Jurnal Nasional Teknologi Komputer
Vol.2; No: 1; Januari 2022
Website: publikasi.hawari.id/index.php/jnastek
E-ISSN: 2808-4845; P-ISSN: 2808-7801

Makalah Penelitian

Prediksi Peringkat Aplikasi di Google Play Menggunakan Metode Random Forest

Bagiya Wahyudi[1], Ina Kuswandi[2]

[1,2]Fakultas Teknik dan Ilmu Komputer
Universitas Potensi Utama, Medan, Indonesia

Corresponding Author: Bagiya Wahyudi

ABSTRACT

Application developers and users are the keys to the market impact on application development. In application development, developers need to predict applications in the market accurately, accurate prediction results are very important in showing user ratings that affect the success of an application. Ratings are given by users to judge that the application is good or not. The higher the rating given by the user, it means that the user likes the application and can be a benchmark for other users to download the application. It is undeniable that there are so many apps available on the google play store, it is impossible for users to select one by one app on the google play store. Therefore, a rating prediction system is needed to determine the right application based on the rating given by the user to an application. Predictions will be made using the random forest algorithm as the method used to predict application ratings. This study using the Google Play Store dataset. This dataset has 10840 rows and 13 attributes. The results of this study can be seen from the use of the random forest algorithm with an average accuracy of 93.8%.

Keywords: Google Play Store, Rating, Prediction, Random Forest

ABSTRAK

Pengembang aplikasi dan pengguna adalah kunci dari dampak pasar pada pengembangan aplikasi. Dalam pengembangan aplikasi, pengembang perlu memprediksi aplikasi di pasar secara akurat, hasil prediksi yang akurat sangat penting dalam menunjukkan penilaian pengguna yang mempengaruhi keberhasilan suatu aplikasi. Rating diberikan oleh pengguna untuk menilai apakah aplikasi tersebut bagus atau tidak. Semakin tinggi rating yang diberikan oleh pengguna, berarti pengguna tersebut menyukai aplikasi tersebut dan dapat menjadi tolak ukur bagi pengguna lain untuk mendownload aplikasi tersebut. Tidak dapat disangkal bahwa begitu banyak aplikasi yang tersedia di google play store, tidak mungkin bagi pengguna untuk memilih satu per satu aplikasi di google play store. Oleh karena itu, diperlukan sistem prediksi rating untuk menentukan aplikasi yang tepat berdasarkan rating yang diberikan oleh pengguna terhadap suatu aplikasi. Prediksi akan dibuat menggunakan algoritma random forest sebagai metode yang digunakan untuk memprediksi rating aplikasi. Penelitian ini menggunakan dataset Google Play Store. Dataset ini memiliki 10840 baris dan 13 atribut. Hasil dari penelitian ini dapat dilihat dari penggunaan algoritma random forest dengan rata-rata akurasi sebesar 93,8%.

Keywords: Google Play Store, Rating, Prediksi, Random Forest

INTRODUCTION

Pada saat ini perkembangan teknologi berkembang sangat pesat salah satunya dalam bidang penyediaan informasi, teknologi informasi dapat digunakan untuk melengkapi data, dan umum digunakan sebagai dasar pengambilan keputusan. Di dalam Google Play Store terdapat informasi berupa deskripsi, komentar dari pengguna, dan rating mengenai aplikasi di dalamnya dengan tujuan untuk mengetahui kekurangan atau kelebihan dari aplikasi yang dibuat.

Pertumbuhan signifikan pasar aplikasi seluler berdampak besar pada teknologi digital dengan jumlah aplikasi yang tersedia di Google Play Store hingga Maret 2021 sekitar 2,8 juta dan akan terus bertambah seiring waktu (Appbrain, 2021). Pengembang dan pengguna aplikasi adalah kunci dari dampak pasar pada pengembangan aplikasi (Hengshu Zhu et al., 2014). Dalam mengembangkan aplikasi, pengembang perlu memprediksi aplikasi yang ada di pasaran secara akurat, karena hasil prediksi yang akurat sangat penting dalam menentukan pengembangan aplikasi di Google Play (Shen, Lu dan Hu, 2017). Pada tahun 2017, Hartmann-Boyce dkk melakukan tinjauan terhadap aplikasi Google Play Store untuk mengeksplorasi apa yang disukai dan tidak disukai pengguna tentang aplikasi penurunan berat badan dan pelacakan berat badan. Hasil penelitian Hartmann-Boyce et al menunjukkan bahwa penilaian pengguna mempengaruhi keberhasilan suatu aplikasi (Hartmann-Boyce et al., 2017). Peringkat aplikasi juga memengaruhi sistem rekomendasi populer aplikasi di pasar Google Play dengan kriteria menggunakan parameter kategori, jumlah pemasangan, peringkat, ulasan (Zhu et al., 2014).

Untuk memprediksi rating aplikasi, ada beberapa metode yang digunakan, seperti pada tahun 2017, Chen dkk membandingkan metode Logistic Model Tree (LMT), Random Forest (RF), dan Decision Tree (CART) untuk memprediksi kerawanan longsor. Hasil penelitian menunjukkan bahwa hasil perbandingan ketiga metode menghasilkan model random forest memiliki prediksi terbaik dibandingkan dengan model LMT atau CART dengan nilai Area Under Curve (UAC) sebesar 0,837 dan nilai akurasi prediksi sebesar 0,772. pada penelitian lain yang membandingkan Random Forest dengan K-Nearest Neighbors pada dataset HAR (human activity recognition), hasil perbandingan ini didapatkan hasil akurasi terbaik menggunakan metode Random forest dengan nilai 93,13% (Bindu, BhanuJyothi dan Suryanarayana, 2017). Pada penelitian lain dimana dilakukan perbandingan antara SVM yang digabungkan dengan classifier lain seperti BayesNet, AdaBoost, Logistics, IBK, J48, Random Forest, JRip, OneR, dan SimpleCart, hasil penelitian ini menemukan bahwa SVM yang dikombinasikan dengan Random Forest mendapatkan hasil yang baik. dengan skor 97,50% dibandingkan dengan menggunakan SVM saja dengan nilai 91,81% (Chand et al., 2016).

Berdasarkan uraian di atas, penelitian ini akan memprediksi rating aplikasi di Google Play menggunakan metode Random Forest sehingga diharapkan dapat membantu menemukan kelemahan aplikasi dalam waktu singkat dari sudut pandang pengguna sebagai bahan untuk meningkatkan produk.

LITERATURE REVIEW

Machine Learning

Pembelajaran mesin adalah bidang ilmu komputer yang melibatkan pembuatan algoritme yang secara berguna mengandalkan kumpulan contoh fenomena tertentu. Contoh-contoh ini bisa alami, buatan manusia, atau dihasilkan oleh algoritma lain. Pembelajaran

mesin juga dapat didefinisikan sebagai proses 1) mengumpulkan kumpulan data dan 2) membangun model statistik berdasarkan kumpulan data ini untuk memecahkan masalah praktis melalui algoritme. Asumsikan bahwa model statistik digunakan dalam beberapa cara untuk memecahkan masalah nyata. Untuk menghemat penekanan tombol, saya menggunakan istilah "belajar" dan "pembelajaran mesin" secara bergantian (Burkov, 2019).

Supervised Learning

Supervised learning merupakan pendekatan dimana sudah ada data latih, dan ada variabel sasaran sehingga tujuan dari pendekatan ini adalah untuk mengelompokkan data ke dalam data yang sudah ada (Andreas Chandra, 2017).

Random Forest

Random Forest (RF) adalah metode ensemble berbasis pohon yang dirancang untuk mengatasi kekurangan metode klasifikasi dan pohon regresi (CART). RF terdiri dari sejumlah besar kelemahan pohon keputusan klasifikasi dan regresi, yang tumbuh secara paralel untuk mengurangi bias dan varians model pada saat yang bersamaan (Breiman, 2001).

Berikut adalah rumus untuk random forest:

$$Entropy\ (Y) = -\sum_{i} p(c|Y) log^{2}! \text{ " } p(c|Y), \quad (1)$$

Information :
Y = Case Set
P(c|Y) = The proportion of the value of Y to class c.

Information Gain (Y,a)

$$= Entropy\ (Y) - \sum_{v \varepsilon Values(a)} \frac{|Y_v|}{|Y_a|} Entropy\ (Y_v). \quad (2)$$

Information :
Values(a) = Possible values of the case set a.
Y_v = Subclass of Y with class v corresponding to class a.
Y_a = All values corresponding to a.

Google Play Store

Google Play adalah layanan distribusi digital yang dioperasikan dan dikembangkan oleh Google. Ini berfungsi sebagai toko aplikasi resmi untuk sistem operasi Android, memungkinkan pengguna untuk menelusuri dan mengunduh aplikasi yang dikembangkan dengan kit pengembangan perangkat lunak Android (SDK) dan diterbitkan melalui Google. Google Play juga berfungsi sebagai toko media digital, menawarkan musik, buku, film, dan program televisi. Ini sebelumnya menawarkan perangkat keras Google untuk dibeli hingga pengenalan pengecer perangkat keras online terpisah, Google Store, pada 11 Maret 2015, dan juga menawarkan publikasi berita dan majalah sebelum perbaikan Google Berita pada 15 Mei 2018 (google, 2012a)

Aplikasi tersedia melalui Google Play baik secara gratis atau berbayar. Mereka dapat diunduh langsung di perangkat Android melalui aplikasi seluler Play Store atau dengan menerapkan aplikasi ke perangkat dari situs web Google Play. Aplikasi yang mengeksploitasi kemampuan perangkat keras suatu perangkat dapat ditargetkan pada pengguna perangkat dengan komponen perangkat keras tertentu, seperti sensor gerak (untuk game yang bergantung pada gerakan) atau kamera depan (untuk panggilan video online). Google Play store memiliki lebih dari 82 miliar unduhan aplikasi pada tahun 2016 dan telah mencapai

lebih dari 3,5 juta aplikasi yang diterbitkan pada tahun 2017. Ini telah menjadi subyek berbagai masalah terkait keamanan, di mana perangkat lunak berbahaya telah disetujui dan diunggah ke toko dan diunduh oleh pengguna , dengan berbagai tingkat keparahan (google, 2012b).

Google Play diluncurkan pada 6 Maret 2012, menyatukan Android Market, Google Music, dan Google eBookstore di bawah satu merek, menandai perubahan dalam strategi distribusi digital Google. Layanan yang termasuk dalam Google Play adalah Google Play Buku, Google Play Game, Google Play Film & TV, dan Google Play Musik. Setelah re-branding, Google secara bertahap memperluas dukungan geografis untuk setiap layanan (google, 2012b)

MATERIALS & METHODS

Methodology

Penelitian ini dilakukan secara bertahap yang akan dilakukan mulai dari penentuan dataset. Tahap selanjutnya adalah proses preprocessing data. Tahap selanjutnya adalah proses prediksi rating aplikasi menggunakan Random Forest. Setelah semua tahapan proses siap, tahap selanjutnya akan dilakukan analisis terhadap hasil yang diperoleh dengan membandingkannya dengan nilai RMSE (Root Mean Squared Error) untuk mengetahui akurasi hasil imputasi dan hasil akurasi prediksi dihitung dengan melihat pada persentase akurasi.

Data Prapemrosesan

Preprocessing data yang digunakan adalah dengan mengubah nilai atribut ke dalam bentuk numerik untuk meminimalkan kesalahan. Alat-alat yang digunakan dalam preprocessing menggunakan aplikasi jupyter python.

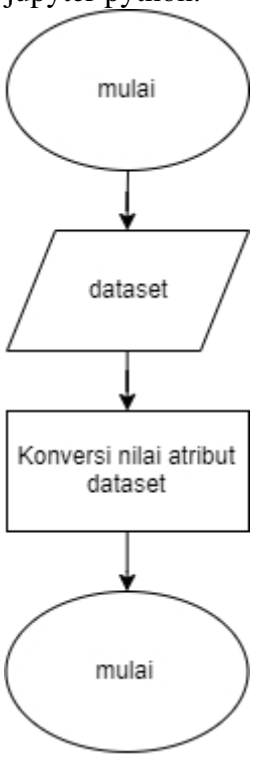

Figure 1. Preprocessing Data

Preprocessing mengonversi nilai atribut dengan integer atau float.

a. Konversi nilai atribut Aplikasi
b. Kategori . konversi nilai atribut
c. Hapus Simbol pada Instal nilai atribut atribut
d. Konversi jenis. nilai atribut
e. Harga . konversi nilai atribut
f. Konversi nilai atribut Pembaruan Terakhir
g. Konversi nilai atribut Android Ver
h. Konversi nilai atribut Ver Saat Ini
i. Convert Ukuran. nilai atribut
j. Konversi nilai atribut Rating Konten
k. Konversi nilai atribut atribut Genre

Prediksi Peringkat Aplikasi

Metode random forest merupakan pengembangan dari metode CART (Classification and Regression Tree) dengan menerapkan metode bootstrap aggregating (bagging) dan random feature selection oleh Breiman (2001). Random forest merupakan salah satu metode yang digunakan untuk klasifikasi dan regresi. Metode ini merupakan metode pembelajaran ensemble dengan menggunakan pohon keputusan sebagai base classifier yang dibangun dan digabungkan (Kulkarni dan Sinha, 2014).

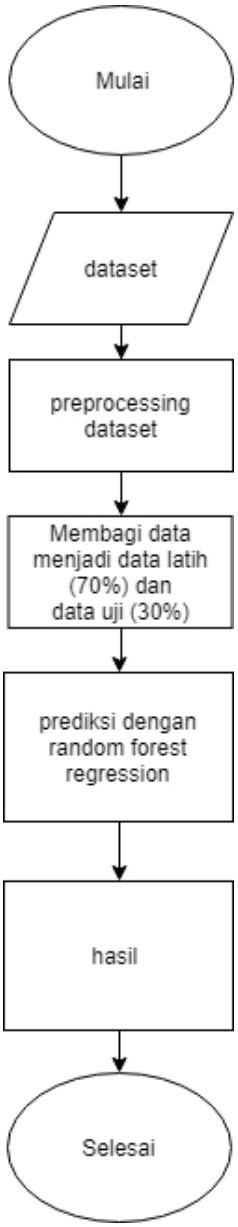

Figure 2. App Rating Prediction

Ada tiga aspek penting dalam menggunakan metode hutan acak.

a. perform bootstrap sampling untuk membangun pohon prediksi.
b. setiap pohon keputusan memprediksi dengan prediktor acak.
c. lalu random forest membuat prediksi dengan menggabungkan hasil dari setiap pohon keputusan dengan cara suara terbanyak untuk klasifikasi atau rata-rata untuk regresi.

RESULT AND DISCUSSION

Penelitian ini akan menggunakan dataset dari Google Play. Pengujian ini akan menggunakan dataset yang telah dibagi berdasarkan hasil preprocessing data seperti yang dijelaskan pada pengujian yang akan dilakukan menggunakan dataset yang telah preprocessed menggunakan integer atau float unit, proses prediksi penggunaan python dengan metode hutan acak.

Hasil Berbagi Data

Pada tahap ini, distribusi data telah dilakukan preprocessing seperti yang telah dijelaskan pada bab sebelumnya dengan jumlah data sebanyak 10840 dan dengan 13 atribut. Berikut adalah informasi mengenai dataset yang digunakan, yang dapat dilihat pada tabel berikut:

Table 1. Information dataset

Atribute	Value	Status	Type
App	10840	non-null	int64
Category	10840	non-null	int64
Rating	9424	non-null	float64
Reviews	10840	non-null	int64
Size	10840	non-null	float64
Installs	10840	non-null	int64
Type	10840	non-null	int64
Price	10840	non-null	float64
Content Rating	10840	non-null	int64
Genres	10840	non-null	int64
Last Updated	10840	non-null	int64
Current Ver	10840	non-null	float64
Android Ver	10840	non-null	float64

Dari tabel 1 terlihat bahwa dari 10840 terdapat nilai missing pada atribut Rating dengan nilai 1416 data.

Berikut ini adalah beberapa contoh missing value yang ditemukan pada dataset yang digunakan dapat dilihat pada tabel simulasi berikut:

Table 2. dataset before imputing missing value

	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	Genres	Last Updated	Current Ver	Android Ver
10810	4305	1	NaN	4	3.9	100	1	0	1	13	1.53E+09	1.36	4.4
10811	4604	11	4.1	80	13	1000	1	0	1	39	1.53E+09	2.02	4.03
10812	2905	4	NaN	20	2.7	10000	1	0	1	22	1.53E+09	2.11	4.1
10813	4309	11	4	785	31	50000	1	0	4	52	1.43E+09	1.31	3
10814	4892	3	4.2	5775	4.9	500000	1	0	1	19	1.53E+09	7.046	4.2
10815	4423	4	NaN	2	6.8	100	1	0	1	22	1.53E+09	2.18	4.1
10816	5086	29	4	885	8	100000	1	0	1	108	1.45E+09	1.061293	5
10817	4888	12	NaN	96	1.5	10000	1	0	1	60	1.46E+09	2.3	2.2
10818	4368	3	3.3	52	3.6	5000	1	0	4	19	1.50E+09	0.34	4.1
10819	4608	11	5	22	8.6	1000	1	0	4	39	1.53E+09	3.8	4.1
10820	7097	11	NaN	6	2.5	50	1	0	1	52	1.53E+09	1	4.03
10821	6842	25	NaN	0	3.1	10	1	0	1	82	1.51E+09	1	4.4
10822	5828	31	NaN	1	2.9	100	1	0	1	114	1.52E+09	1	4.03
10823	2405	20	NaN	67	82	10000	1	0	1	71	1.53E+09	2.22	4.4
10824	6528	27	NaN	7	7.7	100	1	0	4	101	1.52E+09	1	4

Dari tabel 2 terlihat pada tanda merah terdapat nilai kosong atau nilai NaN pada atribut Rating. Sedangkan untuk data sharing akan dibagi dengan menghapus data yang kosong.

Pada tahap ini, kami akan menghapus data kosong di dataset dengan menjatuhkan data menggunakan python. Hasilnya bisa dilihat pada gambar berikut:

Table 3. Dataset information after deletion of missing data

Atribute	Value	Status	Type
App	10840	non-null	int64
Category	10840	non-null	int64
Rating	10840	non-null	float64
Reviews	10840	non-null	int64

Size	10840	non-null	float64
Installs	10840	non-null	int64
Type	10840	non-null	int64
Price	10840	non-null	float64
Content Rating	10840	non-null	int64
Genres	10840	non-null	int64
Last Updated	10840	non-null	int64
Current Ver	10840	non-null	float64
Android Ver	10840	non-null	float64

Tabel 3 menunjukkan nilai untuk semua atribut adalah sama, artinya nilai kosong pada atribut Rating akan menghapus semua baris data.

Hasil divisi pengujian

Pada penelitian ini akan ditampilkan hasil penelitian missing value imputasi untuk digunakan dalam prediksi rating di Google Play Store menggunakan algoritma random forest. Penelitian ini akan membagi data sebanyak 10840 dibagi menjadi 2, dengan perbandingan 70:30, jumlah data latih terdiri dari 7588 data dan jumlah data pengujian terdiri dari 3252 data. pengujian kinerja berdasarkan MAE, RMSE, dan MSE. Kemudian dilakukan evaluasi kinerja hutan acak dengan menggunakan parameter pengukuran yaitu akurasi.

Hasil Tes Algoritma Random Forest

Pengujian pertama akan dilakukan dengan menggunakan eksperimen dengan data yang nilai-nilai yang hilang telah dihilangkan. Percobaan dilakukan dengan menggunakan algoritma random forest dengan parameter jumlah pohon 200 dan kedalaman pohon 10, 20, dan 30. Pengujian akan menggunakan pengukuran akurasi dan kinerja sebagai pembanding hasil. Pengujian dilakukan dengan jumlah pohon sebanyak 200 pohon dengan kedalaman pohon 10, 20, dan 30. Pengujian akan menggunakan akurasi dan performansi. Berikut hasil pengujian dengan jumlah pohon 200 dan menjadi pohon 10, 20, dan 30.

Table 4. Testing with tree 200 and deep tree 10, 20, 30.

nilai K Imputasi	N_estimator	Deep Tree	MAE	RMSE	MSE	Akurasi
Tanpa Imputasi	**200**	**10**	0.242	0.4	0.16	0.938
		20	0.241	0.399	0.159	0.938
		30	0.242	0.401	0.161	0.938

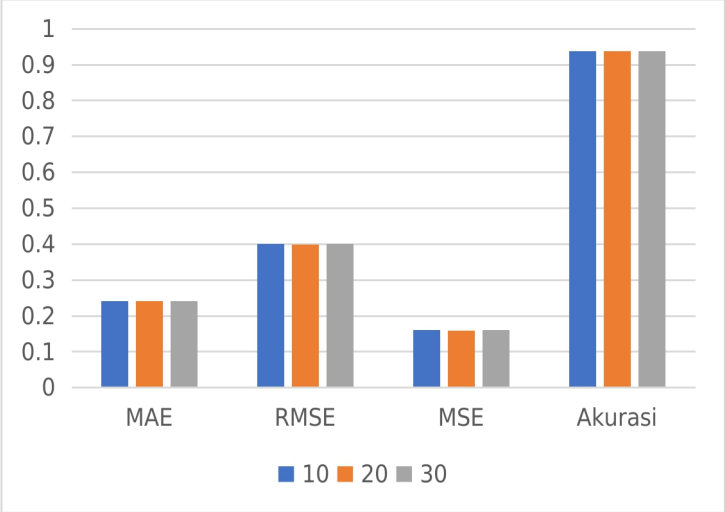

Figure 3. Grafik performance tanpa imputasi deep tree dengan tree 200

Dari Tabel 4. dapat dilihat bahwa secara umum nilai akurasi dapat berbeda-beda untuk setiap pengujian. Pengujian dilakukan sebanyak 3 kali dengan menggunakan nilai deep tree 10, 20, 30 dengan total 200 pohon. Dari pengujian tersebut didapatkan hasil akurasi tertinggi yaitu 93.8% MAE 0.399, RMSE 0.9551, dan MSE 0.159.

KESIMPULAN

Rating biasanya diberikan oleh pengguna dan dijadikan sebagai tolak ukur untuk mengetahui apakah aplikasi yang dibuat sudah bagus atau masih ada kelemahannya. Jika terdapat kelemahan, maka dengan menggunakan model prediksi yang digunakan, pengembang dapat mengetahui faktor-faktor apa saja yang menjadi kelemahan dari aplikasi tersebut. Berdasarkan permasalahan yang terjadi, algoritma Random Forest memiliki performa terbaik dari algoritma lainnya dalam membantu menemukan kelemahan pada dataset google play store. Dengan akurasi 93.8%, MAE 0.339, RMSE 0.9551 dan MSE 0.159.

REFERENCES

1. Andreas Chandra (2017) *PERBEDAAN SUPERVISED AND UNSUPERVISED LEARNING*. Available at: https://datascience.or.id/article/Perbedaan-Supervised-and-Unsupervised-Learning-5a8fa6e6.
2. Appbrain (2021) *Number of Android Apps on Google Play*. Available at: https://www.appbrain.com/stats/number-of-android-apps.
3. Breiman, L. (2001) 'Random Forests', *Machine Learning*, 45(1), pp. 5–32. doi: 10.1023/A:1010933404324.
4. Burkov, A. (2019) 'The Hundred-Page Machine Learning Book-Andriy Burkov', *Expert Systems*, 5(2), pp. 132–150. doi: 10.1111/j.1468-0394.1988.tb00341.x.
5. Chand, N. *et al.* (2016) 'A comparative analysis of SVM and its stacking with other classification algorithm for intrusion detection', *Proceedings - 2016 International Conference on Advances in Computing, Communication and Automation, ICACCA 2016*. doi: 10.1109/ICACCA.2016.7578859.
6. google (2012) *Introducing Google Play: All your entertainment, anywhere you go*, *googleblog*.
7. Hartmann-Boyce, J. *et al.* (2017) 'Insights From Google Play Store User Reviews for the Development of Weight Loss Apps: Mixed-Method Analysis', *JMIR mHealth and*

uHealth, 5(12), p. e203. doi: 10.2196/mhealth.8791.
8. Hengshu Zhu *et al.* (2014) 'Popularity Modeling for Mobile Apps: A Sequential Approach', *IEEE Transactions on Cybernetics*, 45(7), pp. 1303–1314. doi: 10.1109/tcyb.2014.2349954.
9. Kulkarni, V. Y. and Sinha, P. K. (2014) 'Effective Learning and Classification using Random Forest Algorithm', *International Journal of Engineering and Innovative Technolgy*, 3(11), pp. 267–273.
10. Shen, S., Lu, X. and Hu, Z. (2017) 'Towards Release Strategy Optimization for Apps in Google Play'. Available at: http://arxiv.org/abs/1707.06022.
11. Zhu, H. *et al.* (2014) 'Mobile App Recommendations with Security and Privacy Awareness Categories and Subject Descriptors', *Proc. of the 20th ACM SIGKDD international conference on Knowledge Discovery and Data mining (KDD)*, pp. 951–960. doi: 10.1145/2623330.2623705.

******