Graham Publishing, GBR (1988)
18. Store, A.A.: Choosing a category. www.developer.apple.com/app-store/categories/. Accessed 18 Jun 2023
19. Store, G.P.: Choose a category and tags for your app or game. www.support.google.com/googleplay/android-developer/answer/9859673?hl=en. Accessed 18 Jun 2023
20. Vakulenko, S., Müller, O., vom Brocke, J.: Enriching iTunes app store categories via topic modeling. In: International Conference on Information Systems (ICIS) (2014)
21. Wu, Z., Palmer, M.: Verb semantics and lexical selection. In: 32nd Annual Meeting of the Association for Computational Linguistics, pp. 133–138. Association for Computational Linguistics, Las Cruces, New Mexico, USA (1994). https://doi.org/10.3115/981732.981751www.aclanthology.org/P94-1019