REPLICATION of Research Paper

Title: An empirical study on release notes patterns of popular apps in the Google Play Store

Aidan Z. H. Yang · Safwat Hassan · Ying Zou · Ahmed E. Hassan

Paper Replication-friendliness description:
- App dataset collected from Google PlayStore isn't made available for replication. (However, the process of app selection is described well. Customization of available crawlers, websites, packages will help in collecting data)
- Supporting code for the performed analysis isn't made available for replication or validation. (However, the algorithms / methods used for calculation, validation are described well. Most referenced official resources available online on web will help in coding)
- Machine Learning models like Clustering, Logistic Regression used for analysis would completely depend on the trained data. Hence, the results obtained would be just similar but not same as the original paper described.

I.   MOTIVATION

Verifying the motivation described behind the research paper.
SO questions related to release notes:
https://stackoverflow.com/questions/638423/how-should-release-notes-be-written
https://stackoverflow.com/questions/56774669/is-there-any-way-to-order-group-work-item-type-in-azure-devops-release-notes

SO answers referring to release note contents:
1)https://stackoverflow.com/questions/32099530/google-vision-barcode-library-not-found/38100068#38100068

> yeah, according to the release notes the bug should be fixed: developers.google.com/vision/release-notes#bug_fixes – Wirling

2)https://stackoverflow.com/questions/5537993/how-to-use-jquery-mobile-and-backbone-js-together-in-one-app/5835215#5835215

3)https://stackoverflow.com/questions/28376120/messagedlg-in-delphi-xe7-android/28376893#28376893

Read GitHub survey article on medium
https://medium.com/@scottydocs/do-people-actually-read-release-notes-449d099d73ee
https://uxdesign.cc/design-better-release-notes-3e8c8c785231

II.   DATA COLLECTION

Attempt-1: The app analytics and app data industry standard. https://www.appannie.com/ . Top apps according to #The active users filter for each category was pulled manually as a report, but for scraping other metadata information, we need app-id. Hence, this work was discarded.

Attempt-2: Few python libraries like `google-api-python-client`, `google-play-scraper` were tried, but they couldn't fetch the "release notes" or "what's new" section for all release versions. Few websites like app-tweak were tried, however they do not have "release notes" and are paid websites.

Attempt-3: Google Play Developer API was tried, however it required a 'Google API Console' developer account which is a paid version.

Attempt-4: Successful attempt.
Selected top 30 applications for each category of apps from Google Play store.
Respective 'release notes' were collected from free unofficial APIs(like https://apkpure.com/, https://www.apkmirror.com/apk). Octoparse was used to scrape data in required format. Few applications did not have data in APIs, hence we have collected the data from the app store to proceed with actual replication.
App metadata, app reviews were collected using a python script written using the `google-play-scraper` library.
Further analysis is applied on 30 "automobile" applications.

III. CODE

1) CODE AVAILABLE

Data collection

Fetch app names:

https://github.com/Jahnavi-K/Replication_ESR22RelPatterns/blob/main/1_fetch_app_names.ipynb

Fetch app reviews:

https://github.com/Jahnavi-K/Replication_ESR22RelPatterns/blob/main/2_fetch_app_reviews.ipynb

Empirical study analysis

https://github.com/Jahnavi-K/Replication_ESR22RelPatterns/blob/main/Replication_RelNotes_v6.ipynb

2) EXECUTION REQUIREMENTS

- Anaconda – Jupyter Notebook or VSCode is required to run the .ipynb files.
- Download Stanford CoreNLP Lementizer
  https://stanfordnlp.github.io/CoreNLP/index.html#download
- Start the Lementizer server:
  cd <downloaded-file>

```
java -mx4g -cp "*" edu.stanford.nlp.pipeline.StanfordCoreNLPServer
-annotators "tokenize,ssplit,pos,lemma,parse,sentiment" -port 9000 -timeout
30000
```

IV. RESULT VALIDATION

Output data available:

https://github.com/Jahnavi-K/Replication_ESR22RelPatterns/tree/main/output

RQ1: What are the release notes update patterns?

Analysis of clusters

Cluster 4: Long non-updating steady: App 3, 4, 9, 11, 15, 16, 18, 20, 25, 27, 29, 30

Cluster 3: Short rising-updatability with major releases: App 6, 8, 12, 17

Cluster 5: Long updating steady: App 26

Cluster 2: Short updating steady: App 14, 19, 23

Cluster 6: Long rising-updatability with major releases: App 2, 5, 7, 13, 21, 24, 28

Cluster 1: Short non-updating steady: App 1, 10, 22

Summary: 50% of apps are non-updating. "bug", "improvements" and general words are used in non-updating patterns.

Manual analysis: 19 drivers for release:
- General bug fix : Pattern 1
- Detailed bug fix : Pattern 5
- General improvement: Pattern 3
- Detailed improvement: Pattern 5

- Performance: Pattern 4
- New feature: Pattern 5, 6
- Permission and security: Pattern 5
- Platform Device Support: Pattern 5
- UI: Pattern 6
- Localization: Pattern 5
- Crash: Pattern 5
- Resource utilization: Pattern 4
- Ask for feedback Advertisement: Pattern 4
- Non informative: Pattern 4
- Provide support: Pattern 5

RQ2: What are the characteristics of the apps that follow a certain release notes pattern?

Summary:
Patterns 1,4,6 are performing better than rest according to AUC of ROC value.

Wald statis- tics ($\chi 2$) to estimate the relative contribution of each app attribute:
Pattern 1: rel_freq, avg_rel_rating, improve_keywords, bug_keywords
Pattern 2: nan
Pattern 3: size_of_user_base, emergency_keywords, improve_keywords
Pattern 4: emergency_keywords, avg_rel_rating, same_day_rel
Pattern 5: size_of_user_base, bug_keywords, rel_freq
Pattern 6: size_of_user_base, emergency_keywords, avg_rel_rating, same_day_rel

Pattern-1: Tend to have a lower rating. Bug/improve keywords used. They mention generalized sentences.
Pattern-2: lower rating. Improvement keywords
Pattern-3: less downloads, higher emergency fixes.
Pattern-4: medium release ratings. Same day fixes are high, so emergency keywords are used.
Pattern-5: high release ratings. Not-frequently deployed
Pattern-6: high release ratings. Response rate is high.

RQ3: What causes developers to shift their release notes pattern?

Summary:
Apps 3, 5, 20, 23, 27 feature values have increased recently.

Manual analysis:
App 3: Description regarding services offered, new functionalities. Asking for feedback. --Not much of pattern shift
App 5: Short to Long shift. non-freq updatability to frq updatabity. How to use the fuctionalities is described.
App 20: Short to Long shift. non-freq updatability to frq updatabity. Describing impovements made in detail instead of general description.
App 23: Short to Long. BUT, specific to generic. Freq updatability to non-freq updatability.
App 27: Description regarding services offered, new functionalities. Asking for feedback. --Not much of pattern shift

To summarize:

Most apps switch from short to Long release note patterns.

66% of (pattern shifted) apps have shifted from rarelyy updated to frequently updated type.

i.e., Most of them have started giving instructions on usage to users, asking for feedback, swicthed from general description to specific details.

V.   REFERENCES

Original empirical study	https://link.springer.com/article/10.1007/s10664-021-10086-2
Stanford CoreNLP Lemmatizer	https://www.machinelearningplus.com/nlp/lemmatization-examples-python/
Read file from github for google collab	https://towardsdatascience.com/3-ways-to-load-csv-files-into-colab-7c14fcbdcb92
Cosine Similarity	https://studymachinelearning.com/cosine-similarity-text-similarity-metric/
Spearman Rank Correlation Coefficient	https://machinelearningmastery.com/how-to-calculate-nonparametric-rank-correlation-in-python/
Matplot in vscode	https://stackoverflow.com/questions/66121948/matplotlib-plots-not-showing-in-vs-code
Clustering K value detection - gap Statistical method, Silhouette score	https://medium.com/mlearning-ai/deciding-number-of-clusters-using-gap-statistics-davies-bouldin-index-calinski-harabasz-index-2ce9acfb6118
K-means map row with cluster	https://realpython.com/k-means-clustering-python/#understanding-the-k-means-algorithm