Scrapp Recycling

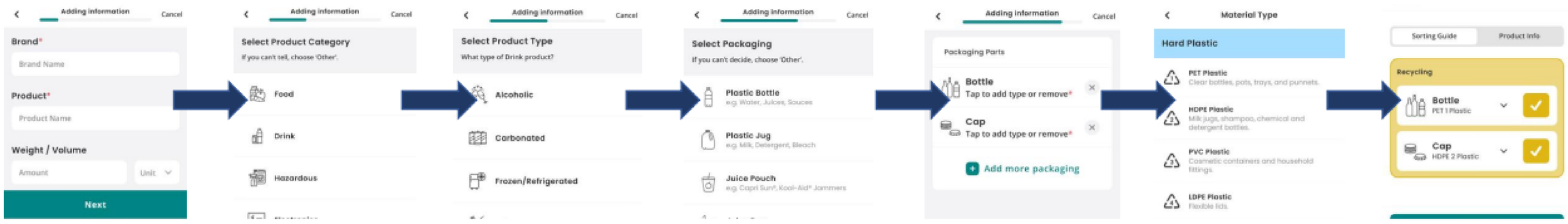
Predicting Packaging Parts and Materials

Tutku Cinkilic

School of Mathematics, University of Edinburgh, Edinburgh, U.K. s2259604@ed.ac.uk

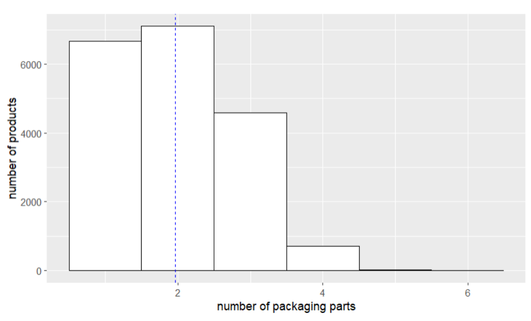
1. Introduction

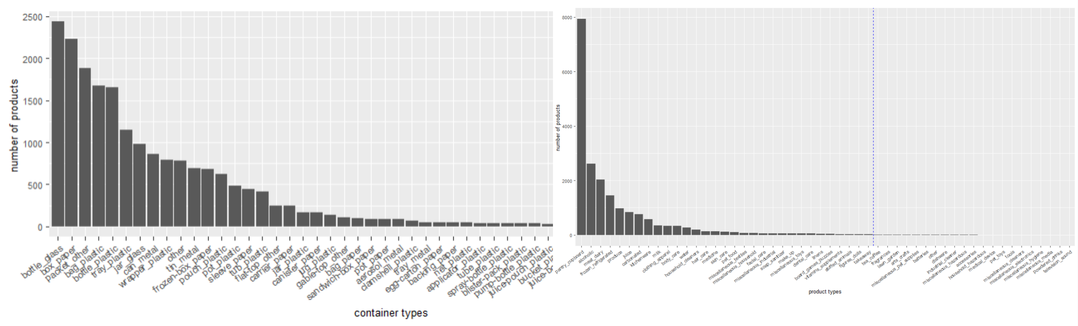
Scrapp app is a mobile app offering its users recycling guideline when they scan the product they want to recycle. In order to do that the most reliable packaging information of products comes directly from companies. Another resource is the app's users which is easier to collect but less convenient. Because the user input needs to be verified by the company specialist before being added to their database. This is a manual and time-consuming process that the company would like to automate. In order to check the accuracy of the submitted data, they would like to utilize their on-hand verified data.



2. Data

- **productCategory:** (15 levels) Category of the product (food, drink etc.)
- **productType** (52 levels) Type of the product (alcoholic, carbonated etc.)
- **container** (50 levels) Container type of the product (glass bottle, plastic bottle etc.)
- **sizeUnit** (13 levels) Unit to define product's size (L, ml, kg etc.)
- **packagingPart1-7**: (65 levels) different packaging parts of the product (lid, bottle etc.)
- **material1-7:** (35 levels) materials of the packaing parts (paper, glass, metal etc.)





Data preparation

For packaging part models, we have altered dataset so that instead of 7 packagingPart columns, 65 binary columns are added, where each stands for a different packaging part level.

For material models, we applied copy transformation [1] which duplicates the entry as many times as the number of labels it has.

3. Models

Packaging Part Models: [2][3][4][5]

- Binary Relevance (BR)
  - BR with decision tree classifier (BR-DT)
  - BR with random forest classifier (BR-RF)
  - BR with Naive-Bayes classifier (BR-NB)
- Classifier Chains (CC):
  - CC with decision tree classifier (CC-DT)
  - CC with random forest classifier (CC-RF)
  - CC with Naive-Bayes classifier (CC-NB)

Material Models: [6]

- Multinomial Logistic Regression (MN-LR)
- Decision Tree (DT)
- Naive-Bayes (NB)

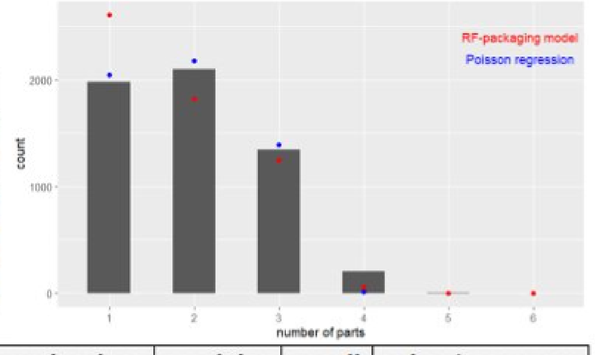
Number of Part Model:

- Poisson Regression [7]

4. Results

- **accuracy:** $\frac{TP+TN}{TP+TN+FP+FN}$
- **precision:** $\frac{TP}{TP+FP}$
- **recall:** $\frac{TP}{TP+FN}$
- **F1-score:** $2 * \frac{Precision*Recall}{Precision+Recall}$
- **Hamming-loss:** $\frac{FN+FP}{TP+TN+FP+FN}$
- **subset-accuracy:** exact match ratio

Algorithm	Set	accuracy	precision	recall	F1
MN-LR	Train	0.7641	0.7928	0.5718	0.6653
	Test	0.7584	0.6366	0.4893	0.5928
DT	Train	0.7891	0.8344	0.5564	0.6706
	Test	0.7782	0.7202	0.4707	0.6324
NB	Train	0.6615	0.6163	0.6209	0.5743
	Test	0.66	0.5575	0.5157	0.5414



Method	Algorithm	Set	accuracy	F1	hamming-loss	precision	recall	subset-accuracy
BR	DT	Train	0.801	0.8587	0.0089	0.9223	0.8341	0.6042
		Test	0.7922	0.8523	0.0095	0.9181	0.827	0.5896
	RF	Train	0.8187	0.8722	0.0081	0.9285	0.851	0.6368
		Test	0.8068	0.8634	0.0089	0.9204	0.8426	0.6149
	NB	Train	0.6876	0.7725	0.0178	0.7826	0.8305	0.3979
		Test	0.6896	0.774	0.0175	0.7866	0.8281	0.4001
CC	DT	Train	0.8023	0.8585	0.0091	0.9093	0.8431	0.6177
		Test	0.7937	0.8525	0.0097	0.9039	0.8365	0.6023
	RF	Train	0.8013	0.8576	0.009	0.9186	0.8341	0.6117
		Test	0.7951	0.8533	0.0094	0.9144	0.8296	0.6004
	NB	Train	0.5945	0.6985	0.0256	0.6375	0.8394	0.2678
		Test	0.6004	0.7028	0.0251	0.645	0.8373	0.2794

5. References

[1] Rachana Buch and Dhatri Ganda. A Survey on Multi Label Classification. pages 19–23, 2018.

[2] Min Ling Zhang, Yu Kun Li, Xu Ying Liu, and Xin Geng. Binary relevance for multi-label learning: an overview, 4 2018.

[3] Assoc Mona Nasr, Essam Shaaban, Rutvija Pandya, Jayati Pandya, and KPDholakiya Infotech Amreli. C5.0 Algorithm to Improved Decision Tree with Feature Selection and Reduced Error Pruning. Technical Report 16, 2015.

[4] Leo Breiman. Random Forests. Technical report, 2001.

[5] Mohammed J. Islam, Q. M. Jonathan Wu, Majid Ahmadi, and Maher A. Sid-Ahmed. Investigating the performance of naïve- bayes classifiers and K-nearest neighbor classifiers, 2010.

[6] P J Smith. Analysis of Failure and Survival Data. Technical report.

[7] Tammy Harris, Zhao Yang, and James W Hardin. Modeling underdispersed count data with generalized Poisson regression. Technical Report 4, 2012.