Toward a Fair Review-Management System

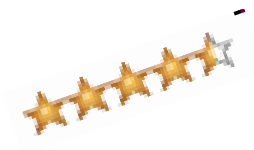
Theodoros Lappas & Evimaria Terzi

Item Reviews

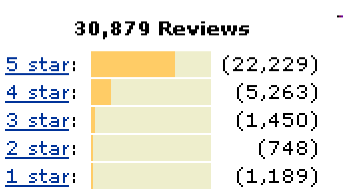
- Abundant in Review-hosting sites (E.g. Amazon, Yelp)
- Opinions on items and their attributes
- Valuable source of information
- Great Impact on purchase decisions

Review Management

- Challenges for review hosting sites
  - Quality (structural, informational)
  - Volume & Redundancy
  - Presentation & Ranking



- Handled by the Review Management System



E-Commerce & Reviews

- Key players: customers, businesses
- Businesses are satisfied through sales
- Customers are satisfied by
  - quality products
  - Well-presented, high-quality information on the products→ Reviews!!
- A third player to satisfy: the Reviewers.

Keeping Reviewers Motivated

- What can the RMS offer?
- Motivation for reviewers to submit content:
  - Genuine desire to help others
  - Frustration or excitement due to the reviewed item
  - The desire to influence others and gain acknowledgment via positive ratings (e.g. helpfulness votes)
  - The need to express one's self.
- Common factor: Visibility

Current Status

- Reviews ranked by date
  - No consideration of review quality
  - No Visibility guaranteed
- Reviews ranked by user ratings
  - Favors older reviews, reviews with many ratings
  - No Visibility guaranteed

Our Idea

- Formalize a compact ***spotlight set*** of high quality reviews that capture all item attributes.
- Periodically shuffle the reviews in the spotlight set to distribute visibility
- Inclusion in the spotlight set should be proportional to the reviews quality & contribution

Goals of our fair RMS

- **Attribute coverage**
- **Review Quality**
- **Fair spotlight share**
- **Compactness**

Attribute Coverage

- Each review can be represented by the vector of attributes it discusses
  - (*screen, battery-life, price*)
- Each attribute should be discussed in ***at least one*** of the reviews in the spotlight set.
- A generalization could ask for ***at least k*** reviews per attribute.
- Alternative formulation: ask for coverage of *opinionated* attributes.

Review Quality

- Various available measures
  - Structural Readability (e.g. Flesch Reading Ease [9])
  - Helpfulness [14, 20]
  - Spam Analysis [8]
- We use Threshold-based pruning
  - Simple, compatible with any measure or combination of measures

Fair spotlight share

- Let $\mathcal{C}$ be the complete universe of all possible spotlight sets (set covers)
- Some reviews participate in more covers than others
- On a high level: reviews that cover many attributes and reviews that cover rare attributes participate in more covers
- Formalize review contribution based on *the number of covers it participates in.*

Fair spotlight share

- Let ***p(r)*** be the number of covers that a review ***r*** participates in.
- If we can sample uniformly from $\bar{\mathcal{C}}$,the number of sampled spotlight sets with review ***r*** will eventually converge to ***p(r).***
- Problem: $\bar{\mathcal{C}}$ is not available, includes an exponential number of covers
- Can we still sample from $\bar{\mathcal{C}}$?

Importance Sampling

- Input: the collection $\mathcal{M}$ of all ***minimal*** covers
- Let $\mathcal{C}_i$ include all the supersets of a minimal cover M_i
- 3 conditions:
  - We can compute $|\mathcal{C}_i|$ in polynomial time.
    - Simple: $|\mathcal{C}_i| = 2^{n-|M_i|}$.
  - We can sample uniformly at random from $\mathcal{C}_i$.
    - append to M_i each review in $R \setminus M_i$ with probability 1/2
  - Given any subset of reviews $R' \subseteq R$, we can verify in polynomial time if $R' \in \mathcal{C}_i$.
    - simply check if R' is a superset of M_i.

The Algorithm

Algorithm 1 The `ImportanceSampling` algorithm.

Input: Set of minimal covers $\mathcal{M}$, number of desired samples N.
Output: Spotlight sequence $\mathcal{S}$ of length N .

0: Define a fixed order for the covers in $\mathcal{M}$.
1: $\mathcal{S} \leftarrow \emptyset$
2: **while** $|\mathcal{S}| < N$ **do**
3: pick M_i from $\mathcal{M}$ with probability $\frac{2^{n-|M_i|}}{\sum_{M \in \mathcal{M}} 2^{n-|M|}}$
4: Generate a superset $S \in \mathcal{C}_i$ of M_i by appending each review $r \in R \setminus M_i$ with probability $1/2$.
5: Go over the covers in $\mathcal{M}$ in order, let M_{i^*} be the first cover that is a subset of S
6: **if** $i^* = i$ **then** $\mathcal{S} = \mathcal{S} \cup \{S\}$
7: **return** $\mathcal{S}$

Algorithm Discussion

- Instead of sampling from the space of all possible review subsets, our algorithm samples from the subspace of subsets that are also solutions (covers).

- Computing $\mathcal{M}$ can be a non-trivial task. Our experiments show that a subset of is sufficient.

Compactness

- Standard Importance sampling can return covers of arbitrary size.
- The Limited attention span of users motivates compact covers.
- We propose a modified version of the algorithm that allows us to tune the size of the sampled covers.

Evaluation

Datasets

- Four review datasets provided by Lappas and Gunopulos [11].
- GPS and TVS datasets include the complete review corpora from Amazon.com for 20 GPS systems and 20 TV sets, respectively.
- The VEG and SFR datasets include the complete review corpora from Yelp.com for 20 Las Vegas Hotels and 20 San Francisco restaurants, respectively.
- We use the method by Hu and Liu [7] for attribute extraction.
- We do an additional pass to prune out trivial attributes and address synonymy issues (e.g. *bathroom=restroom=toilet*).

Qualitative evidence - Spotlight Sets

Item 1 (**TVS**), Attributes: { *picture, price, warranty, sound, design, menu*}:

*"...Of all the LCD Tvs the *** overall seemed to have a brighter picture, has 120Hz, 2 year warranty, reasonably priced and..."*
*"...The *** delivers outstanding picture quality and sound..."*
"...Intuitive menu, easy to plug and play with most any hook up and source... The design of the TV is stunning, beautiful work all around..."

Item 2 (**SFR**), Attributes: {*food, price, staff (service), restrooms (bathrooms)* }

"...The food is delicious, prices are fair, venue is nice, staff is friendly, restrooms are clean..."
"...BAD SERVICE, WORSE ATTITUDES, AND EXTREMELY HIGH PRICES ..."
"...The food was substandard, unfortunately..."
"...the only drawback were the bathrooms..."

Spotlight Shuffling

- 3 baselines:
  - **RandomSampling:** pick reviews uniformly at random, until all the attributes are covered.
  - **GreedySampling:** Greedily append the reviews that covers the most attributes until all are covered.
  - **HelpSampling:** append a review with probability of proportional to the number of its helpfulness votes, until all attributes are covered

Experimental Setup

- Focus on the item with the most reviews from each dataset
- Use each approach to sample 1000 spotlight sets for each item.
- Compactness: allow for a maximum of 10 reviews per spotlight set.
  - If an approach reaches the bound without covering all the attributes, the cover is marked as “incomplete”.

Results

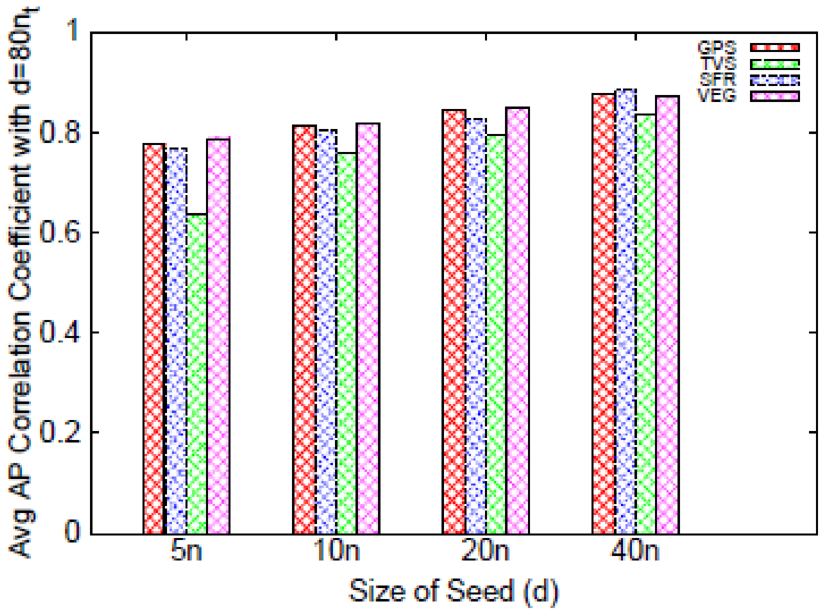
Table 1. Evaluation on the spotlight-set shuffling task

	0	[1-20)	[20-40)	[40-60)	[60-80)	$80 \leq$	Inc. %
TVS							
ImportanceSampling	0.0	0.67	0.17	0.06	0.05	0.05	8%
GreedySampling	0.78	0.14	0.03	0.03	0.0	0.02	0%
RandomSampling	0.0	0.0	0.32	0.68	0.0	0.0	90%
HelpSampling	0.41	0.19	0.18	0.09	0.02	0.1	48%

- Columns 2-6 contain the percentage of reviews that appeared in the respective number of sampled spotlight sets.
- Column 7 contains the percentage of Incomplete covers
- GreedySampling limits visibility to a very small portion of reviews
- RandomSampling fails to produce complete covers
- HelpSampling has a 50% “incomplete” percentage and also fails to distribute visibility
- ImportanceSampling does well on both accounts

Minimal Covers

- **ImportanceSampling** requires as seed the collection of all minimal covers
- We experiment with using a subset thereof



- Start with full **R**, randomly remove reviews until reaching a minimal cover
- Show size of Seed Vs. AP correlation Coefficient

Conclusion

- A Fair Review Management System
  - Presents thorough & compact sets of high-quality reviews to the customers
  - Keeps reviewers motivated by fairly distributing visibility
- Our framework is flexible and practical for virtually every review-hosting site

Thank You!