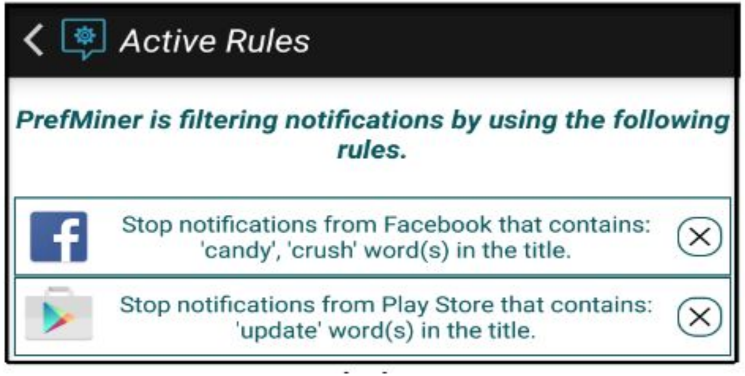
PrefMiner: Mining User's Preferences for Intelligent Mobile Notification Management

Motivation:

- Interruptibility management system should
  - send notifications only in OM
  - stop non-useful, uninteresting or irrelevant notifications

Goal:

- An app that learns the *types of information users prefer to receive via notifications* in different situations



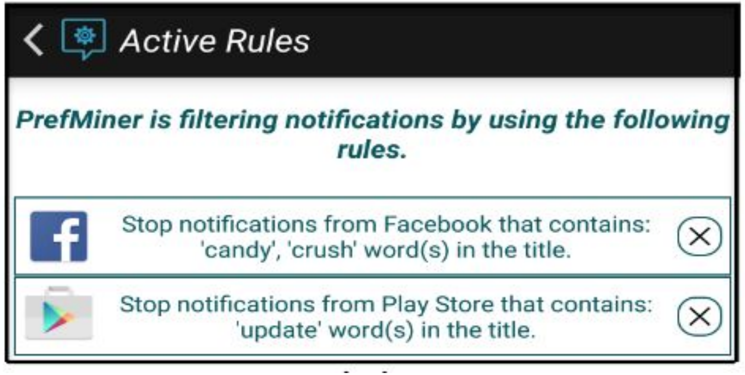
[1] Mehrotra, Abhinav, et al. "My phone and me: understanding people's receptivity to mobile notifications." *Proceedings of the 2016 CHI conference on human factors in computing systems*. 2016.

PrefMiner: Mining User's Preferences for Intelligent Mobile Notification Management

Motivation:
- Interruptibility management system should
  - send notifications only in OM
  - stop non-useful, uninteresting or irrelevant notifications

Goal:
- An app that learns the *types of information users prefer to receive via notifications* in different situations

Summary:
- Offline evaluation on MyPhone & Me dataset
  - 18 users // 11,185 notifications // with at least 14 days of data for each user
- In-the-wild 15-day experiment
  - 16 subjects are suggested 179 rules
  - 56.98% of suggested rules are accepted
  - 45% of unwanted notifications are filtered out



[1] Mehrotra, Abhinav, et al. "My phone and me: understanding people's receptivity to mobile notifications." *Proceedings of the 2016 CHI conference on human factors in computing systems*. 2016.

PrefMiner: Design Choices

1. Learning users' preferences instead of modeling their interruptibility by
    a. learning the different types of *unwanted* interruptions that users explicitly refuse by dismissing notifications
    b. Then, dismiss *unwanted* types of interruption
2. Association rules over ML models because:
    a. Users should be involved while removing *unwanted* interruptions without compromising the reception of any important information
    b. Interpretability of rules
        i. ML models from previous studies are not interpretable [21,28,24]
        ii. Mined association rules can be easily understood by users
        iii. Users can provide their feedback about the mined association rules

Mining User Preferences:

1. **Notification title classification**
   a. Standard NLP preprocessing of notification title
   b. Constructing a clusters for each app's notifications
      - Compute Document-Term Matrix (DTM)
      - Clustering each application's notifications separately with DTM and DBSCAN (e.g., N_1, N_2)

- D1 = "I like databases"
- D2 = "I dislike databases",

then the document-term matrix would be:

	I	like	dislike	databases
D1	1	1	0	1
D2	1	0	1	1

2. **Constructing Association Rules using**
   a. Context
      - Type of notification
      - Location
      - Time of the day
      - User's physical activity (e.g., Walking, still)
   b. Consequent
      - Whether Accept or Dismiss notification

Mining User Preferences:

1. **Notification title classification**
   a. Standard NLP preprocessing of notification title
   b. Constructing a clusters for each app's notifications
      i. Compute Document-Term Matrix (DTM)
      ii. Clustering each application's notifications separately with DTM and DBSCAN (e.g., N_1, N_2)

2. **Constructing Association Rules**
   a. ARM is constructed using
      i. Context
         - Type of notification
         - Location
         - Time of the day
         - User's physical activity (e.g., Walking, still)
      ii. Consequent
         - Whether Accept or Dismiss notification

Example:

Observed User Behavior:

1. always dismiss Twitter suggestion (N1) notifications for who to follow;
2. accepts Facebook birthday reminder notifications (N2) only in the morning while she is at home;
3. does not accept WhatsApp notifications from Alice (N3) while at work.

Association Rules from User Behavior:

$$\{N_1\} \to \{Dismiss\}$$
$$\{N_2, Home, Morning\} \to \{Accept\}$$
$$\{N_2, Home, Afternoon\} \to \{Dismiss\}$$
$$\{N_2, Home, Evening\} \to \{Dismiss\}$$
$$\{N_2, Home, Night\} \to \{Dismiss\}$$
$$\{N_2, Work\} \to \{Dismiss\}$$
$$\{N_2, Other\} \to \{Dismiss\}$$
$$\{N_3, Home\} \to \{Accept\}$$
$$\{N_3, Other\} \to \{Accept\}$$
$$\{N_3, Work\} \to \{Dismiss\}$$

Evaluation of Rule-based Mechanism: Setting

Dataset: My Phone and Me [25]

- 18 users
- 11,185 notifications

Eval procedure

- For each user 10-fold CV ~ 9:1 train:test split
  - So for each fold
    - ARs are mined from 90% of user data
    - Tested on 10% of user data
- Aggregated users' scores by
  - Mean
  - (margin of) Standard error with 95% CI

Eval Metrics:

- *Recall*
- *Precision*

Outcome variable

- *notification response*: the user's response (i.e., click or dismiss) to a notification;

Context Features:

- *notification type*: the identifier of the cluster to which the notification belongs;

- *arrival time*: the arrival time of the notification considering four time slots – morning (6-12), afternoon (12-16), evening (16-20) and night (20-24 and 0-6);

- *activity*: the user's physical activity (includes still, walking, running, biking and in vehicle) when the notification arrived;

- *location*: the user's location when the notification arrived;

Evaluation of Rule-based Mechanism: Results

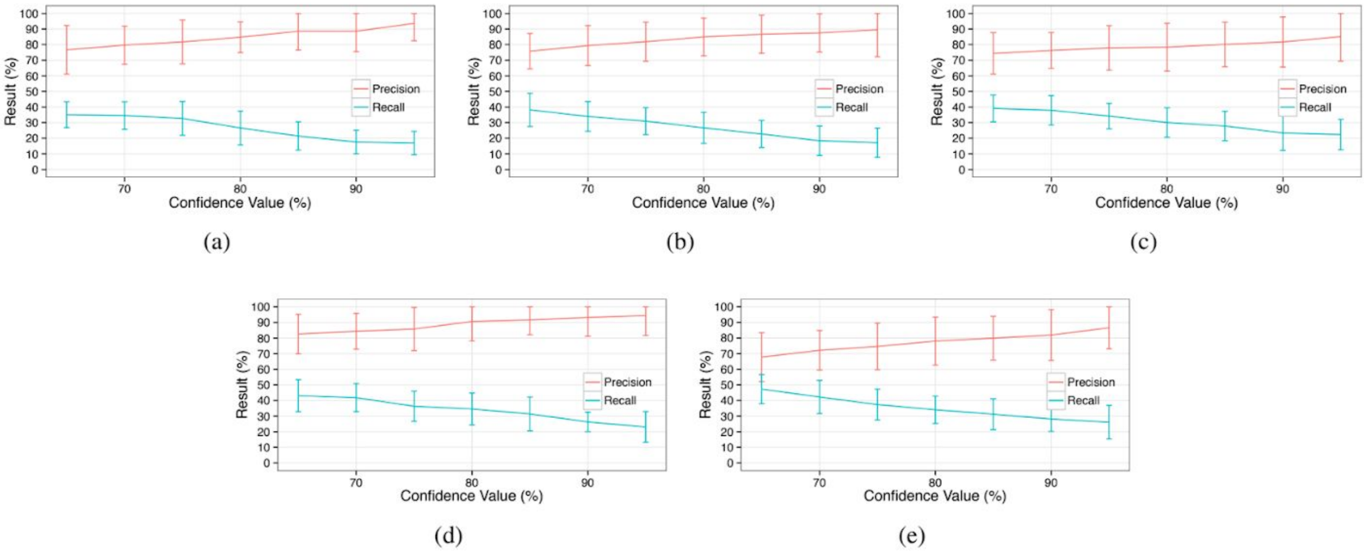


Figure 2. Prediction results for association rules discovered by using notification response along with: (a) AR1: notification type; (b) AR2: notification type and activity; (c) AR3: notification type and arrival time; (d) AR4: notification type and location; (e) AR5: notification type, activity, arrival time and location.

Key Insight:

- user's preference for receiving notifications does not depend on the activity and arrival time
- User preference for notification depends on the type of information it contains and the location of the user
    - at a confidence of 80%, 35 recall and 90% precision is achieved

Evaluation Results: Online Learning

Setting:

- It is not mentioned how many users and notifications
- New rules are constructed at the end of each day using notifications from Day 1 till Day N-1, evaluated on Notifications of Day N
- Confidence of the rules are color coded

Observation:

- Rules with C_{min} as 80 and below start filtering notifications from the 3rd day and by the 7th day they achieve the recall of 25-35% and precision around 90%.
- other rules could achieve the recall of 10% and precision above 90% by 7-th day.

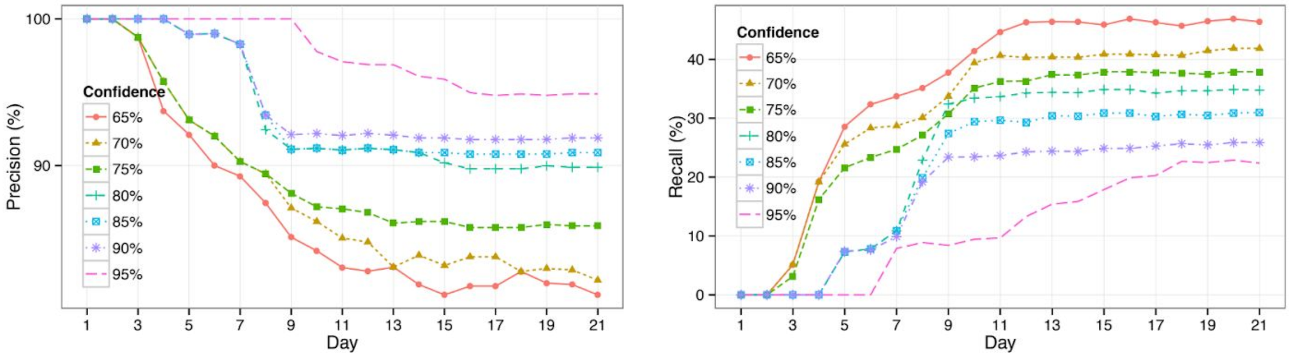


Figure 3. Prediction results for association rules (with notification title and location) using the online learning approach.

Evaluation results: In the wild Evaluation

- 16 users // 15 days
- During the study PrefMiner suggested 179 rules to the participants out of which 102 rules (i.e., ~57%) were accepted (Fig. 7).

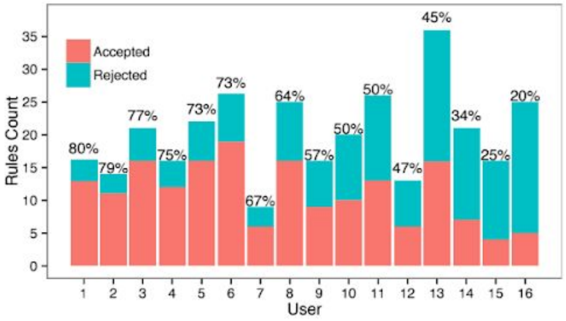


Figure 7. Users response to the rules suggested by the PrefMiner application.

Evaluation results: In the wild Evaluation

- 16 users // 15 days
- During the study PrefMiner suggested 179 rules to the participants out of which 102 rules (i.e., ~57%) were accepted (Fig. 7).
- Pref Miner identifies around 60% of unwanted (dismissed by the user) notifications (Fig. 8).

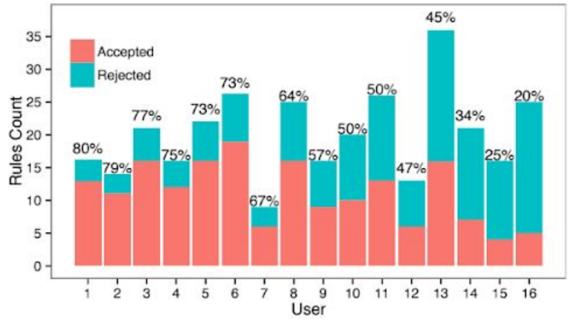


Figure 7. Users response to the rules suggested by the PrefMiner application.

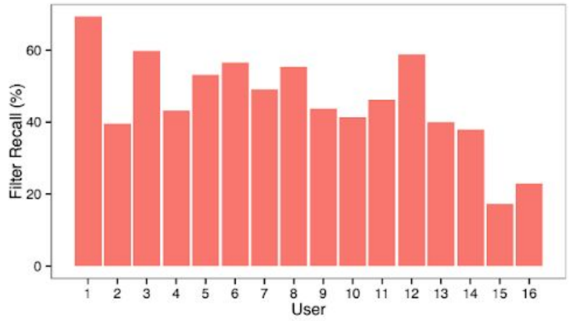


Figure 8. Performance of PrefMiner in filtering notifications.