Additive Smoothing for Relevance-Based Language Modelling of Recommender Systems

Daniel Valcarce, Javier Parapar, Álvaro Barreiro
Information Retrieval Lab
Department of Computer Science
University of A Coruña, Spain
{daniel.valcarce, javierparapar, barreiro}@udc.es

ABSTRACT

The use of Relevance-Based Language Models for top-N recommendation has become a promising line of research. Previous works have used collection-based smoothing methods for this task. However, a recent analysis on RM1 (an estimation of Relevance-Based Language Models) in document retrieval showed that this type of smoothing methods demote the IDF effect in pseudo-relevance feedback. In this paper, we claim that the IDF effect from retrieval is closely related to the concept of novelty in recommendation. We perform an axiomatic analysis of the IDF effect on RM2 concluding that this kind of smoothing methods also demotes the IDF effect in recommendation. By axiomatic analysis, we find that a collection-agnostic method, Additive smoothing, does not demote this property. Our experiments confirm that this alternative improves the accuracy, novelty and diversity figures of the recommendations.

CCS Concepts

•Human-centered computing → **Collaborative filtering; •Information systems** → *Recommender systems;* Language models;

Keywords

Recommender systems, Relevance-Based Language Models, collaborative filtering

1. INTRODUCTION

Recommender systems are becoming increasingly popular these days. The enormous growth of data available to users has changed the way we access information. Users are becoming more and more demanding: they are eager to receive personalised contents instead of explicitly stating their information needs. Therefore, the applicability of recommender systems is undeniable. This technology is designed for providing relevant items of information by learning from the users' past behaviour. Continuous developments in this field have been made to meet these high expectations.

We can distinguish multiple approaches to recommendation [24]. They are often classified in three main categories: content-based, collaborative filtering and hybrid techniques. Content-based approaches generate recommendations based on the item and user descriptions: they suggest items similar to those liked by the target user [9]. In contrast, collaborative filtering methods rely on the interactions (typically ratings) between users and items in the system [21]. Finally, there exist hybrid algorithms that combine both collaborative filtering and content-based approaches.

Traditionally, Information Retrieval (IR) has focused on delivering the information that users demand [1]. On the other hand, Information Filtering (IF) explores ways of selecting relevant pieces of information from a stream of data [12]. Recommender systems are active information filters: these methods learn from the users' behaviour providing personalised suggestions. Given the similarities between IR and IF, some authors have considered them to be sibling fields or *two sides of the same coin* [2]. The main difference between these fields is the way in which the information need is obtained. Information Retrieval systems usually receive a query prompted by the user meanwhile Information Filtering systems infer the users' needs.

Due to the closeness of IR and IF, exploiting Information Retrieval methods for recommenders systems has become a fertile area of research [31, 4, 23, 27, 30]. In particular, in this paper we want to further investigate the use of Relevance-Based Language Models for recommendation. Lavrenko and Croft [17] devised the Relevance-Based Language Modelling framework for the pseudo-relevance feedback task proposing two methods: RM1 and RM2. Later, Parapar et al. adapted this technique to the collaborative filtering scenario achieving high figures of precision [23]. For pseudo-relevance feedback, RM1 is the preferred method; however, RM2 yields better results than RM1 in top-N recommendation.

To be effective, language models employ smoothed probability estimates. The selection of the smoothing method is crucial for the performance of language models both in Information Retrieval [33] and in recommendation [29]. A recent study presented an axiomatic analysis of smoothing methods for different pseudo-relevance feedback techniques (including RM1) [13]. The authors concluded that the traditional collection-based methods that have been used for smoothing language models in the ad hoc retrieval task [33] are not suitable for performing pseudo-relevance feedback

CERI '16, June 14 - 16, 2016, Granada, Spain

ISBN 978-1-4503-4141-7/16/06...$15.00

DOI: http://dx.doi.org/10.1145/2934732.2934737

because they demote the IDF (inverse document frequency) effect. Instead, they proposed the use of RM1 with Additive smoothing, a collection-agnostic smoothing method. This choice is supported by their axiomatic analysis and outperformed traditional alternatives in their experiments.

The IDF is a measure of term specificity [26, 25]. We claim that this concept is related to novelty in recommender systems [7]. In this paper, we study the connection between these two concepts and its implications in recommendation. Furthermore, we perform an axiomatic analysis of the IDF effect in RM2 in the context of recommendation. We study three collection-based smoothing methods (Jelinek-Mercer, Dirichlet Priors and Absolute Discounting) and a collection-agnostic method (Additive smoothing). Our goal is to determine if the application of Additive smoothing to Relevance-Based Language Models is valuable not only for the pseudo-relevance feedback task, but also for top-N recommendation. Our axiomatic analysis proves that the aforementioned collection-based smoothing methods demote the IDF effect on RM2 in recommendation as it does on RM1 for pseudo-relevance feedback. Moreover, we find that Additive smoothing neither promotes nor demotes the IDF effect on RM2. Thus, if the IDF heuristic is valuable for recommendation, Additive smoothing should work better than the other methods.

To verify this last assumption—the utility of the IDF effect in recommendation—we test experimentally the quality of the recommendations generated by RM2 with different smoothing methods. We use accuracy, diversity and novelty metrics to cover different aspects of top-N recommendation. Our experiments show that Additive smoothing provides better figures of accuracy, diversity and novelty and, at the same time, it is more stable with respect to the smoothing parameter than the collection-based smoothing methods.

In summary, the contributions of this paper are (1) an investigation of the relationship between the IDF effect in retrieval and novelty in recommendation, (2) an axiomatic analysis of the IDF effect of RM2 using the three most popular collection-based smoothing methods and Additive smoothing and (3) an empirical comparison of the four studied smoothing methods in terms of accuracy, novelty and diversity of the recommendations concluding that Additive smoothing is the method that provides best figures for these metrics.

2. BACKGROUND

In this section, first, we describe the top-N recommendation task and the notation. Second, we contextualise the Relevance-Based Language Modelling approach in its original task (i.e., pseudo-relevance feedback) and, then, we describe its adaptation to top-N recommendation. Finally, we present different smoothing strategies for language models in the context of recommendation.

2.1 Top-N Recommendation

Top-N recommendation refers to the task of finding the top-N most relevant items for a user [8]. This approach contrasts with the rating prediction task where a recommender system is supposed to predict the values of the ratings that the users will give to the items. Several authors have argued that the top-N recommendation task is more realistic than rating prediction [14, 8, 11, 24]. Therefore, this work is devoted to the top-N recommendation problem.

Recommender systems work with a set of users $\mathcal{U}$ and a set of items $\mathcal{I}$. Collaborative filtering approaches employ the interactions between users and items to generate recommendations. This work is devoted to explicit feedback collaborative filtering techniques which are based on ratings. A rating from a user $u \in \mathcal{U}$ to an item $i \in \mathcal{I}$ is denoted by $r(u, i)$. Additionally, $\mathcal{I}_u$ refers to the set of items rated by the user u. Finally, the recommendation list for the user u of length k is represented by L_u^k.

2.2 Relevance-Based Language Models

Relevance-based Language Models (often abbreviated as Relevance Models or simply RM) are a state-of-the-art technique for the pseudo-relevance feedback task in text retrieval [17]. Nonetheless, this framework has also been adapted to top-N recommendation with great success [23].

Pseudo-relevance feedback (PRF) is a way of expanding queries with new terms in a text retrieval system. Since a standard retrieval system returns a list of documents according to the query prompted by the user, the performance of this process depends fundamentally on the retrieval algorithm and the quality of the user's query [1]. For this reason, expanding this query with relevant terms is a way of improving the outcome of a retrieval system. Pseudo-relevance feedback is an automatic and effective query expansion approach. In PRF, we assume that the top documents retrieved with the original query are relevant (they form the pseudo-relevant set). PRF methods expand the initial query with the most relevant terms from the pseudo-relevant set. Then, the expanded query is used for performing a second retrieval which provides the results to be presented to the user.

The PRF approach has been adapted to collaborative filtering recommendation [23] in the following way. Instead of a query, we have a user whose profile (i.e., the set of items that the user has rated) has to be expanded with new relevant items. For doing so, we use a set of neighbours or similar users. The intuitive idea is that the candidates for recommendation are those items that are relevant in the neighbourhood of the user. In this way, users play a dual role: they act as queries when they are the target user of the recommendation process but they also act as documents of the pseudo-relevant set when they are considered as neighbours. On the other hand, items only play the role of terms in retrieval.

There exists two estimations of Relevance Models—RM1 and RM2—which differ in the probability assumptions they make [17]. In this work, we employed the latter estimation because it outperformed the former one in the collaborative filtering scenario [23]. In RM2, the probability of an item i under the Relevance Model of the user u is given by:

$$p(i|R_u) \propto p(i) \prod_{j \in \mathcal{I}_u} \sum_{v \in V_u} \frac{p(i|v)\, p(v)}{p(i)}\, p(j|v) \qquad (1)$$

Recommendations are presented to the user ordered according to decreasing relevance, that is, decreasing values of $p(i|R_u)$. We consider the prior probabilities $p(i)$ and $p(v)$ to be uniform for the sake of simplicity.

In addition, V_u is the set of neighbours of user u. These neighbourhoods are computed using a clustering technique. Following a common practice in the literature, we decided to use k-NN algorithm [8, 21]. This method consists in finding the k most similar users to the target user using a pairwise metric. In this work, we used cosine similarity for this purpose:

$$\text{cosine}(u, v) = \frac{\sum_{i \in \mathcal{I}_u \cap \mathcal{I}_v} r(u,i)\, r(v,i)}{\sqrt{\sum_{i \in \mathcal{I}_u} r(u,i)^2}\sqrt{\sum_{i \in \mathcal{I}_v} r(v,i)^2}} \tag{2}$$

Finally, it only remains to describe the estimation of $p(i|v)$ which is based on the maximum likelihood estimate (MLE) over a multinomial distribution of positive ratings:

$$p_{ml}(i|u) = \frac{r(u,i)}{\sum_{j \in \mathcal{I}_u} r(u,j)} \tag{3}$$

The problem of the maximum likelihood estimate stems from its high sparsity: if a user did not rate an item, the estimate yields a value of zero. For this reason, language models are smoothed. In text retrieval, the most common practice is to smooth the MLE with the collection model [33]. Likewise, in recommendation, collection-based smoothing methods have been applied to Relevance Models [29]. Next, we present the most common collection-based smoothing methods for addressing the sparsity problem of MLE.

2.3 Collection-Based Smoothing Methods

Using a collection-based smoothing method, the probability of an item given a user $p(i|u)$ is calculated by smoothing the maximum likelihood estimate $p_{ml}(i|u)$ with the background model of the collection $p(i|\mathcal{C})$. This collection model is given by:

$$p(i|\mathcal{C}) = \frac{\sum_{v \in \mathcal{U}} r(v,i)}{\sum_{j \in \mathcal{I},\, v \in \mathcal{U}} r(v,j)} \tag{4}$$

For text retrieval, the prominent smoothing methods are Jelinek-Mercer and Dirichlet Priors [33]. In contrast, it has been shown that Absolute Discounting works better than the previous methods when Relevance Models are applied to collaborative filtering [29].

2.3.1 *Jelinek-Mercer (JM)*

This method performs a linear interpolation between the maximum likelihood estimator and the collection model [16] which is regulated by the parameter $\lambda \in [0, 1]$:

$$p_\lambda(i|u) = (1 - \lambda)\, p_{ml}(i|u) + \lambda\, p(i|\mathcal{C}) \tag{5}$$

2.3.2 *Dirichlet Priors (DP)*

DP is derived from a Bayesian analysis using Dirichlet priors [18]. It has a parameter $\mu > 0$ to control the amount of smoothing applied:

$$p_\mu(i|u) = \frac{r(u,i) + \mu\, p(i|\mathcal{C})}{\mu + \sum_{j \in \mathcal{I}_u} r(u,j)} \tag{6}$$

2.3.3 *Absolute Discounting (AD)*

AD subtracts a value of $\delta > 0$ from the count of the rated items [20]. This discount is compensated with the background collection:

$$p_\delta(i|u) = \frac{\max[r(u,i) - \delta, 0] + \delta\, |\mathcal{I}_u|\, p(i|\mathcal{C})}{\sum_{j \in \mathcal{I}_u} r(u,j)} \tag{7}$$

Collection-based smoothing methods have in common that they substitute part of the probability mass provided by the MLE (Eq. 3) with probability mass obtained from the collection model (Eq. 4). This reallocation of probability is performed to avoid zeroes in non-rated items [33]. However, this is done in a way that popular items in the collection receive more probability than those less common. Intuitively, this type of smoothing may reduce novelty. In fact, previous studies found that RM2, compared to state-of-the-art recommenders, do not have very good results of novelty and diversity [27]. Therefore, in the next section, we explore alternatives to collection-based smoothing in order to improve novelty figures.

3. IDF EFFECT AND NOVELTY ON RM

Novelty and diversity have been two important aspects in text retrieval, to the point that several TREC tracks and tasks have been devoted to them[1]. However, regarding the recommendation task, there has been little interest in novelty or diversity until the beginning of the 2000s [7]. Accuracy, particularly measured with error metrics, was the primary objective of any recommender. For example, the Netflix Prize goal was to improve the accuracy of Cinematch (Netflix recommendation system) by 10% [5]. Nowadays, research efforts have moved from the rating prediction task to the top-N recommendation task [14, 8]. Additionally, there exist a consensus on the critical importance of measuring different properties of recommenders systems such as diversity and novelty [14, 15, 7].

On the other hand, in the Information Retrieval community, recent criticism has been raised to the use of collection-based smoothing methods in the context of pseudo-relevance feedback. Hazimeh and Zhai analysed the effect that this type of smoothing has on three PRF techniques [13]. In particular, they found that applying collection-based smoothing methods to RM1 conflicts with the IDF effect—a desired property of a retrieval system.

We claim that there exists a connection between the concept of novelty from recommendation and the IDF effect from Information Retrieval. In this section, we describe this relationship. We start by defining novelty and diversity in recommendation and, then, we present the IDF effect and its similarities with novelty.

3.1 Novelty and Diversity

Novelty has been studied in Information Retrieval as the proportion of relevant documents in the result set that are unknown to the user [1]. Since this definition is not very pragmatic when using top-N recommenders in a collaborative filtering scenario, novelty is usually measured as how unusual the recommended items are [15]. Diversity, on the other hand, measures whether a recommender systems suggest different items or, on the contrary, it recommends mostly the same items [7]. Both properties, novelty and diversity, are closely connected and to some degree complementary [7].

The rationale behind the importance of novelty is that an accurate recommendation can be useless if the suggested items are already known by the user. Recommendations should also try to suggest items that users would not have discovered by themselves. However, this property, called serendipity, is difficult to measure. Thus, it is usually approximated by novelty and relevance [7]. Somehow, novelty is a similar concept to serendipity but weaker: novel recommendations provide the users with information about uncommon items, although these items could have been discovered even-

[1] http://trec.nist.gov/tracks.html

tually. Recommendations are considered diverse when they suggest a great variety of items instead of recommending a few popular ones.

Recommender systems that strongly focus on accuracy may give poor results on diversity and novelty metrics and vice versa. Intuitively, we can see that if we recommend to the users the most popular items for their similar neighbours, the suggestions will be accurate but diversity and novelty will suffer. On the contrary, recommending unusual items can improve novelty and diversity at the risk of making some mistaken suggestions. This is perhaps the most prominent trade-off in the field of recommender systems [34].

3.2 IDF Effect

In Information Retrieval, the inverse document frequency (IDF) is a measure of term specificity [26, 25]. It is defined as the inverse of the number of documents in the collection that contains the target term. For example, *stopwords* are terms that appear in almost every document and they do not provide much information. In contrast, those terms that only appear in a few documents are highly informative and help in discriminating which documents are relevant. Thus, the IDF effect gives more importance to those query terms that are more specific (i.e., higher IDF).

IDF was not born from a formal analysis, however it was considered a useful and robust heuristic [26]. Later, Robertson provided a theoretical justification for this term weighting function [25]. Mostly all the text retrieval algorithms introduce the IDF effect to weight query terms [26]. This property can be included in the retrieval model either explicitly (e.g., the vector space model or BM25 [1]) or implicitly (e.g., the probabilistic model [26] or language models [33]).

We claim that when adapting the Relevance Modelling framework to collaborative filtering [23], term specificity is related to item novelty. The IDF effect promotes specific terms over popular—and to some extent meaningless—ones. Since items play the role of terms when using Relevance Models for recommendation, promoting uncommon terms will be beneficial for improving novelty figures.

Previous work has explored different estimations of Relevance Models that promote divergent terms with great success [6, 22]. Another work in this line of research was the study of Hazimeh and Zhai [13]. They performed an axiomatic analysis of the IDF effect in several pseudo-relevance feedback methods. They found that collection-based methods (those from Sec. 2.3) penalise the IDF effect on RM1. To overcome this problem, they propose to use Additive smoothing which does not rely on a background collection model. Their analysis showed that this type of smoothing neither promotes nor demotes the IDF effect. However, it is not clear whether this conclusion is applicable to RM2. For this reason, we present an axiomatic analysis of the IDF effect on RM2 for recommender systems.

4. AXIOMATIC ANALYSIS OF RM2

In this section, we study the IDF effect on RM2. We performed an axiomatic analysis of the IDF effect on RM2. Our goal is to examine if RM2, in the context of top-N recommendation, penalises the IDF effect as RM1 with collection-based smoothing methods does in the pseudo-relevance feedback task [13].

In recommendation, given two items with the same ratings in the neighbourhood, the IDF effect promote the item that is less popular in the collection (in terms of probability, see Eq. 4). This property is desirable in order to enhance the novelty of the recommendations while keeping high accuracy. This effect does not conflicts with accuracy because uncommon items are preferred over common items only when they have the same ratings.

Formally, we can define the IDF effect for recommendation as follows:

Definition (IDF effect)**.** Let u be a user from the set of users $\mathcal{U}$ and V_u be her/his neighbourhood. Given two items i_1 and i_2 with the same ratings $r(v, i_1) = r(v, i_2)\ \forall\, v \in V_u$ and different popularity $p(i_1|C) < p(i_2|C)$, a recommender system that outputs $p(i_1|R_u) > p(i_2|R_u)$ is said to support the IDF effect.

Now we proceed to analyse axiomatically RM2. If we assume that i_1 and i_2 are two items as in the previous definition, studying the sign of $\Delta = p(i_1|R_u) - p(i_2|R_u)$ allows to check whether RM2 supports the IDF effect or not. If $\Delta > 0$, the recommender system supports this property. On the contrary, if $\Delta < 0$, the algorithm violates the definition of the IDF effect. Finally, $\Delta = 0$ means that the system neither promotes nor demotes the IDF effect. Given the formula of RM2, Δ is computed as follows:

$$\begin{aligned}\Delta &= p(i_1|R_u) - p(i_2|R_u)\\ &= p(i) \prod_{j \in \mathcal{I}_u} \sum_{v \in V_u} \frac{p(i_1|v)\, p(v)}{p(i)}\, p(j|v)\\ &\quad - p(i_2) \prod_{j \in \mathcal{I}_u} \sum_{v \in V_u} \frac{p(i_2|v)\, p(v)}{p(i_2)}\, p(j|v) \qquad (8)\end{aligned}$$

If we suppose that item priors are uniform, $p(i) = |\mathcal{I}|^{-1}$, we obtain:

$$\begin{aligned}\Delta = p(i) &\prod_{j \in \mathcal{I}_u} \sum_{v \in V_u} \frac{p(j|v)\, p(v)}{p(i)}\, p(i_1|v)\\ &- p(i) \prod_{j \in \mathcal{I}_u} \sum_{v \in V_u} \frac{p(j|v)\, p(v)}{p(i)}\, p(i_2|v) \qquad (9)\end{aligned}$$

We can observe that the sign of Δ depends on the sign of $p(i_1|v) - p(i_2|v)$ which may vary among smoothing methods. Therefore, we need to analyse each smoothing technique one by one. Next, we examine the three collection-based methods described Sec. 2.3. Moreover, we present and study Additive smoothing, a collection-agnostic method, as a possible alternative to the traditional ones.

4.1 Analysis of Jelinek-Mercer

Applying Jelinek-Mercer smoothing from Eq. 5 to Eq. 9:

$$\begin{aligned}\Delta = p(i) &\prod_{j \in \mathcal{I}_u} \sum_{v \in V_u} \frac{p(j|v)p(v)}{p(i)} \big[(1-\lambda) p_{ml}(i_1|v) + \lambda p(i_1|\mathcal{C})\big]\\ &- p(i) \prod_{j \in \mathcal{I}_u} \sum_{v \in V_u} \frac{p(j|v)p(v)}{p(i)} \big[(1-\lambda) p_{ml}(i_2|v) + \lambda p(i_2|\mathcal{C})\big]\\ &< 0 \qquad (10)\end{aligned}$$

we obtain that the difference is negative because $\lambda \in [0, 1]$, all the probabilities are positive and $p(i_1|C) < p(i_2|C)$ from definition. Note that $p_{ml}(i_1|u) = p_{ml}(i_2|u)$ because both items have the same ratings. Thus, Jelinek-Mercer demotes the IDF effect for RM2 as it does for RM1 in pseudo-relevance feedback [13].

4.2 Analysis of Dirichlet Priors

Plugging DP smoothing method from Eq. 6 into Eq. 9:

$$\begin{aligned}\Delta = p(i) &\prod_{j\in\mathcal{I}_u}\sum_{v\in V_u}\frac{p(j|v)p(v)}{p(i)}\frac{r(v,i_1)+\mu\, p(i_1|\mathcal{C})}{\mu+\sum_{k\in\mathcal{I}_u} r(v,k)}\\ -\, p(i)&\prod_{j\in\mathcal{I}_u}\sum_{v\in V_u}\frac{p(j|v)p(v)}{p(i)}\frac{r(v,i_2)+\mu\, p(i_2|\mathcal{C})}{\mu+\sum_{k\in\mathcal{I}_u} r(v,k)}\\ < 0 & \end{aligned} \tag{11}$$

we obtain that the difference is also negative because $\mu > 0$, all the ratings and probabilities are positive and, by definition, $p(i_1|C) < p(i_2|C)$. We can conclude that Dirichlet Priors violates the IDF effect for RM2. This also happens for RM1 in pseudo-relevance feedback [13].

4.3 Analysis of Absolute Discounting

Absolute Discounting was not studied in the context of pseudo-relevance feedback; however, since it is the preferred method in recommendation [29], we analyse if it supports the IDF effect:

$$\begin{aligned}\Delta = p(i) &\prod_{j\in\mathcal{I}_u}\sum_{v\in V_u}\frac{p(j|v)p(v)}{p(i)}\frac{r_\delta(v,i_1)+\delta\,|\mathcal{I}_v|\, p(i_1|\mathcal{C})}{\sum_{k\in\mathcal{I}_v} r(u,k)}\\ -\, p(i)&\prod_{j\in\mathcal{I}_u}\sum_{v\in V_u}\frac{p(j|v)p(v)}{p(i)}\frac{r_\delta(v,i_2)+\delta\,|\mathcal{I}_v|\, p(i_2|\mathcal{C})}{\sum_{k\in\mathcal{I}_v} r(u,k)}\\ < 0 & \end{aligned} \tag{12}$$

where $r_\delta(v,i) = \max[r_\delta(v,i)-\delta, 0]$. We can observe that the difference Δ is negative taking into account that $\delta > 0$, $|\mathcal{I}_v| > 0$, all the ratings are positive and, by definition, $p(i_1|C) < p(i_2|C)$.

We can observe that the three collection-based smoothing methods demote the IDF effect on RM2 for recommendation. For this reason, next we also explore Additive smoothing as a collection-agnostic smoothing method.

4.4 Analysis of Additive Smoothing

Additive smoothing (also known as Laplace smoothing) is a collection-agnostic method. It increases all the ratings by a parameter $\gamma > 0$. If the user u has not rated the item i, that item will receive a rating value of γ. The probability estimate with this technique is computed as follows:

$$p_\gamma(i|u) = \frac{r(u,i)+\gamma}{\sum_{j\in\mathcal{I}_u} r(u,j)+\gamma|\mathcal{I}|} \tag{13}$$

Since this method is collection-agnostic, it does not rely on the probability of an item in the collection, $p(i|\mathcal{C})$, which is a measure of item popularity—opposed to novelty. Applying the same axiomatic analysis as before:

$$\begin{aligned}\Delta = p(i) &\prod_{j\in\mathcal{I}_u}\sum_{v\in V_u}\frac{p(j|v)p(v)}{p(i)}\frac{r(u,i_1)+\gamma}{\sum_{j\in\mathcal{I}_u} r(u,j)+\gamma|\mathcal{I}|}\\ -\, p(i)&\prod_{j\in\mathcal{I}_u}\sum_{v\in V_u}\frac{p(j|v)p(v)}{p(i)}\frac{r(u,i_2)+\gamma}{\sum_{j\in\mathcal{I}_u} r(u,j)+\gamma|\mathcal{I}|}\\ = 0 & \end{aligned} \tag{14}$$

we find that this method neither supports nor violates the IDF effect. This result coincides with the analysis of RM1 for pseudo-relevance feedback [13].

Table 1: Datasets statistics

Dataset	Users	Items	Ratings
MovieLens 100k	943	1,682	100,000
MovieLens 1M	6,040	3,706	1,000,209

5. EXPERIMENTS

Next, we present the empirical evaluation of Relevance Models with Additive smoothing applied in a collaborative filtering scenario. We used the MovieLens 100k and MovieLens 1M collections[2]. These datasets were extracted from a film recommendation platform. Their statistics are presented in Table 1.

We used the training/test splits provided by the MovieLens 100k collection. Since the other dataset do not offer a default partition, we split the dataset in the following manner: 80% of ratings of each user are for training and the rest for test.

5.1 Evaluation Methodology

Traditionally, Recommenders Systems intended to predict the ratings of unknown items. Thus, error measures such as Root Mean Squared Error (RMSE) or Mean Average Error (MAE) were commonly used to assess the performance of the recommenders [11]. Acknowledging that evaluating recommender systems using error metrics does not lead to better recommenders, several studies proposed the use of IR metrics for evaluating the rankings of recommendations (i.e., top-N recommendation) as well as diversity and novelty metrics [14, 8, 11, 7].

In this work, all the metrics are evaluated at a given cutoff rank, that is, taking into account only the top k recommendations computed by the recommender. The rationale for this decision is that users seldom consider more recommendations than the first ones. Thus, we are interested in studying the quality of the top suggestions generated by the system.

5.1.1 Accuracy

In contrast with error measures, top-N recommendation metrics analyse the quality of the recommendation lists. To apply these metrics, we followed the *TestItems* approach described by Bellogín et al. [3]. For each user, we computed a ranking composed of all the items having a test rating by some user and no training rating by the target user. The set of relevant items for the user u consists of all the items rated by the user u in the test set that have a rating greater than or equals to 3. As it has been acknowledged, considering non-rated items as irrelevant may underestimate the true metric value; however, it provides a better estimation of the recommender quality [19].

We used normalised discounted cumulative gain (nDCG) for measuring the quality of the recommendation list. In addition, this metric takes into account graded relevance (i.e., a higher rating is preferred) and position (i.e., relevant suggestions in the top positions are better than in bottom places). We used the *standard formulation* as described in [32]. Adapted to collaborative filtering, this metric is com-

[2] http://grouplens.org/datasets/movielens

puted as follows:

$$nDCG@k = \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} \frac{DCG_u@k}{IDCG_u@k} \tag{15}$$

where the Discounted Cumulative Gain is defined as:

$$DCG_u@k = \sum_{i=1}^{k} \frac{rel(u, L_u^k[i])}{\log_2(i+1)} \tag{16}$$

and $IDCG_@k$ (Ideal DCG at k for user u) refers to the maximum possible DCG till position k for the user u. It is computed calculating the $DCG_u@k$ of the perfect ranked list of items that can be recommended to user u. Last, $rel(u, L_u^k[i])$ is the graded relevance, for the user u, of the item located in the i-th position in the ranking list L_u^k. We consider the graded relevance to be equal to the rating if it is superior or equals to 3; otherwise, the relevance will be zero.

5.1.2 Diversity

We measured diversity with the Gini index. This coefficient is commonly used for quantifying wealth distribution inequalities, but it has also been utilised for measuring recommendation diversity [10, 11, 7]. Note that we use the complement of this metric for convenience. In this way, when the index is 0, it indicates that a single item is recommended for every user which corresponds to the minimum diversity scenario. On the contrary, a value of 1 means that all the items are equally recommended across the users. The Gini index is computed as follows:

$$Gini@k = 1 - \frac{1}{|\mathcal{I}| - 1} \sum_{j=1}^{|\mathcal{I}|} (2j - |\mathcal{I}| - 1)\, p(i_j|rec@k) \tag{17}$$

where $i_1, \cdots, i_{|\mathcal{I}|}$ is the list of items sorted by increasing $p(i_j|rec@k)$. This term refers to the probability that item i_j is being recommended in some recommendation list of length k and is given by:

$$p(i|rec@k) = \frac{|\{u \in \mathcal{U} | i \in L_u^k\}|}{\sum_{u \in |\mathcal{U}|} |L_u^k|} \tag{18}$$

5.1.3 Novelty

We measured novelty using an Information Theoretic metric. Zhou et al. proposed to use the mean self-information to quantify the ability of a recommender system to generate unexpected recommendations [34]. This metric is also called *surprisal* because it measures the improbability of an outcome. We use the concept of popularity in this metric, that is, the proportion of users that interacted with the item.

$$MSI@k = \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} \sum_{i \in L_u^k} \log \frac{1}{pop(i)} \tag{19}$$

where the popularity of an item is computed as:

$$pop(i) = \frac{|\{u \in \mathcal{U} | i \in \mathcal{I}_u\}|}{|\mathcal{U}|} \tag{20}$$

Please note, that this definition of popularity is equivalent to the document frequency in Information Retrieval when replacing users by documents and items by terms. Therefore, with this metric we can measure the IDF effect which, as we argued before, is directly related to novelty.

5.2 Results and Discussion

We tested the four different smoothing methods on the MovieLens 100k and MovieLens 1M datasets. We tuned the smoothing parameters $\gamma \in \{0.001, 0.01, 0.1, 1, 10\}$, $\delta \in \{0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0\}$, $\lambda \in \{0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1\}$ and $\mu \in \{100, 200, 300, 400, 500, 600, 700, 800, 900, 1000\}$ of the Additive, AD, JM and DP methods, respectively. We used k-NN algorithm with cosine similarity for computing neighbourhoods. We set $k = 50$ for the MovieLens 100k and $k = 100$ for the MovieLens 1M. The results in terms of nDCG@10, Gini@10 and MSI@10 are presented in Figs. 1 and 2 for the MovieLens 100k and 1M, respectively.

Smoothing is a crucial aspect of RM2: the choice of the smoothing method as well as its correct parameter optimisation notably affects the final quality of the recommendations. Although the absolute values of performance vary, the trends in both datasets are very similar. This supports the generalisation of these results to other collections.

Overall, additive smoothing provides the best recommendations in terms of precision, diversity and novelty followed by Absolute Discounting. Previous work reported that AD outperformed DP and JM in terms of accuracy [29]. However, this is the first study that also presents diversity and novelty figures of these methods.

In recommendation, there is always a trade-off between accuracy and diversity or novelty [34]. It is possible to improve the diversity or novelty of the recommendations at the expense of a reduction of accuracy (e.g., recommending very unpopular items to different users). Therefore, simultaneous improvements in accuracy and in novelty or diversity are highly valuable. Additive smoothing obtains notable improvements on these three aspects. This confirms the importance of the IDF effect in recommendation.

To further explore this trade-off, we plot the geometric mean of nDCG@10, Gini@10 and MSI@10. We used the geometric mean because the arithmetic mean does not address different scales properly. Figure 3 shows that Additive smoothing has the best trade-off among accuracy, diversity and novelty. In general, we can set γ between 0.001 and 0.01 to obtain good results.

An important property of Additive smoothing is the stability to the changes in its parameter. We used a logarithmic scale to visualise very large variations of parameter γ. This method only showed a small decrease in accuracy, novelty and diversity when we used enormous values of γ. AD also showed quite stable results, as stated in previous work [29], but the method deteriorates with high amount of smoothing.

In contrast, the performance of Jelinek-Mercer and Dirichlet Priors is far lower than the rest. Additionally, the computational complexity of the three collection-based methods is the same. Thus, there is no reason to consider using these smoothing methods with RM2 for recommendation.

Additive smoothing also presents another advantage over AD (and also over JM and DP): it does not depend on collection statistics. This fact not only preserves the IDF effect but also reduces the computational resources required to maintain global statistics of the collection.

6. CONCLUSIONS AND FUTURE WORK

Recommender systems have greatly improved accuracy results in the last years. Recently, research has focused on

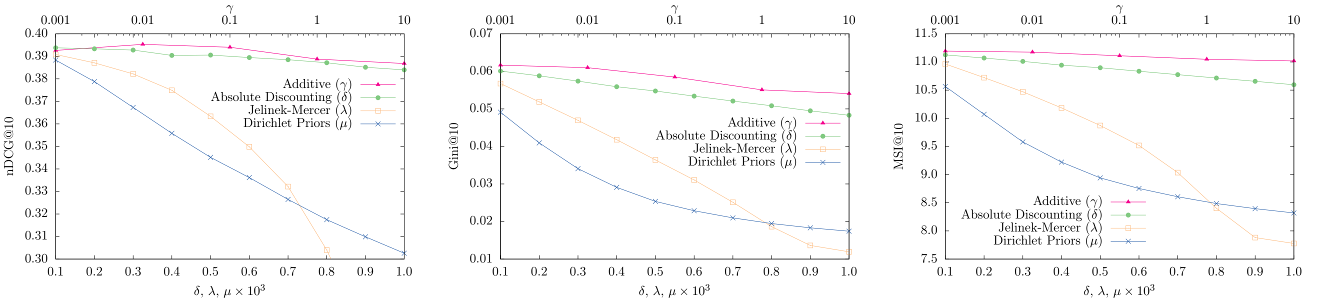

Figure 1: Values of nDCG@10 (left), Gini@10 (centre) and MSI@10 (right) for RM2 using Additive, Absolute Discounting, Jelinek-Mercer and Dirichlet Priors methods on the Movielens 100k collection varying the smoothing parameter. Neighbourhoods are computed taking the 50 closest users according to cosine similarity.

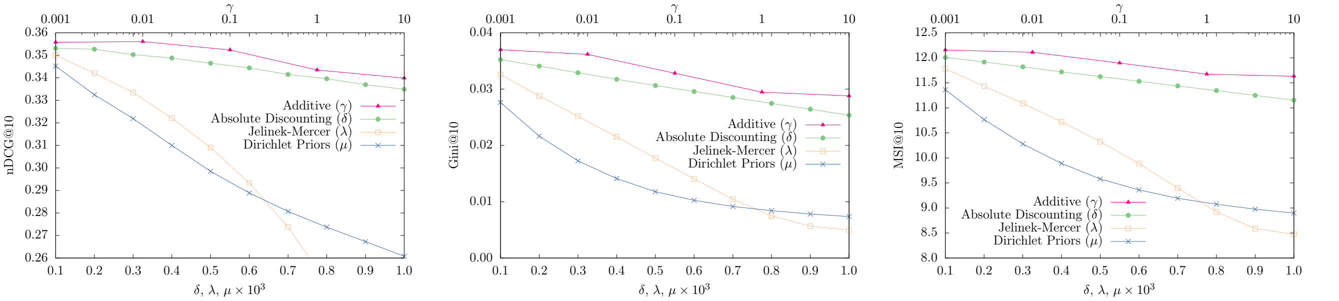

Figure 2: Values of nDCG@10 (left), Gini@10 (centre) and MSI@10 (right) for RM2 using Additive, Absolute Discounting, Jelinek-Mercer and Dirichlet Priors methods on the Movielens 1M collection varying the smoothing parameter. Neighbourhoods are computed taking the 100 closest users according to cosine similarity.

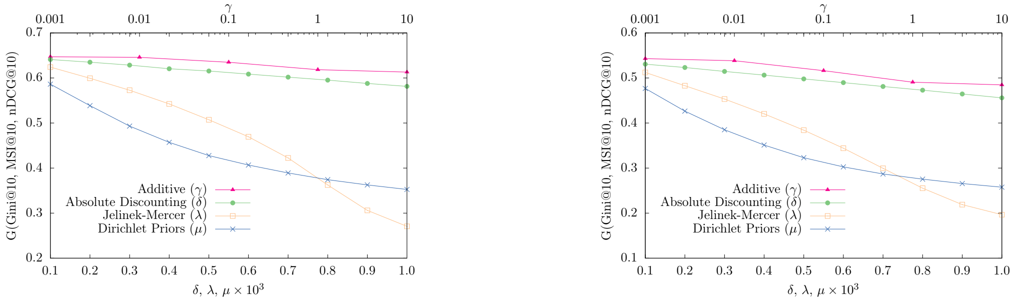

Figure 3: G-measure of nDCG@10 , Gini@10 and MSI@10 for RM2 using Additive, Absolute Discounting, Jelinek-Mercer and Dirichlet Priors methods on the Movielens 100k (left) and 1M (right) collections varying the smoothing parameter. Neighbourhoods are computed taking the 50 and 100 closest users according to cosine similarity, respectively for each dataset.

enhancing many other aspects such as novelty and diversity. In the present paper, we study the connection between the IDF effect from Information Retrieval to the concept of novelty in recommendation. We analysed axiomatically how different smoothing methods affect the IDF effect on RM2. We found that collection-based methods penalise this effect while Additive smoothing neither promotes nor demotes this property. Our experiments confirmed that Additive smoothing provides better result than collection-based smoothing methods improving accuracy, diversity and novelty figures.

As future work, it would be interesting to develop new smoothing methods that do actively promote the IDF effect in Relevance Models. Additionally, in this work we considered only uniform priors; however, it has been shown that different priors can improve the quality of recommendations [28]. We think that studying these priors axiomatically may be worthwhile.

7. ACKNOWLEDGMENTS

This work was supported by the grant TIN2015-64282-R of *Ministerio de Economía y Competitividad* of the Goverment of Spain and FEDER funds. The first author also wants to acknowledge the support of *Ministerio de Educación, Cultura y Deporte* of the Government of Spain under the grant FPU014/01724.

8. REFERENCES

[1] R. Baeza-Yates and B. Ribeiro-Neto. *Modern Information Retrieval: The Concepts and Technology Behind Search*. Addison Wesley, 2011.

[2] N. J. Belkin and W. B. Croft. Information Filtering and Information Retrieval: Two Sides of the Same Coin? *Commun. ACM*, 35(12):29–38, 1992.

[3] A. Bellogín, P. Castells, and I. Cantador. Precision-Oriented Evaluation of Recommender Systems. In *RecSys '11*, pages 333–336, 2011.

[4] A. Bellogín, J. Wang, and P. Castells. Bridging Memory-Based Collaborative Filtering and Text Retrieval. *Inf. Retr.*, 16(6):697–724, 2013.

[5] J. Bennett and S. Lanning. The Netflix Prize. In *Proceedings of KDD Cup and Workshop 2007*, pages 3–6, 2007.

[6] C. Carpineto, R. de Mori, G. Romano, and B. Bigi. An Information-Theoretic Approach to Automatic Query Expansion. *ACM Trans. Inf. Syst.*, 19(1):1–27, 2001.

[7] P. Castells, N. J. Hurley, and S. Vargas. Novelty and Diversity in Recommender Systems. In F. Ricci, L. Rokach, and B. Shapira, editors, *Recommender Systems Handbook*, pages 881–918. Springer, 2nd edition, 2015.

[8] P. Cremonesi, Y. Koren, and R. Turrin. Performance of Recommender Algorithms on Top-N Recommendation Tasks. In *RecSys '10*, pages 39–46, 2010.

[9] M. de Gemmis, P. Lops, C. Musto, F. Narducci, and G. Semeraro. Semantics-Aware Content-Based Recommender Systems. In F. Ricci, L. Rokach, and B. Shapira, editors, *Recommender Systems Handbook*, pages 119–159. Springer, 2nd edition, 2015.

[10] D. Fleder and K. Hosanagar. Blockbuster Culture's Next Rise or Fall: The Impact of Recommender Systems on Sales Diversity. *Manage. Sci.*, 55(5):697–712, 2009.

[11] A. Gunawardana and G. Shani. Evaluating Recommender Systems. In F. Ricci, L. Rokach, and B. Shapira, editors, *Recommender Systems Handbook*, pages 265–308. Springer, 2nd edition, 2015.

[12] U. Hanani, B. Shapira, and P. Shoval. Information Filtering: Overview of Issues, Research and Systems. *User Model. User-Adapt. Interact.*, 11(3):203–259, 2001.

[13] H. Hazimeh and C. Zhai. Axiomatic Analysis of Smoothing Methods in Language Models for Pseudo-Relevance Feedback. In *ICTIR '15*, pages 141–150, 2015.

[14] J. L. Herlocker, J. A. Konstan, L. G. Terveen, and J. T. Riedl. Evaluating Collaborative Filtering Recommender Systems. *ACM Trans. Inf. Syst.*, 22(1):5–53, 2004.

[15] N. Hurley and M. Zhang. Novelty and Diversity in Top-N Recommendation – Analysis and Evaluation. *ACM Trans. Internet. Technol.*, 10(4):1–30, 2011.

[16] F. Jelinek and R. L. Mercer. Interpolated Estimation of Markov Source Parameters from Sparse Data. In *Proceedings of the Workshop on Pattern Recognition in Practice*, pages 381–397, 1980.

[17] V. Lavrenko and W. B. Croft. Relevance-Based Language Models. In *SIGIR '01*, pages 120–127, 2001.

[18] D. J. C. MacKay and L. C. B. Peto. A hierarchical Dirichlet language model. *Nat. Lang. Eng.*, 1(03):289–308, 1995.

[19] M. R. McLaughlin and J. L. Herlocker. A Collaborative Filtering Algorithm and Evaluation Metric that Accurately Model the User Experience. In *SIGIR '04*, pages 329–336, 2004.

[20] H. Ney, U. Essen, and R. Kneser. On Structuring Probabilistic Dependences in Stochastic Language Modelling. *Comput. Speech Lang.*, 8(1):1–38, 1994.

[21] X. Ning, C. Desrosiers, and G. Karypis. A Comprehensive Survey of Neighborhood-Based Recommendation Methods. In F. Ricci, L. Rokach, and B. Shapira, editors, *Recommender Systems Handbook*, pages 37–76. Springer, 2nd edition, 2015.

[22] J. Parapar and Á. Barreiro. Promoting Divergent Terms in the Estimation of Relevance Models. In *ICTIR '11*, pages 77–88. Springer, 2011.

[23] J. Parapar, A. Bellogín, P. Castells, and Á. Barreiro. Relevance-Based Language Modelling for Recommender Systems. *Inf. Process. Manage.*, 49(4):966–980, 2013.

[24] F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor. *Recommender Systems Handbook*. Springer, 2nd edition, 2015.

[25] S. Robertson. Understanding inverse document frequency: on theoretical arguments for IDF. *J. Doc.*, 60(5):503–520, 2004.

[26] K. Spärck Jones. A Statistical Interpretation of Term Specificity and its Retrieval. *J. Doc.*, 28(1):11–21, 1972.

[27] D. Valcarce. Exploring Statistical Language Models for Recommender Systems. In *RecSys '15*, pages 375–378, 2015.

[28] D. Valcarce, J. Parapar, and Á. Barreiro. A Study of Priors for Relevance-Based Language Modelling of Recommender Systems. In *RecSys '15*, pages 237–240, 2015.

[29] D. Valcarce, J. Parapar, and Á. Barreiro. A Study of Smoothing Methods for Relevance-Based Language Modelling of Recommender Systems. In *ECIR '15*, pages 346–351. Springer, 2015.

[30] D. Valcarce, J. Parapar, and Á. Barreiro. Language Models for Collaborative Filtering Neighbourhoods. In *ECIR '16*, pages 614–625. Springer, 2016.

[31] J. Wang, A. P. de Vries, and M. J. T. Reinders. A User-Item Relevance Model for Log-based Collaborative Filtering. In *ECIR '06*, volume 3936, pages 37–48. Springer, 2006.

[32] Y. Wang, L. Wang, Y. Li, D. He, W. Chen, and T.-Y. Liu. A Theoretical Analysis of NDCG Ranking Measures. In *COLT '13*, pages 1–30. JMLR.org, 2013.

[33] C. Zhai and J. Lafferty. A Study of Smoothing Methods for Language Models Applied to Information Retrieval. *ACM Trans. Inf. Syst.*, 22(2):179–214, 2004.

[34] T. Zhou, Z. Kuscsik, J.-G. Liu, M. Medo, J. R. Wakeling, and Y.-C. Zhang. Solving the Apparent Diversity-Accuracy Dilemma of Recommender Systems. *PNAS*, 107(10):4511–5, 2010.