Enriching iTunes App Store Categories via Topic Modeling

Research-in-Progress

Svitlana Vakulenko
University of Liechtenstein, and
MODUL University Vienna
Vienna, Austria
svitlana.vakulenko@modul.ac.at

Oliver Müller
University of Liechtenstein
Vaduz, Liechtenstein
oliver.mueller@uni.li

Jan vom Brocke
University of Liechtenstein
Vaduz, Liechtenstein
jan.vom.brocke@uni.li

Abstract

Mobile application development is an emerging lucrative and fast growing market. With the steady growth of the number of apps in the repositories the providers will inevitably face the need to fine-grain the existing hierarchy of categories used to organize the apps. In this paper we present a method to bootstrap the categorization process via topic modeling. We apply Latent Dirichlet Allocation (LDA) to the textual descriptions of iTunes apps in order to identify recurrent topics in the collection. We evaluate and discuss the results obtained from training the model on a set of almost 600,000 English-language app descriptions. Our results demonstrate that automated categorization via LDA-based topic modeling is a promising approach, that can help to structure, analyze and manage the content of app repositories. The topics produced complement the original iTunes categories, concretize and extend them by providing insights into the underlying category content.

Keywords: Text mining, Electronic marketplace, Content analysis, Topic Modeling

Introduction

Software applications ("apps") for mobile platform ecosystems (e.g., Apple iOS, Android) are among the fastest growing consumer product categories in the history of commerce (Kajanan et al. 2012). Both, Apple's iTunes App Store and Google's Play Store now – only 6 years respectively 5 years after their opening – host more than 1,000,000 different apps and the number of apps offered is growing at a rate of 4-7% per month (Datta et al. 2011). According to Gartner, the total revenues from app sales in 2013 were estimated at $26 billion, up from $18 billion in 2012, and consumers will download 179 billion apps in 2015 (Gartner 2013).

The rapid growth of mobile app markets creates challenges for consumers, developers, and marketplace providers. As the number of apps offered is constantly rising, it becomes more and more difficult for consumers to gain an overview of the overall app market, explore a specific app store, and discover apps that they need or like. Apple, for example, has organized its app web store into 66 categories and displays only the 240 most popular apps per category.[1] Consequently, a user can browse only through 15,840 of the more than 1,000,000 apps hosted on the marketplace. This also represents a serious challenge for developers, as their app will only be displayed to potential consumers if it reaches the top-240 of a category (Kajanan et al. 2012). As the number of apps offered and the number of app downloads are not uniformly distributed across categories, placing an app into one of the predefined categories offered by the marketplace provider is a strategic decision that might have far-reaching consequences. Finally, from a marketplace provider perspective, creating and continuously maintaining the app categorization scheme and surveying and verifying the correct mapping of apps to categories is a costly process (in terms of time and financial resources spent for establishing and maintaining the categorization schema). It is especially relevant nowadays as the app market is still evolving and new categories of apps are appearing and old ones are disappearing (Nickerson et al. 2009). In addition, the quality of the categorization scheme is likely to have an effect on the number of app downloads and, in turn, on revenues, as an irrelevant or hard to use categorization scheme will lead to high search costs and frustration on the user side.

In this research-in-progress paper we propose an alternative to the manual design of categorization schemes for mobile app markets. In particular, we try to tackle three limitations of existing categorization approaches. First, the design of categorization schemes is a complex and *costly* process, that involves domain experts who have to identify distinguishing characteristics of items, group characteristics into categories, and assign items to categories (Bailey 1994; Nickerson et al. 2009). Second, a manual categorization design process, even if it is empirically grounded, is always biased by the mental models of the designers and is, therefore, inherently *subjective*. This is especially the case when categories have to be defined before the actual content of a repository is known (note that apps are developed by hundreds of thousands of developers and not by the app store provider). Last, the resulting categorizations are *static*, meaning that in order to reflect changes in the content of the repository over time – when new categories emerge or existing categories decline – the categorization scheme has to be repeatedly updated.

As an alternative to the traditional category development process we advocate the application of topic modeling. Topic modeling algorithms are statistical methods that analyze the words of texts in large document collections to uncover latent topics that are inherent in the overall collection and to annotate documents with topic labels (Blei 2012). Furthermore, with topic models it is possible to discover how topics are interrelated (Blei and Lafferty 2007), how topics change over time (Blei and Lafferty 2006), and how authors are related to topics (Rosen-Zvi et al. 2004). Topic modeling algorithms do not require human intervention or prior labeling of documents, which allows a *cheap*, *unbiased* (or objective, solely based on the word statistics), and *repeatable* analysis of documents.

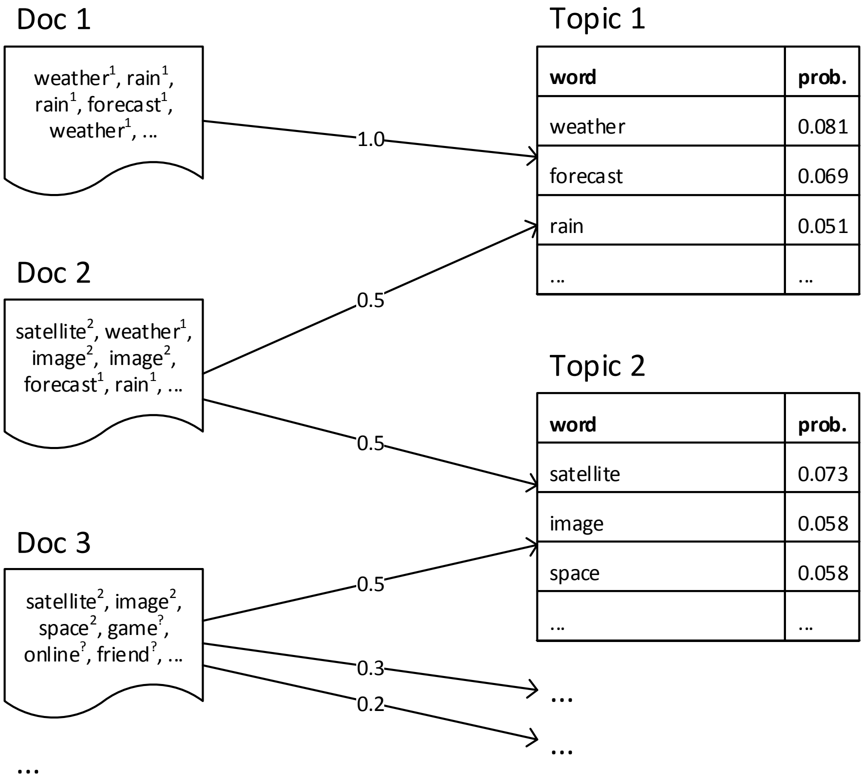
The remainder of this paper is structured as follows. In the next section we provide background on probabilistic topic modeling, especially Latent Dirichlet Allocation (LDA), and explain how we have used LDA to analyze the iTunes App Store. In Section 2 we present the results of applying LDA to almost 600,000 apps from the iTunes App Store by showing the topics our empirical analysis uncovered and comparing and contrasting these topics to the original categories of the iTunes App Store. We close with a brief discussion of related work and limitations and implications of our research.

[1] https://itunes.apple.com/us/genre/ios/id36?mt=8

Method

Topic Modeling

The core idea behind Latent Dirichlet Allocation (LDA), and probabilistic topic models in general, is a generative process that assumes that authors write documents by first choosing a mix of topics to write about and then by drawing words from the typical vocabulary of each of the selected topics. Accordingly, LDA assumes that documents, represented as bags of words, exhibit different topics in different proportions (cf., Figure 1). For example, in Figure 1 Document 2 is half about Topic 1 (50%) and half about Topic 2 (50%), while Document 1 is only about Topic 1 (100%). Each topic is represented as a probability distribution over a controlled vocabulary, usually all the words appearing in the document collection. In our example, Topic 1 has words like "weather" (8.1%), "forecast" (6.9%), and "rain" (5.1%) with high probability and Topic 2 has words like "satellite" (7,3%), "image" (5,8%), and "space" (5,8%) with high probability. Given this information, we could label Topic 1 as "weather forecasting" and Topic 2 as "satellite imaging". Consequently, we could say that Document 1 is purely about "weather forecasting", while and Document 2 is a mix of the "weather forecasting" and the "satellite imaging" topics.


Doc 1
weather^{1}, rain^{1}, rain^{1}, forecast^{1}, weather^{1}, ...
Doc 2
satellite^{2}, weather^{1}, image^{2}, image^{2}, forecast^{1}, rain^{1}, ...
Doc 3
satellite^{2}, image^{2}, space^{2}, game$^{?}$, online$^{?}$, friend$^{?}$, ...
...
1.0
0.5
0.5
0.5
0.3
0.2
...
...
Topic 1
Topic 2

Topic 1

word	prob.
weather	0.081
forecast	0.069
rain	0.051
...	...

Topic 2

word	prob.
satellite	0.073
image	0.058
space	0.058
...	...

Figure 1: Illustration of the generative process underlying probabilistic topic models

In reality, the only variables a reader of a collection of documents can observe are the words of the documents, all other variables (i.e., the topic distributions for each document and the word distributions for each topic) are hidden. The goal of LDA is to infer these hidden distributions, given the observed words per document (for a more detailed explanation refer to Blei, Ng, & Jordan, 2003).

In recent years, topic modeling via LDA, or its predecessor Latent Semantic Analysis (LSA), experienced increasing popularity as a research method for the quantitative analysis of qualitative data. In the IS discipline, topic modeling has been used, for instance, for content analysis of academic papers (Sidorova et al. 2008), social media posts (Evangelopoulos and Visinescu 2012), job advertisement (Müller et al. 2014), sustainability reports (Reuter et al. 2014), vendor case studies (Herbst et al. 2014), and customer feedback (Coussement and Poel 2008).

Data Collection and Analysis

In this research, we used a snapshot of the iTunes App Store taken in October 2013. We downloaded the app data using a custom Python script that crawls the web interface of the iTunes App Store[2] and collects the IDs of all apps. Further on, the script requests the app metadata using these IDs from the official iTunes Search API[3]. In this way, we discovered more than 700,000 iPhone apps and obtained their metadata. The metadata contains name of the app, price, description, content advisory rating, release date, primary category, a list of secondary categories, name of the developer, and name of the distributor. Although each of these fields may contain valuable data for categorizing apps, we limited our topic modeling to the app description field only. The app descriptions can be written in different languages; in order to be able to interpret the results produced by the LDA algorithm, we filtered out all non-English-language descriptions (18,4% of apps) using the Google Compact Language Detector[4]. Finally, we used the Python Natural Language Toolkit (NLTK) to perform standard natural language preprocessing (e.g., tokenization, stop-word removal) of the app descriptions. After preprocessing, the final dataset which we used for all further analysis reported in this paper comprised almost 600,000 English-language app descriptions.

We performed topic modeling using the LDA implementation of the Gensim library (Rehurek and Sojka 2010). We used the standard parameters provided by Gensim (alpha='symmetric', eta=None, decay=0.5, eval_every=10, iterations=50, gamma_threshold=0.001, update_every=1). Due to the large number of documents we increased the default chunksize to 10,000 documents and the number of training passes through the collection to 2.

The number of topics was set to 66 which correspond to the number of app categories currently provided by the iTunes App Store (23 top-level categories + 16 Games sub-categories + 27 Newsstand sub-categories). Such an experimental design allows us to use the existing categorization scheme of iTunes as the ground truth for evaluating the approach proposed by us. This way, we are able to compare and contrast our results with the original categories assigned by Apple.

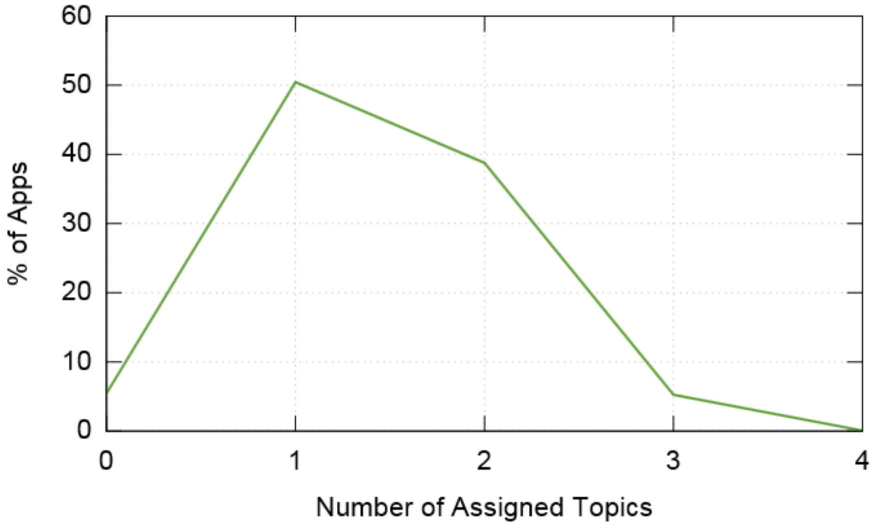


Figure 2: Distribution of assigned topics

LDA produces a soft clustering of documents, that is, a probabilistic distribution of topics is assignment to each document. To go from this soft clustering to a hard clustering we set a probability threshold of 0.2.

[2] https://itunes.apple.com/us/genre/ios/id36?mt=8

[3] https://www.apple.com/itunes/affiliates/resources/documentation/itunes-store-web-service-search-api.html

[4] https://code.google.com/p/chromium-compact-language-detector/

In other words, topics with a probability of less than 20% for a given document were disregarded. This resulted in the assignment of one or two topics per app (cf., Figure 2). Some of the apps remained uncategorized (5%) and about 5% of the apps were assigned to 3 or 4 topics.

Evaluation

Topic model evaluation remains an open research problem. Many researchers have stated that the standard measures of perplexity and held-out likelihood are misleading when evaluating topic models meant for exploratory analysis that need to be understandable by humans (Chang et al. 2009; Newman et al. 2010; Wallach and Murray 2009). Therefore, they advocate the need of human evaluation of topic models or for comparing topic models against existing gold standard categorization schemes.

In this work we took the existing categorization scheme of iTunes as the ground truth and used it for the evaluation of the LDA model. Our assumption behind this approach was that the iTunes categories were manually created and populated by domain experts. This makes iTunes category assignment a reputable and a high-quality candidate for the gold standard. In reality, however, the original categorization does not necessarily provide the best or the only way to organize the collection. Therefore, we take a closer look at the differences between our results and the ground truth in order to qualitatively evaluate whether they contain threats or opportunities. By threats we mean errors (messy, useless or incorrect data). By opportunities we mean insights that provide new information about the collection and cast light on its trends.

In order to quantitatively evaluate the quality of the extracted LDA topics, we calculated the overlap between latent topics generated by LDA and categories assigned by Apple by counting for every unique category-topic combination the number of apps assigned to both of them simultaneously. In order to gauge this overlap with respect to the sizes of the two sets, we calculated the overlap coefficient (Charikar 2002) as follows:

$$k_{ab} = \frac{|A \cap B|}{min(|A|,|B|)}$$

, where k_{ab} is the overlap coefficient for category *a* and topic *b*; A is the set of all apps assigned to category *a;* B is the set of all apps assigned to topic *b*; $|A \cap B|$ is the size of the overlap between the app sets assigned to category *a* and topic *b* (in number of apps). In the following, we will only report on overlaps that exceed a threshold level for the overlap coefficient of 0.3 (mean = 0.063; SD = 0.136). This threshold captures nearly 50% of the area under the distribution curve (cf., Figure 3).

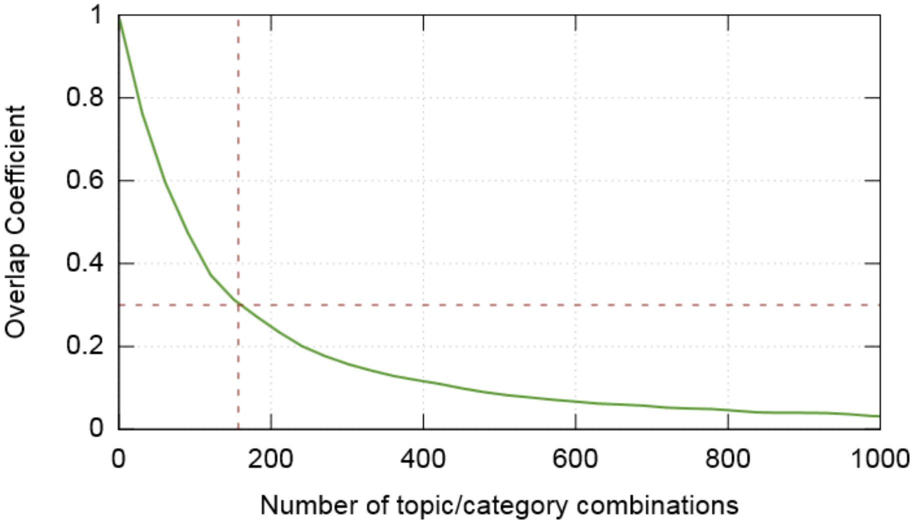


Figure 3: Distribution of the overlap coefficient

We also calculate the overlap coefficients with respect to each of the two sets separately: $ka_{ab} = \frac{|A \cap B|}{|A|}$ and $kb_{ab} = \frac{|A \cap B|}{|B|}$ in order to identify the relationship between the latent topics generated by LDA and the categories assigned by Apple. If $ka_{ab} > kb_{ab}$, we call *a* the sub-category of the topic b. Otherwise, if $ka_{ab} < kb_{ab}$ *a* is the parent category of topic *b*. For example: *a*='Games'; *b*='0.153*puzzle + 0.045*piece + 0.028*solve + 0.014*level'; $ka_{ab} = 0.044$; $kb_{ab} = 0.908$. $ka_{ab} < kb_{ab}$, therefore *a (Games)* is identified as the parent category for topic *b (puzzle piece solve)*.

Results

For illustration purposes, some results of our empirical analysis are summarized in Table 1. Due to space limitations we report here only the top-20 topics (the complete analysis along the enhanced visualization is available online at: https://github.com/vendi12/whatsIn). The content of Table 1 is sorted descending by the number of apps assigned to the topic. For each extracted latent LDA topic, the table displays the top-5 words along with their probabilities, the number of apps assigned to the topic, and the relation to the original iTunes categories (parent category, sub-category).

Table 1. Latent Topics, Related iTunes Categories, and Number of Apps

	LDA Topic	# Apps	iTunes Parent Categories	iTunes Sub-categories
1	0.110*game + 0.032*play + 0.024*level + 0.018*player	83,476	Games, Entertainment	Arcade, Action, etc.[5]
2	0.051*news + 0.037*latest + 0.023*event + 0.022*access	37,771		News
3	0.020*game + 0.012*enemy + 0.010*world + 0.009*battle	35,553	Games, Entertainment	Arcade, Action, Adventure, Strategy, Role Playing
4	0.019*list + 0.019*email + 0.018*note + 0.015*data	35,015	Utilities, Productivity	
5	0.022*business + 0.016*product + 0.013*service + 0.012*information	33,360	Business	Professional & Trade, Business & Investing
6	0.061*child + 0.038*kid + 0.020*story + 0.016*learn	32,731	Education, Games	Educational
7	0.034*like + 0.026*want + 0.022*know + 0.019*love	31,327	Entertainment	
8	0.024*help + 0.023*time + 0.019*make + 0.018*need	30,935		
9	0.122*photo + 0.042*picture + 0.031*image + 0.029*camera	29,697	Entertainment	Photo & Video

[5] 14 out of 18 subcategories of *Games* (excluding *Educational*, *Music*, *Role Playing* and *Sports*).

	LDA Topic	# Apps	iTunes Parent Categories	iTunes Sub-categories
10	0.024*screen + 0.024*iphone + 0.020*text + 0.018*touch	29,330	Entertainment	
11	0.047*search + 0.037*find + 0.028*event + 0.026*view	28,323	Lifestyle	
12	0.071*video + 0.054*share + 0.048*friend + 0.047*facebook	22,126	Social Networking, Entertainment	
13	0.054*application + 0.053*user + 0.041*iphone + 0.018*ipad	22,022	Business	
14	0.054*time + 0.039*button + 0.016*screen + 0.015*timer	20,946	Utilities	
15	0.022*learn + 0.016*video + 0.015*student + 0.013*learning	16,499	Education	
16	0.030*calculator + 0.018*value + 0.018*calculate + 0.016*calculation	16,269	Utilities	
17	0.072*question + 0.051*test + 0.037*answer + 0.024*quiz	15,705	Education	
18	0.028*order + 0.021*restaurant + 0.018*vehicle + 0.017*deal	14,844	Lifestyle	Food & Drink
19	0.044*location + 0.031*map + 0.030*data + 0.022*offline	14,647	Navigation, Travel	
20	0.067*number + 0.016*shape + 0.016*memory + 0.016*match	14,643	Puzzle, Education, Games, Entertainment	

Similar to the iTunes category structure LDA identified games to be the most popular type of apps in the repository and produced 14 game-related topics. Some of them resemble the iTunes subcategories, e.g. *word letter learn* (*Word), race racing control (Racing), card flash rule (Card), slot machine meeting (Casino).* Others fuse several categories together, e.g. *child kid story* (*Education*, *Educational*, *Games*), *game enemy world* (*Arcade, Action, Adventure,* etc.), *number shape memory* (*Education*, *Entertainment*, *Games*, *Puzzle*).

The major difference between the categorization and topic modeling schemes is that LDA did not produce *Newsstand* subcategories (*Outdoors & Nature, Science, Teens,* etc.) but aggregated them into one major topic: '0.068*subscription + 0.044*issue + 0.034*current + 0.030*magazine'. On the other hand, LDA generated more topics identified as subsets of the *Entertainment* category (25 topics). This category is the most populated iTunes category and comprises 165,610 apps or 28% of the total number of apps. In contrast, the *Newsstand* category is relatively small 6,608 apps (only 1%).

The quantitative representation of the relations between the topics and iTunes categories (the overlap) is shown on the correspondence chart (cf., Figure 4). The correspondence chart demonstrates the overlap relations in a form of a matrix filled with bubbles on the intersection of a topic and a category. Size of the bubbles corresponds to the size of the overlap ($|A \cap B|$). The topic labels here are automatically generated through merging of the top-3 high-loaded terms. The design of this plot is inspired by the correspondence chart from (Chuang, Gupta, et al. 2013).

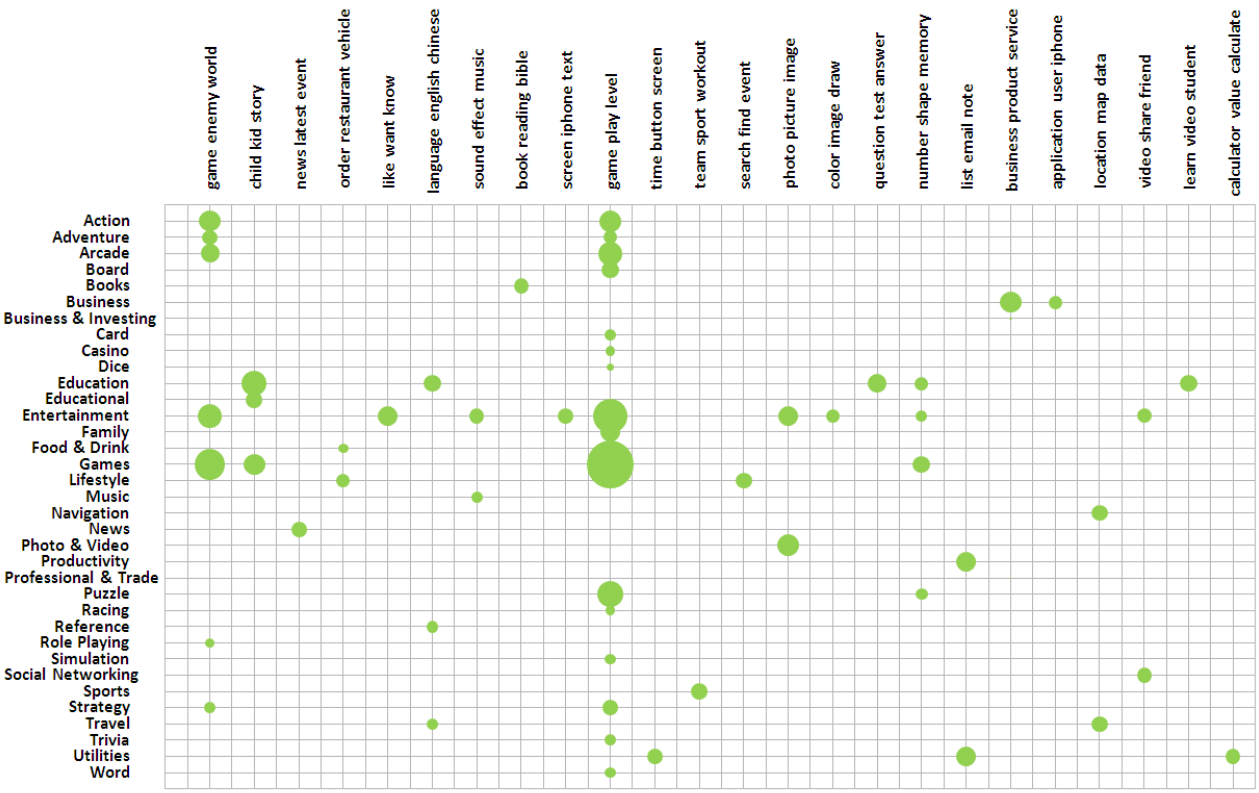


Figure 4: Correspondence chart showing the overlap of LDA topics and iTunes categories

The set of keywords identifying the topic cast light on the actual content of the repository and often are more descriptive and detailed than the general categories provided by iTunes. The topic modeling approach is based on the word counts; therefore, the top keywords demonstrate also the frequent co-occurrences. Thereby, we learn some curious facts about our app collection, such as:

- Bible is the most popular book in iTunes
  ('0.049*book + 0.019*reading + 0.013*bible + 0.011*story');
- English and Chinese are the most popular languages for studying and traveling
  ('0.047*language + 0.044*english + 0.025*chinese + 0.025*word + 0.023*phrase + 0.021*dictionary + 0.016*spanish + 0.014*learn + 0.012*translation + 0.012*japanese').

Related Work

Our research is related to work on software repository mining (Hassan 2008). Topic models have already been applied to mining of the source code hosted in the large software repositories (Thomas 2011), yet unlike traditional software repositories (e.g., Github, SourceForge) the mobile app stores do not host the source code but rather the already precompiled applications.

Harman et al. (2012) were among the firsts to apply text mining techniques on descriptions of mobile apps. They analyzed the textual descriptions, categories, customer ratings, download rankings, and prices of more than 30,000 Blackberry apps using basic natural language processing techniques and found strong correlations between an app’s rating and ranking, but no correlation between price and ranking nor between price and rating. Comparing the work of Harman et al. to our research, we see a different focus; while Harman et al. examine the economics of mobile app markets, we are aiming at developing more effective and more efficient categorization schemes to support consumers in exploring mobile app markets.

More recently, Gorla et al. (2014) applied LDA on 22,521 Android app descriptions collected from the Google Play Store in order to detect malicious apps. More specifically, they automatically compared textual app descriptions with the Android APIs an app requires access to. Apps with discrepancies between description and API access, for example, a weather app demanding access to the SMS services, were flagged as potentially malicious. The technical approach of Gorla et al. is similar to ours; however, Gorla et al. perform LDA-based topic modeling for security reasons, while we are interested in providing better user interfaces via topic modeling.

Conclusion

Results of our experiments show that the method proposed in this paper provides a handy solution for analyzing content of the repository and delivers valuable insights. In particular, it identified new topics absent from the original categorization, e.g. educational games for children, radio stations, timers, calculators, quizzes, applications for mobile banking, document management and sound effects.

The method proposed in this paper fully automates the categorization process. Nevertheless, it is beneficial to include a human evaluator, a domain expert, who could analyze the results returned by the algorithm and efficiently summarize them through manual labeling of the generated topics.

Thus, our approach is applicable in the two following scenarios:

- In case of a new repository with uncategorized content, assists in developing an efficient categorization schema and automates assignment of the new items to the schema;
- Supports evaluation, further improvement and refinement of existing categorization schema.

In essence, the topics produced with LDA are based on a considerable number of apps that share a common vocabulary in the description of their functionality, i.e. word statistics. This fact defines also the limitations of the proposed approach, namely the analysis is limited to the content of the app descriptions. Therefore, in case the provided description is incomplete, obscure or deceitful, the algorithm will return unsatisfactory results. However, the only way to overcome this limitation is to analyze the source code of the app, which is not always accessible.

Moreover, the LDA algorithm produces a flat topic model, while the original app categories are organized into a hierarchy. Some advanced algorithms, e.g. Hierarchical LDA (Blei, Griffiths, et al. 2003), are designed to overcome this limitation. However, they are still in the development phase and were not yet widely adopted by the community. A semi-automated solution suggests to start with the flat topic model and then organize it into a topic hierarchy manually which can be a rather trivial task for a domain expert. It can be combined with the topic evaluation and labeling steps as well. We would look into extending our approach towards building a hierarchy of topics in the future work.

In the future work we would like to experiment with adding more topics to the model to achieve an even more fine-grained solution. We also plan to generate other categories from the texts of the descriptions that employ different basis for categorization. One candidate method for that is Named Entity Recognition (NER, (Nadeau and Sekine 2007)), which allows for extraction of named entities, e.g. names of persons, companies, countries, titles of books and albums.

Furthermore, we will perform correlation analysis between the various categories and other metadata in order to spot interesting patterns and dependencies (Harman et al. 2012). We will also try to detect trends describing evolution of the market place over time (Blei and Lafferty 2006; Wang and McCallum 2006). Application of this method to the Google Play Store will allow us to cover virtually the whole apps market and compare the level of development and competition across the two major app stores.

References

Bailey, K. 1994. *Typologies and taxonomies: an introduction to classification techniques*, Sage Publications, Inc.

Blei, D. 2012. “Probabilistic topic models,” *Communications of the ACM* (55:4), pp. 77–84.

Blei, D., Griffiths, T., Jordan, M., and Tenenbaum, J. 2003. “Hierarchical Topic Models and the Nested Chinese Restaurant Process,” in *Advances in Neural Information Processing Systems*, Vancouver, pp. 1–8.

Blei, D., and Lafferty, J. 2006. “Dynamic topic models,” in *International conference on Machine Learning*, New York, pp. 113–120.

Blei, D., and Lafferty, J. 2007. “A correlated topic model of science,” *The Annals of Applied Statistics* (1:1), pp. 17–35.

Blei, D., Ng, A., and Jordan, M. 2003. “Latent dirichlet allocation,” *The Journal of Machine Learning Research* (3), pp. 993–1022.

Chang, J., Boyd-Graber, J., Gerrish, S., Wang, C., and Blei, D. 2009. “Reading tea leaves: How humans interpret topic models,” in *Proceedings of the Advances in Neural Information Processing Systems Conference*, Vancouver, pp. 1–9.

Charikar, M. 2002. “Similarity estimation techniques from rounding algorithms,” in *Annual ACM Symposium on Theory of Computing*, New York, pp. 380–388.

Chuang J., Gupta S., Manning C. D., H. J. 2013. “Topic Model Diagnostics: Assessing Domain Relevance via Topical Alignment,” *International conference on Machine Learning*.

Coussement, K., and Poel, D. Van den. 2008. “Improving customer complaint management by automatic email classification using linguistic style features as predictors,” *Decision Support Systems* (44:4), pp. 870–882.

Datta, A., Dutta, K., Kajanan, S., and Pervin, N. 2011. “Mobilewalla: A mobile application search engine,” in *Mobile Computing, Applications, and Services*, Los Angeles, pp. 172–187.

Evangelopoulos, N., and Visinescu, L. 2012. “Text-mining the voice of the people,” *Communications of the ACM* (55:2), pp. 62–69.

Gartner. 2013. “Gartner Says Mobile App Stores Will See Annual Downloads Reach 102 Billion in 2013,” http://www.gartner.com/newsroom/id/2592315 [Accessed 2014-09-07].

Gorla, A., Tavecchia, I., Gross, F., and Zeller, A. 2014. “Checking App Behavior Against App Descriptions,” in *International Conference on Software Engineering*, Hyderabad, pp. 1–11.

Harman, M., Jia, Y., and Zhang, Y. 2012. “App store mining and analysis: MSR for app stores,” in *IEEE Working Conference on Mining Software Repositories*, Zürich, pp. 108–111.

Hassan, A. 2008. “The road ahead for mining software repositories,” in *Frontiers of Software Maintenance*, Beijing.

Herbst, A., Simons, A., vom Brocke, J., Müller, O., Debortoli, S., and Vakulenko, S. 2014. "Identifying and Characterizing Topics in Enterprise Content Management: A Latent Semantic Analysis of Vendor Case Studies," in *Proceedings of the 22nd European Conference on Information Systems*, Tel Aviv.

Kajanan, S., Pervin, N., Ramasubbu, N., Dutta, K., and Datta, A. 2012. "Takeoff and Sustained Success of Apps in Hypercompetitive Mobile Platform Ecosystems: An Empirical Analysis," in *International Conference on Information Systems*, Orlando.

Müller, O., Schmiedel, T., Gorbacheva, E., and vom Brocke, J. 2014. "Toward a Typology of Business Process Management Professionals: Identifying Patterns of Competence through Latent Semantic Analysis," *Enterprise Information Systems* (in press), pp. 1–31.

Nadeau, D., and Sekine, S. 2007. "A survey of named entity recognition and classification," *Lingvisticae Investigationes* (30:1), pp. 3–26.

Newman, D., Noh, Y., and Talley, E. 2010. "Evaluating topic models for digital libraries," in *Proceedings of the 10th annual joint conference on Digital libraries*, Gold Coast.

Nickerson, R., Muntermann, J., Varshney, U., and Isaac, H. 2009. "Taxonomy development in information systems: Developing a taxonomy of mobile applications," in *European Conference on Information Systems*, Verona.

Rehurek, R., and Sojka, P. 2010. "Software Framework for Topic Modelling with Large Corpora," in *LREC 2010 Workshop on New Challenges for NLP Frameworks*, Valletta, pp. 46–50.

Reuter, N., Vakulenko, S., vom Brocke, J., Debortoli, S., and Müller, O. 2014. "The Role of Information Systems in Achieving Energy-related Environmental Sustainability," in *European Conference on Information Systems*, Tel Aviv.

Rosen-Zvi, M., Griffiths, T., Steyvers, M., and Smyth, P. 2004. "The author-topic model for authors and documents," in *Conference on Uncertainty in Artificial Intelligence*, Arlington, pp. 487–494.

Sidorova, A., Evangelopoulos, N., Valacich, J. S., and Ramakrishnan, T. 2008. "Uncovering the intellectual core of the information systems discipline," *MIS Quarterly* (32:3), pp. 467–482.

Thomas, S. 2011. "Mining software repositories using topic models," in *International Conference on Software Engineering*, New York, pp. 1138–1139.

Wallach, H., and Murray, I. 2009. "Evaluation methods for topic models," in *Proceedings of the 26th Annual International Conference on Machine Learning*, Montreal.

Wang, X., and McCallum, A. 2006. "Topics over time," in *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining,* New York, p. 424.