Contribution of

Personalizing Product Rankings using Collaborative Filtering on Opinion-derived Topic Profiles

(appeared at IJCAI 2015)

to the RECNLP workshop at AAAI 2019

Within the large volume of literature that uses review texts for recommendation, this work is very different since the purpose of using review texts is not to improve on the precision of rating-based recommendation.

Instead, we use review texts as the basis for a similarity metric, and thus make it possible to find users with similar tastes whose ratings can be used to recommend items. In this way, it becomes possible to apply the collaborative filtering idea to domains such as hotels, where it is extremely rare for even 2 people to have both stayed in several hotels, and thus no rating-based similarity exists.

There is no other comparable work in the literature and we think this approach deserves more attention in the community, as it opens up entirely new possibilities for recommenders. Furthermore, the results shown in the paper are very strong : we show that the unpersonalized ranking currently provided by the travel site has almost no correlation with actual users' ranking, while the personalized ranking using NLP-based collaborative filtering matches it perfectly for the cases where there is enough data.

We believe that there is a lot of potential for this direction that would be interesting for the workshop community.

Personalizing Product Rankings using Collaborative Filtering on Opinion-derived Topic Profiles

Claudiu-Cristian Musat, Boi Faltings
Ecole Polytechnique Federale de Lausanne
Lausanne, Switzerland
firstname.lastname@epfl.ch

Abstract

Product review sites such as TripAdvisor, Yelp or Amazon provide a single, non personalized ranking of products. The sparse review data makes personalizing recommendations difficult. Topic Profile Collaborative Filtering exploits review texts to identify user profiles as a basis for similarity. We show that careful use of the available data and separating users into classes can greatly improve the performance of such techniques. We significantly improve MAE, RMSE, and Kendall tau, compared to the previous best results. In addition, we show that personalization does not benefit all the users to the same extent. We propose switching between a personalized and a non personalized method based on the user opinion profile. We show that the user's opinionatedness is a good indicator of whether the personalization will work or not.

1 Introduction

Product review platforms have become a key source of information for most consumers. It is now commonplace to look for hotels, restaurants and products by their ranking on these sites, and being ranked highly is a big commercial advantage. However, not everyone can stay at the top-ranked hotels and eat at the top-ranked restaurants, especially since they are often small establishments. Furthermore, the ranking is obtained by mixing evaluations that are made by different people with often very different criteria: the hotel in a busy downtown street with small and noisy rooms may be perfect for the business traveller, but the family on vacation might prefer a hotel in the suburbs. Thus, it is not clear that the ranking actually reflects anyone's preferences, and it would be much better to personalize rankings to make them fit the criteria of a particular user.

The texts associated with reviews contain the necessary information for this task: they justify the rating by observations regarding the different aspects of the hotel, restaurant or product. Using natural language processing, this information can be automatically extracted to obtain a summary for each aspect of the review, and allow content-based recommendation, using the preferences expressed by a user. However, even as a multitude of approaches that use the review text are available [Chen *et al.*, 2015], they have rarely been used in practice due to the less reliable text-derived data.

An alternative to content-based recommendation is collaborative filtering. However, the numeric rating data is so sparse that this is not directly applicable. For example, in a TripAdvisor[1] hotel review dataset [Musat *et al.*, 2013], less than 10% of 65'000 users had a peer with whom they have co-rated two or more hotels and out of the 220 hotels, 25 did not have a single co-rating user. Given that the review space is sparse, collaborative filtering based on the ratings alone cannot be accurately used there. Review texts can be used to determine the similarity of users, without the requirement that they rate the same hotels. The intuition is that users only make the effort of writing about topics that they actually care about. In Figure 1 *Eric* has written twice about cleanliness and price. This signals an increased interest in these topics, compared to ones that he has not mentioned, such as the location. The overlap between the users' interests is a reliable way of measuring similarity.

However, when applied naively to the user population and reviews found on Tripadvisor, the performance of content-based techniques can disappoint. To obtain significant improvements, it is necessary to use additional measures, which are the focus of this paper.

The first is to personalize the prediction only when there is *sufficient data* to apply collaborative filtering. The main weakness of traditional collaborative filtering is that the space is sparse, and not enough users that have co-rated an item can be found to make a prediction. For opinion-based collaborative filtering, the risk is that the system will infer interest overlap based on too little data, for instance that someone mentioned the pool once in all their reviews. We thus propose a data selection step, that will apply opinion-based personalization only where the data is sufficient. We obtain prediction errors that are significantly smaller than those of the non personalized benchmark and previous Topic Profile Collaborative Filtering (TPCF) [Musat *et al.*, 2013] results. We confirm the assumption that, with increased data availability, TPCF will perform better than a non personalized benchmark in predicting a user's product ranking. The accuracy of the prediction, measured using Kendall's τ, is a good indicator of the actual performance of the recommender.

[1] http://www.tripadvisor.com/

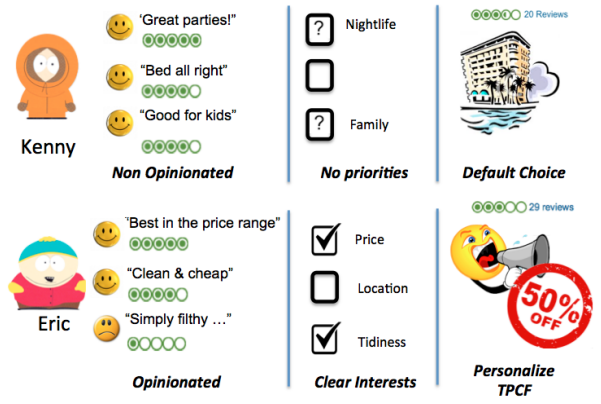

Figure 1: User Differentiated Recommendations

We then show that not all the users should be treated the same, and that opinion-based result ranking personalization should not be applied indiscriminately. In the case of hotel reviews, we separate regular users from *non opinionated* ones, that only express positive opinions about all hotels. This user set is likely to include most spammers [Mukherjee *et al.*, 2012]. We are the first to create a user-oriented recommender hybridization method where users are separated using the opinions in their reviews.

This paper brings the following contributions:

- We show, for the first time, how to choose the relevant review texts to obtain personalized rankings of good quality that are significantly more accurate than the non-personalized rankings.
- We formally define relevant review data.
- We obtain higher improvements in both MAE (mean average error) and RMSE (root-mean-square error) compared to previous results
- We create a user-oriented recommender hybridization method where users are separated using the opinions in their reviews.

2 Related Work

Personalization requires a form of user profiling. Most methods use the user's previous numeric ratings as a readily available, easy to use information source. The downside is that the data is sparse - we cannot properly define user preferences over sets of products [Kazienko and Kolodziejski, 2006]. The review text can help reduce the data sparseness, but it is difficult to process to at a level sufficient to compete with simpler, numeric alternatives. Review texts have been used to profile the users by detecting their implicit interests [McAuley and Leskovec, 2013]. We propose a similar approach, but the key difference is that we look for explicit opinions as signs of genuine interest. It is important to distinguish cases where a topic is mentioned accidentally (e.g. *When we walked into the room..*) from ones where the topic is actually the important element (*e.g. The room was dark..*).

Mining textual opinions offers a wide range of possibilities. As observed in [Snyder and Barzilay, 2007], the overall opinion within a given review is not helpful. Along with most researchers [Jakob *et al.*, 2009; Levi *et al.*, 2012; Liu *et al.*, 2013], we use aspect-based opinions. [Jakob *et al.*, 2009] used textual product features, along with demographic information to complement numeric ratings for collaborative filtering, but showed marginal improvements of the AUC, of roughly 1%. In [Levi *et al.*, 2012] hotel recommendations are generated using sentiment towards selected textual features, along with the user's nationality and the trip purpose. The user's preferences are manually extracted, and the evaluation is based on questionnaires, making the method hard to apply on a large scale.

Topic profile collaborative filtering [Musat *et al.*, 2013] is a collaborative filtering method that uses textual opinions to quantify the user's interests and compute the user similarity. Unlike in traditional collaborative filtering, two users are considered related if they share the same interests, not necessarily if they have rated the same products. This circumvents the greatest disadvantage of collaborative filtering - the sparseness of the space. The numeric ratings are then used to recommend items using collaborative filtering. PORE [Liu *et al.*, 2013] also use the review opinions to quantify the user's interests, which they use in collaborative filtering. They show that traditional CF methods perform significantly worse than random, because the data is too sparse and PORE performs slightly better than random, but it still does not yield a good result. TPCF thus solves a simpler problem using textual opinions - that of reducing the sparseness limitation - and get better results. This idea is also present in [Zhang *et al.*, 2013]. Their technique infers the reviewer's overall opinion, also named virtual rating using advanced opinion mining techniques. The virtual rating complements the real rating and results show that the best recommendations are obtained using the real rating and virtual rating together.

In this paper, we find where the above personalization brings the most benefit, given the available data and given the user characteristics. Hybridization is a natural response to the lack of quality data for a given method. *The Ensemble* [Sill *et al.*, 2009] won the second place in the Netflix Prize by blending predictions from multiple machine learning models. Hybridization with respect to the user has been, however, more rarely used [Ekstrand and Riedl, 2012] and the users are always modeled using their numeric ratings [van Setten *et al.*, 2003; Sill *et al.*, 2009; Dooms *et al.*, 2013]. Either viewed as an optimization problem [Dooms *et al.*, 2013], or as switching according to predefined rules [van Setten *et al.*, 2003], all the methods used by the hybrid systems are based on ratings. In this paper we propose a user-based hybridization method and apply different techniques for different user groups. Our goal is to show that separating the users, with respect to their opinions, improves the recommendation quality.

3 Opinion-based Recommendations

Mixing the text and numeric information sources is of great *practical importance*. While the text component of the reviews is more useful in defining the interests, the numeric part is more useful for determining the overall opinion about a given product. This is due to the imperfect nature of the

aspect-based opinion mining method used. In the user study presented in [Musat and Faltings, 2013], it has been shown that between 10% and 20% of the extracted aspect-based opinions had an incorrect polarity. The reasons included incorrect negation handling and irony. This uncertainty makes using individual opinions unreliable in product recommendations, at least for reviews written in English. Good results have been reported for inferring ratings for the review text for Chinese reviews [Zhang *et al.*, 2013].

We thus use TPCF as a basis for our analysis, as it only uses the textual opinions to model the writer's interests in certain topics, regardless of the subjectivity extraction performance. By using the opinion count, instead of each individual opinion polarity, it reduces the task complexity while using all available data.

3.1 Alternatives in Topic-based User Profiling

In collaborative filtering the user profiles are implicit - sets of social connections. Opinion-based interest profiles, as in other content based techniques, are explicit. They use the opinions expressed in the review text to get a rich description of the user's expectations and needs and measure similarity between users. The user's interests are aggregated into opinionated topics $z \in Z$. This increases the likelihood that there will be an overlap between the interests of various users and reduces the sparseness of the space in collaborative filtering. A profile is a set of opinionated topics. While in principle any opinionated topic could be part of the profile, we believe that only strong preferences should be relied upon in defining the profile. This can significantly alter the recommendation result.

A first, straightforward aggregation method is to consider each user individually. Let R_i be the reviews that user i has previously written. In addition, let $count(z, R)$ be the number of opinions associated with a topic z. We define the absolute importance of a topic z to user i as

$$ai(z, i) = count(z, R_i)/|R_i| \quad (1)$$

We sort the topics by their absolute importance to user i and keep the most important k_z ones in the profile Z_i: $ai(z, i) > ai(z', i), \forall z \in Z_i, \forall z' \in Z \setminus Z_i$. The choice of k_z depends on the quantity of available data and, along with the other parameters, is discussed in section 4.2.

Definition 1. *The absolute preference profile of user i, Z_i^a is the set of the k_z topics whose absolute importance in R_i is the highest.*

Alternatively, the user profile can be generated by focusing on the topics which she is interested in *more than average*. If everyone is interested in the price and the target user is also quite interested then her relative interest is low. If, however she is quite interested in free Wi-Fi and nobody else is, her relative interest is high. Let R_j be the review set of any user j in the dataset. We compute the mean importance of a topic for all the users j:

$$\overline{ai(z)} = avg_{j \in U}(ai(z, j)/|R_j|) \quad (2)$$

We then define the relative importance of a topic z for the user i as

$$ri(z, i) = ai(z, i) - \overline{ai(z)} \quad (3)$$

Definition 2. *The relative preference profile of user i, Z_i^r is the set of the k_z topics whose relative importance in R_i is the highest.*

In TPCF, after the user profiles are constructed, the recommendation step follows. For each product A evaluated for user i, it uses the reviews for product A from all users j: $r_{j,A}$, with $sr_{j,A}$ their associated numeric (also known as star) ratings. The method weighs each review $r_{j,A}$ by its topic overlap with user i's profile, $Z_{i,r_{j,A}}$. The overlap represents how many topics in user i's profile are present in connection to opinions in $r_{j,A}$. It then assigns a utility score, $TPScore$, to each product, that is the weighted mean of the respective numeric ratings $sr_{j,A}$. Finally, it recommends to user i the items with the highest $TPScore$ values.

3.2 Applicability Filters and Parameters

Even when using both the text and numeric ratings, when applied indiscriminately, the accuracy of the TPCF system is lackluster. Its performance can be improved by filtering the cases where the method has significant benefits. The manner in which the selection is made greatly impacts the recommendation quality, as we show in Sections 4.3 and 4.4. The filter has three components, all related to the reviews $r_{j,A}$ available for the product A that we want to recommend to user i. The idea behind the filter is to only include the product A in the recommendation if we can base it on enough reviews $r_{j,A}$ that are relevant from user i's perspective. Let $Z_{i,r_{j,A}}$ be the topics associated with opinions in review $r_{j,A}$ that are also in Z_i.

Definition 3. *A review $r_{j,A}$ is relevant for user i if it contains at least α opinionated topics within user i's profile: $|Z_{i,r_{j,A}}| \geq \alpha$.*

The parameter choice is discussed in section 4.2.

Filter 1: Relevance Only use reviews $r_{j,A}$ that are relevant given a relevance threshold α.

The sparseness of the data means that there can be too few reviews that have significant topic overlap with user i. In this case, weighing the review scores by topic overlap is not desirable. We define the *amount of available data* $\gamma(i, A)$ as the number of reviews $r_{j,A}$ with $|Z_{i,r_{j,A}}| \geq \alpha$.

Definition 4. *The product A is recommendable to user i, given his interest profile Z_i, if A has at least γ_0 reviews relevant to user i, $\gamma(i, A) \geq \gamma_0$.*

Filter 2: Density Only recommend products A that are recommendable given a minimum number of relevant reviews γ_0

A third factor that can make the relevance of a review $r_{j,A}$ vary wildly is its age, defined as the amount of time that passed between the moment when it was written and the present. Given that the product and service quality changes over time, recent reviews are more accurate. The downside is the reduced number of eligible reviews.

Definition 5. *A review $r_{j,A}$ is fresh given a time threshold ϕ if it was written in the last ϕ days.*

Filter 3: Freshness For product A, only use reviews $r_{j,A}$ that are fresh given a time threshold ϕ in the recommendation process.

In addition to filtering the data on which we base the recommendation decision once user i's profile is established, we define two parameters related to the profile construction.

Granularity We control the number of topics k_z in the profile , detailed in Section 3.1. Bigger corpora come with the possibility of having higher profile granularity.

Profile type We use either the absolute $ai(z,i)$ or relative $ri(z,i)$ importance of each topic z the the profile aggregation method: $\iota \in \{ai, ri\}$

Preference Strength: Let A and B be two products reviewed by user i, with numeric ratings $sr_{i,A} \neq sr_{i,B}$. The strength of the preference is the minimum rating difference between the two reviews in the pair $|sr_{i,A} - sr_{i,B}| > \delta$. TPCF was shown to have a better performance than a non personalized benchmark for strong preferences ($\delta \geq 3$), but not for lower ones. While it is more important to correctly predict stronger preferences, the value of δ is not known before the prediction. Thus it cannot be used to calibrate a recommender the method. We thus did not put any restriction δ in the analysis.

If the restrictions regarding the five parameters are met, we consider we have enough confidence in the model to generate a recommendation. In this case, when computing a recommendation for user i, we personalize using the opinion-derived profile. If not, we use a non-personalized method. The average star rating is the most commonly used one on commercial platforms: $\bar{sr}_A$

We compare with the rating aggregation because traditional CF results were even worse than random, which also happened in [Liu *et al.*, 2013].

4 Data-driven Personalization

4.1 Dataset

Product review corpora are product oriented. They typically consist of all the available reviews for a given set of products. Modeling the user preferences thus relies on the chance of having multiple products rated by the same user.

To have a solid representation of the user profiles, we take a user-centric approach, We gather all the available hotel reviews from selected users from the Tripadvisor website[2]. We gathered all the reviews authored by 50 top contributors, who each have a minimum of 50 hotel reviews written. For each hotel of the 2684 reviewed by these users, we downloaded additional reviews. Due to download speed constraints, a maximum of 500 reviews from other users were downloaded for each hotel. In total, we had 435102 reviews.

4.2 Filter Values

To show the relevance of the discussed filters, we restrain the analysis using the following constraints:

- the **relevant** reviews j must share at least $\alpha = 2$ topics with user i's profile.

[2]The corpus is available on demand.

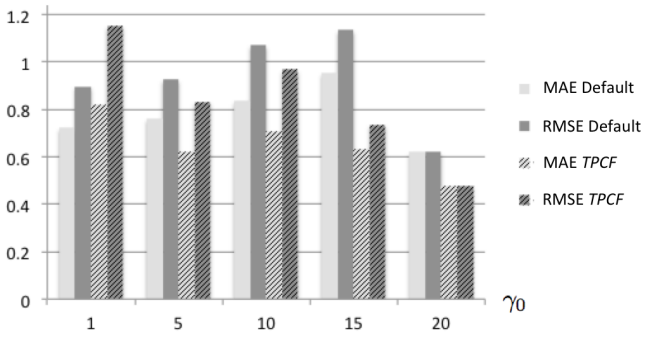

Figure 2: MAE and RMSE Improvement depending on γ_0

- the data must be **recent**: $\phi = 100$.
- the profiles are **community-dependent**, $\iota = ri$,
- each profile consists of $k_z = 3$ topics.

It is noteworthy that these parameter choices have an additive effect and that for smaller datasets, where, for instance, there are too few recent reviews, some restrictions can be relaxed. We use the remaining filter - the density γ_0 - to show the dependence of the prediction accuracy on data availability and compare our results with the ones previously reported. To be able to compare directly is the reason we ran the tests on the same target domain - hotel reviews - and why we used the same topics and α values as in [Musat *et al.*, 2013].

4.3 Rating Prediction Evaluation

The first experiments determine the rating prediction accuracy of topic profile collaborative filtering. We quantify the TPCF improvements over standard rating aggregation that ranks items according to their average rating. We show the mean average error (MAE) and the root mean squared error (RMSE). The MAE and RMSE for the default, non personalized prediction, use the average star rating of hotel A, $\bar{sr}_A$ instead of the personalized method score, in this case $TPScore_{i,A}$. It is noteworthy that the analysis can be replicated using other opinion-based aggregation mechanisms.

When applied to *all users*, without using any filters, TPCF has an MAE of 0.72, and RMSE of 0.96. It thus underperforms the benchmark, which obtains an MAE of 0.68 on the same samples. and an RMSE of 0.90. It is noteworthy, however, that these results are much better than random.

We use the filter values in 4.2 and we show the dependence of the personalized and unpersonalized MAE and RMSE scores on the relevant data density γ_0. In Figure 2 we show the comparative rating prediction results. Given parameter changes, the result for the benchmark method change as well. This is because we run the default method on exactly the same samples as TPCF, R_γ, which vary with the choice of relevant reviews.

For $\gamma_0 = 1$ - when we predict the rating even when we have a single relevant review for that product - ***TPCF*** underperforms. Let MAE_{TPCF} be the mean average error and $RMSE_{TPCF}$ the root-mean-square error of the TPCF method and $MAE_d and RMSE_d$ the corresponding values for the benchmark. $MAE_{TPCF} = 0.82$, 13.8% higher than

MAE_d, while $RMSE_{TPCF} = 1.15$, 29.2% higher than $RMSE_d$.

With an increased availability of data - for larger values of γ_0 - the performance of $TPCF$ improves, with respect to the unpersonalized method. For instance, for $\gamma_0 = 15$, we obtain $MAE_{TPCF} = 0.63$, a very significant 33.6% **lower** than MAE_d and $RMSE_{TPCF} = 0.73$, again a very significant 34.5% **lower** than $RMSE_d$. Because we only have at most 500 reviews for each hotel in the experiment, there is little data for the extreme point ($\gamma_0 = 20$). This can explain the decrease in the difference, compared to $\gamma_0 = 15$.

The rating prediction performance evolution is remarkable both in magnitude and in implications. From underperforming by 29.2% to outperforming by 34.5%, the usefulness of text based personalization is tightly related to the availability of sufficient relevant reviews. This shows that applying the measure indiscriminately is *misleading* and this is perhaps the main reason why the very useful textual information has been overlooked so far.

4.4 Ranking Evaluation

The rating prediction performance does not accurately describe the effect of a method on the user experience. Users mainly see the *relative ranking* of different items. We thus measure how well the product rankings computed with *TPCF* agree with the user's rankings, obtained from their review ratings. Kendall's τ rank correlation [Kendall, 1938] is a common measure for this quality. [Noether, 1981] argues that it is an intuitively simple measure of strength of relationship between two rankings and supports its usage instead of the Spearman correlation coefficient. It has been used extensively in social choice theory and decision theory [Dwork *et al.*, 2001].

For users that have rated several hotels, we can test if the predicted ranking agrees with the ranking known from their numeric ratings. We apply this to each user $i \in U$ for whom we have at least a pair of recommendable items with different numeric scores, $sr_{i,A} \neq sr_{i,B}$. We use the filter and parameter values discussed in Section 4.2. We test on the set of users for whom we have at least a pair of reviews for recommendable items, as specified in definition 4.

For the two products, we compute their respective $TPScore$ values and determine whether the preference prediction of A over B was successful or not. Let $sp(i)$ be the number of successfully ranked pairs and $np(i)$ the number of mistakenly ranked pairs. We define a recommender system's relevance as its aggregate performance over the analyzed users:

$$\tau = \frac{\sum_{i \in U} sp(i) - np(i)}{\sum_{i \in U} sp(i) + np(i)} \quad (4)$$

Let τ_{TPCF} be the relevance of the proposed method. We compare it to that of the default recommender, $\tau_{\bar{sr}}$. Higher τ values indicate a better performance, with a maximum of 1 reached in the absence of unsuccessful predictions. Figure 3 shows their dependency on the relevant reviews density. Here too, a higher γ_0 is associated with better performance for *TPCF*.

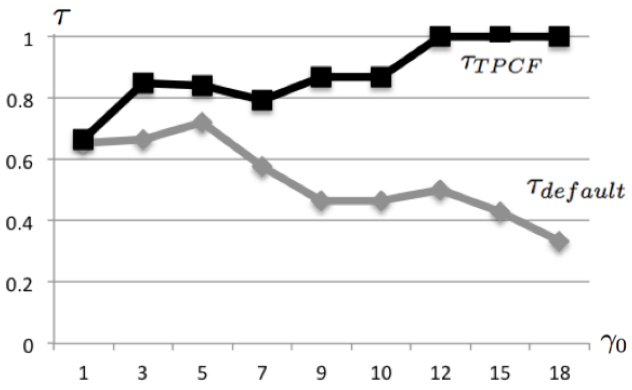

Figure 3: Ranking Improvement depending on γ_0

For the indiscriminate application of the method, with $\gamma_0 = 1$, we do worse than the non personalized method. The improvement improves steadily with higher γ_0 values, as the non personalized method does not benefit from the extra data. This result is coherent with the one obtained for rating prediction in Section 4.3.

The peak performance, in this case a perfect accuracy is reached for $\gamma_0 = 9$, where $\tau_d = 0.2$ and $\tau_{TPCF} = 1$, *five times higher* than the benchmark. The difference is much wider than for rating prediction, measured either with MAE or RMSE. However we only had so much data for five cases, which makes the final point an unreliable result. However, even for $\gamma_0 = 5$, τ_{TPCF} is more than 4 times higher than $\tau_d = 1$. The result increases in significance, as we have obtained it without recourse to the preference strength parameter δ. For this setting, in the previous results TPCF underperformed when compared to the same benchmark.

5 User-driven Personalization

Some users do not fit normal behavior. These might include spammers [Mukherjee *et al.*, 2012] and people who just like everything. A common observable feature is that they are non-opinionated. We compute a user's opinionatedness from the available review texts.

Definition 6. *We define opinionatedness of user i as the presence of at least φ% negative opinions related to the topics within their profile within a user's reviews R_i.*

The opinions are not the simple presence of a polarized words, but aspect-based opinions related to the topics used by the model. To exemplify - in Figure 1, *Kenny* is a non opinionated user, as no negative opinions are available. Eric, however, is a normal user, as we were able to retrieve a negative opinion about cleanliness. In our dataset, for $\varphi = 5$, out of the 50 top contributors 17 were not opinionated.

We test the impact of a user's opinionatedness on the recommender performance. We hypothesize that personalization based on previously expressed opinions cannot work for non opinionated users. We cannot trust their profiles, because we do not know which topic is really related to the decision making. If all the topics seem to be important, none really are.

We thus distinguish three user sets:

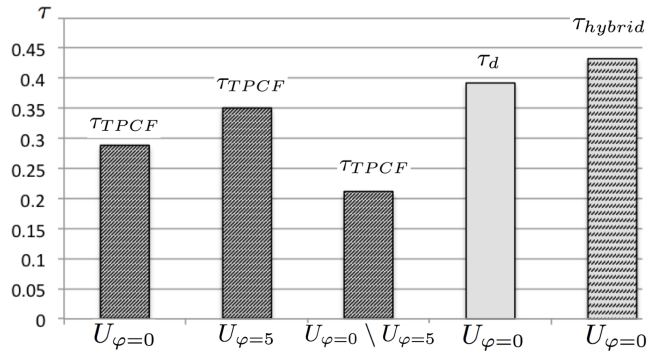

Figure 4: Blending personalized and unpersonalized recommendation

- all the users $U_{\varphi=0}$,
- opinionated users $U_{\varphi=5}$
- non opinionated users $U_{\varphi=0} \setminus U_{\varphi=5}$.

We have previously shown that when sufficient relevant data is not available, personalizing is not desirable, as it yields worse results than default aggregation. In the following experiment, we show that the method does indeed work, even in conditions of sparse data, but only for the opinionated users.

We used the parameter combination described in Section 4.2. We increased the number of topics in the profile to 9, to search for more relevant opinions. In addition, we also fix the remaining parameter, the density, and we use the most permissive setting $\gamma_0 = 1$.

Figure 4 shows the comparative results. In the first three columns of Figure 4 we show the ranking evaluation, τ_{TPCF}, for all users $U_{\varphi=0}$, opinionated ones $U_{\varphi=5}$ and non opinionated ones $U_{\varphi=0} \setminus U_{\varphi=5}$. The performance on opinionated users $\tau_{TPCF}(U_{\varphi=5})$ is 65% better than for non opinionated ones $\tau_{TPCF}(U_{\varphi=0} \setminus U_{\varphi=5})$. This translates into a significant performance boost of 21.1%, compared to $\tau_{TPCF}(U_{\varphi=0})$. This shows that the opinionatedness is a good feature to complement data density in the decision when to use opinion-based personalization.

We use this insight to create a hybrid recommender system that switches between a personalized and a non personalized method, depending on the user's opinionatedness. Switching between methods is a known method of improving the performance of recommenders [van Setten *et al.*, 2003]. We are, however, the first to switch based on the textual opinions within the reviews. In Figure 1, Kenny seems to care about family and nightlife at the same time and seems to like everything. We show that the resulting topic profile is not trustworthy and should not be used in TPCF, as for him it's better to use the benchmark method.

In the presented case, there are two advantages to switching:

- The hybrid method is applicable to all the users $U_{\varphi=0}$
- τ_{hybrid} is superior to both τ_d and τ_{TPCF}

The fourth column in Figure 4, of light grey color, shows the performance of the non personalized recommender, τ_d. It outperforms the *TPCF* by 38.8% because of the too permissive data requirements. The rightmost column in Figure 4 shows the results for the hybrid method, τ_{hybrid}, which improves τ_d by 12.7%. This result is relevant as it shows that, even for low relevant data density $\gamma_0 = 1$, opinion-based personalization helps improve the overall rankings.

6 Conclusion

Personalizing recommendations is difficult. Some methods fail because of the sparseness caused by not using all the information. Others use all the information in review texts and numeric ratings, and can still fail if applied indiscriminately. In this paper we showed how the review texts and numeric ratings can be successfully used to obtain very significant improvements over a strong benchmark when certain conditions are met. We identified two key aspects that make the difference between a mediocre and a very good performance:

- We use topic based personalization only when sufficient relevant reviews are available.
- We personalize recommendations for opinionated users but not for non opinion'ated ones.

We compared what an opinion-based personalization method can achieve with and without focusing on data quality and availability. Our improvements are significantly higher than those previously reported with the same personalization method - TPCF. To obtain them, we formalize what relevant data is, and we parametrize the applicability the method using five parameters. When the personalized recommendation is applicable, we reduced the RMSE by 34.5% and MAE by 33.6%, while previous MAE improvements were limited to 8%.

More importantly, from a ranking performance perspective, we obtained good results even without knowing the preference strength beforehand. The method even reaches a perfect accuracy, albeit on very few samples in cases where relevant reviews are abundant. This shows the importance of data. Only a little more is sufficient to achieve a much better accuracy. This can become an incentive for users to write more reviews, as they will get a superior recommendation in return.

We then showed that even in cases with few relevant reviews, we can still use personalization for *opinionated users*. We thus create a hybrid recommendation method, that switches between the personalized and unpersonalized method, depending on the user type. This hybrid method, applicable to all users, yields better results than the benchmark. We are the first to create a hybrid recommender that applies personalization based on the opinion profile of the users.

References

[Chen *et al.*, 2015] Li Chen, Guanliang Chen, and Feng Wang. Recommender systems based on user reviews: the state of the art. *User Modeling and User-Adapted Interaction*, pages 1–56, 2015.

[Dooms *et al.*, 2013] Simon Dooms, Toon Pessemier, and Luc Martens. Offline optimization for user-specific hy-

brid recommender systems. *Multimedia Tools and Applications*, pages 1–24, 2013.

[Dwork *et al.*, 2001] Cynthia Dwork, Ravi Kumar, Moni Naor, and D. Sivakumar. Rank aggregation methods for the web. In *Proceedings of the 10th International Conference on World Wide Web*, WWW '01, pages 613–622, New York, NY, USA, 2001. ACM.

[Ekstrand and Riedl, 2012] Michael Ekstrand and John Riedl. When recommenders fail: Predicting recommender failure for algorithm selection and combination. In *Proceedings of the Sixth ACM Conference on Recommender Systems*, RecSys '12, pages 233–236, New York, NY, USA, 2012. ACM.

[Jakob *et al.*, 2009] Niklas Jakob, Stefan Hagen Weber, Mark Christoph Müller, and Iryna Gurevych. Beyond the stars: exploiting free-text user reviews to improve the accuracy of movie recommendations. In *Proceedings of the 1st international CIKM workshop on Topic-sentiment analysis for mass opinion*, TSA '09, pages 57–64, New York, NY, USA, 2009. ACM.

[Kazienko and Kolodziejski, 2006] Przemyslaw Kazienko and Pawel Kolodziejski. Personalized integration of recommendation methods for e-commerce. int. *Journal of Computer Science and Applications*, 3, 2006.

[Kendall, 1938] M. G. Kendall. A New Measure of Rank Correlation. *Biometrika*, 30(1/2):81–93, June 1938.

[Levi *et al.*, 2012] Asher Levi, Osnat Mokryn, Christophe Diot, and Nina Taft. Finding a needle in a haystack of reviews: cold start context-based hotel recommender system. In *Proceedings of the sixth ACM conference on Recommender systems*, RecSys '12, pages 115–122, New York, NY, USA, 2012. ACM.

[Liu *et al.*, 2013] Hongyan Liu, Jun He, Tingting Wang, Wenting Song, and Xiaoyang Du. Combining user preferences and user opinions for accurate recommendation. *Electronic Commerce Research and Applications*, 12(1):14 – 23, 2013.

[McAuley and Leskovec, 2013] Julian McAuley and Jure Leskovec. Hidden factors and hidden topics: Understanding rating dimensions with review text. In *Proceedings of the 7th ACM Conference on Recommender Systems*, RecSys '13, pages 165–172, New York, NY, USA, 2013. ACM.

[Mukherjee *et al.*, 2012] Arjun Mukherjee, Bing Liu, and Natalie Glance. Spotting fake reviewer groups in consumer reviews. In *Proceedings of the 21st International Conference on World Wide Web*, WWW '12, pages 191–200, New York, NY, USA, 2012. ACM.

[Musat and Faltings, 2013] Claudiu Cristian Musat and Boi Faltings. A novel human computation game for critique aggregation. In *AAAI*, 2013.

[Musat *et al.*, 2013] Claudiu-Cristian Musat, Yizhong Liang, and Boi Faltings. Recommendation using textual opinions. In *Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence*, IJCAI'13, pages 2684–2690. AAAI Press, 2013.

[Noether, 1981] Gottfried E Noether. Why kendall tau. *Teaching Statistics, 3(2), 41-43.*, 1981.

[Sill *et al.*, 2009] Joseph Sill, Gábor Takács, Lester Mackey, and David Lin. Feature-weighted linear stacking. *CoRR*, abs/0911.0460, 2009.

[Snyder and Barzilay, 2007] Benjamin Snyder and Regina Barzilay. Multiple aspect ranking using the good grief algorithm. In *In Proceedings of the Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics (HLT-NAACL*, pages 300–307, 2007.

[van Setten *et al.*, 2003] M. van Setten, M. Veenstra, E. M. A. G. van Dijk, and A. Nijholt. Prediction strategies in a tv recommender system - method and experiments. In P. Isaísas and N. Karmakar, editors, *Proceedings IADIS International Conference WWW/Internet 2003, Algarve, Portugal*, pages 203–210, Lisbon, Portugal, November 2003. IADIS.

[Zhang *et al.*, 2013] Weishi Zhang, Guiguang Ding, Li Chen, Chunping Li, and Chengbo Zhang. Generating virtual ratings from chinese reviews to augment online recommendations. *ACM Trans. Intell. Syst. Technol.*, 4(1):9:1–9:17, February 2013.