Comparative evaluation of different recommendation strategies in a commercial context

Markus Zanker, University Klagenfurt

Markus Jessenitschnig, eTourism Competence Center Austria

Dietmar Jannach, University Klagenfurt

Sergiu Gordea, University Klagenfurt

Abstract. Recommender Systems (RS) have a long tradition in reducing search costs of users by proposing items based on users` preferences and/or aggregated information about other users. In electronic commerce scenarios different types of user preferences – implicitly collected ratings as well as explicitly formulated requirements – are available. Therefore, we perform a comparative evaluation across different recommendation techniques such as knowledge-based sales advisory or collaborative filtering on a commercial dataset. By making this dataset in addition publicly available we want to foster research efforts on the specific requirements of commercial shopping platforms.

Keywords*: Recommender Systems, Personalization, Evaluation.*

Introduction

Seen from an industrial perspective recommender systems (RS) constitute the base technology for providing interactivity and personalization in electronic business-to-consumer (B2C) marketplaces. However, most reported recommender applications are not validated on commercial datasets or in real-world sales situations. Therefore, we present a study on the use of a conversational recommender within a Web shop for fine Cuban cigars and evaluate the performance of different recommendation techniques against each other within the context of this electronic commerce environment.

Common recommendation methods: Burke [1] distinguishes between five different recommendation techniques: collaborative, content-based, utility-based, demographic, and knowledge based. **Collaborative filtering** is the most prominent one amongst them; it exploits clusters of users that showed similar tastes in the past and proposes them those products that their statistically nearest neighbors also liked [4][5][8]. **Content-based** and **knowledge-based** methods both rely on product descriptions: Pure content-based approaches learn a preference model for a user from the product characteristics of those items that she liked in the past, while the latter exploit deep domain knowledge in the form of mappings between abstract user preferences and required product characteristics [1][9]. **Utility-based** RS are comparable to knowledge-based ones in the sense of producing recommendations based on user preferences and product knowledge. However, utility-based systems do not possess explicit mapping rules but require definitions of utility values that specific product characteristics contribute to the fulfillment of given user requirements. Finally, **demographic information filtering** approaches are based on the assumption that users with a similar social, cultural, and regional background share similar tastes and needs.

While in most systems solely a single variant of the above mentioned techniques is implemented, we argue that in electronic commerce scenarios several different recommendation approaches should complement each other and be presented to the user according to her situational context. For instance, different preference elicitation dialogues that lead to knowledge-based recommendations within a product domain may be offered: Personalized cross-selling offers can be presented to the online shopper using for instance information filtering techniques or additional functions like similarity-based browsing or critiquing. Thus enabling users to quicker navigate to those items they are looking for.

In this paper we therefore contribute a comparative evaluation of eight different algorithms on real-world commercial data and emphasize on the observed commonalities and differences.

Background

Our industrial experience stems from founding a company for interactive selling applications in 2002. Since then we have deployed more than 35 recommender systems in diverse domains such as financial services, consumer electronics, investor counseling and quality & taste products. One of our first deployed applications was a virtual sales assistant for Cuban cigars in 2003 and by the courtesy of the shop owner we were allowed to perform evaluation experiments on the dataset[1]. According to the classification scheme for hybrid algorithms in [2] the virtual sales agent *Mortimer* is a hybrid RS that is cascading knowledge- and utility-based methods. The system was developed on the basis of the Advisor Suite framework [6] and converses with its users to find out about their requirements and preferences. The first question in this application is: *Whom is it for?* If a cigar is bought as a gift, prestige, exclusiveness, and originality become important. In case someone buys it for its own, an enjoyable taste and a smoking experience without regret are the determinants when selecting among a range of more than 140 models. Therefore, depending on the answers the system determines the next appropriate question or comes up with a ranked list of at most five proposals. The number of possible recommendations is in a first step reduced by applying a set of matching filtering rules. They have been derived from the domain knowledge of the shop owner as part of the knowledge engineering effort in the setup phase, e.g. *if the user has not yet smoked a cigar, only models with a smoking duration of sixty minutes and less apply*. Rules are either soft or hard. Therefore, in case no recommendation would be left, the recommender system can relax some rules according to an optimization function. In a second step a utility-based recommender produces a ranking and outputs those five items with the highest computed user utility. The utility function is automatically derived from user's answers: If, for instance, the user's purpose is to make a present, prestigious makes like *Cohiba* or *Montecristo*

[1] The dataset can be downloaded at http://isl.ifit.uni-klu.ac.at/.

are associated with higher utility values than less prominent ones like *Hoyo de Monterrey* or *Rafael Gonzales*. A more detailed description of the system can found in [6].

Within the evaluation period, one eighth of all online shoppers that showed interest in different products also interacted with the conversational recommender system. Consequently, recommendation strategies that follow a one-shot interaction paradigm (e.g. collaborative filtering) do potentially address a larger share of users. Sarwar et al. [3] compared variants of collaborative filtering algorithms with mined association rules on commercial data. However, knowledge-based RS use heuristics that are elicited from a domain expert before the going live of the system. As a consequence, knowledge-based RS do not suffer from cold-start problems for new users or new products. As a consequence they require a considerable amount of knowledge acquisition efforts for their setup and some additional maintenance activities during their operational lifetime.

Therefore, our evaluation follows the goal to compare accuracy as well as user and catalog coverage of these different recommendation algorithms in order to learn about their appropriateness for different commercial application scenarios.

Experimental setup

Based on actual data from the period of November 2005 to November 2006 we instrumented a comparative offline analysis on different recommendation algorithm variants. We collected two types of (implicit) ratings from binary transaction data: customers` buying history and their click-stream data. Furthermore, we evaluated the interaction log of the conversational sales advisor and used product descriptions for computing item similarities.

As only a fraction of all users of the shop was interacting with the sales advisor we compiled four different datasets. Dataset *Buys-All* contains 1005 buying transactions of 508 different users. The binary ratings thus signify that a user bought a specific item at least once during the evaluation period. Dataset *Buys-Advisor* holds only this subset of users from *Buys-All* that also interacted with the sales advisor. Dataset *Views-All* consists of 18434 binary ratings from 1260 different users. Here, a rating denotes that a user showed interest in a specific item. Interest is derived by the fact that the user accessed the detailed description page of an item during the evaluation period. Access to detailed item descriptions requires at least two clicks using the catalog navigation. Again, *Views-Advisor* is the subset of those users who were also interacting with the virtual sales agent. As can be easily observed, the average number of ratings per user in the *Buys-All* dataset is much lower than in the one compiled from clickstream data. However, it reflects a real-world situation recommender systems have to face in commercial domains. In case of electronic consumer goods we can expect an even lower average number of ratings per user as these goods are typically less repetitiously bought.

Methodology

In order to find out about characteristic properties of recommendation algorithms in a commercial context, we pose the following research questions:

- *How accurate are different recommendation algorithms in predicting user transactions in an online shopping environment?*
- *Which differences exist with respect to User Coverage and Product Coverage between algorithms, i.e. their capability to present recommendations to new users and to produce novel and serendipitous results?*

Our contribution lies in the instrumentation of a comparative analysis of the following eight recommendation algorithm variants. By answering the above questions we want to identify appropriate application scenarios for the different recommendation methods in commercial contexts.

- ‘Top n’ recommendation based on sales records (*Topn*)
- Knowledge-based and utility-based cascading hybrid without relaxation (*KBUt*)
- Knowledge-based and ‘Top n’ cascading hybrid without relaxation (*KBTopn*)
- Knowledge-based and utility-based cascading hybrid with relaxation (*KBUt-relax*)
- Knowledge-based and ‘Top n’ cascading hybrid with relaxation (*KBTopn-relax*)
- Collaborative filtering (*CF*)
- Content-based filtering (*CB*)
- Collaborative and Content-based filtering feature-augmentation hybrid (*CBCF*)

Topn is the baseline algorithm: It produces an impersonalized ranked list of recommendations according to the frequency of the item within the dataset, i.e. the top selling products.

KBUt-relax constitutes the knowledge-based RS actually deployed in the commercial environment. The knowledge-based RS require their users to engage in a question/answer style of dialogue to elicit their preferences. In a second step filtering rules restrict the set of recommendable products, if their condition part evaluates to true. Finally, all remaining recommendable products are sorted according to a ranking function and the *n* highest ranked items are recommended. The utility-based ranking function personalizes the weighting scheme depending on the user input [6]. The *Topn* variant ranks the items according to their popularity in the buying resp. viewing history. In addition we varied the relaxation of filtering rules. The knowledge-based algorithm variants without relaxation do not return a recommendation to their users if all applicable filtering rules produce an empty result set, while the variants with relaxation do release some of the filtering rules depending on their priority. The collaborative filtering algorithm is implemented with standard Pearson-correlation on a user-to-user rating matrix as presented in [3]. Note that only binary transaction data is available and therefore rating values do not need to be normalized with average ratings. Neighborhood formation is based on a center-based approach. For each user we choose the *l* nearest neighbors, while the parameter *l* was set to 30. The *top-N* recommendations for a user *u* are derived by computing the weighted sum of ratings r_{iv} for each product *i* and neighbor *v*, i.e. $weight_i = \sum_v r_{iv} \cdot pearson_{uv}$ and selecting those products with the highest weight that have not yet been rated by user *u*.

The content-based algorithm retrieves those items that are most similar to at least one of the items the user likes resp. has positively rated. We defined the similarity function as the weighted sum of the similarities for the product features *price*, *taste*, *effect* and *smoking duration*. As price and smoking duration are defined on a linear scale, we used a formula of relative distance. The *taste* and *effect* of a cigar are each circumscribed with sets of terms such as *spicy* or *harsh*. Therefore, the *Jaccard coefficient* was used, i.e. $\frac{|A \cap B|}{|A \cup B|}$, where A and B are the sets of terms that circumscribe for instance the *taste* of two different cigars. The collaborative and content-based feature augmentation hybrid is based on the content-boosted collaborative filtering algorithm described in [7]. It produces additional system ratings for items similar to an item the user actually rated. This way the sparsity of the rating matrix is reduced and larger user neighborhoods can be computed. However, the maximum size of the user neighborhood was also set to 30 like in the case of the collaborative filtering algorithm. When computing the *top-N* recommendations the algorithm considers only the actual ratings of the user and discards the system rates.

We randomly separate the set of ratings for a given user into a set of ratings used for training the algorithm and a remaining set for testing. We instrumented 25 experiment runs for each variant of the *Given n* and the *All but n* methods: *Given n* defines the training set with size *n*, while at least one additional rating must remain in the testing set. *All but n* sets the testing set to *n* and at least two additional ratings must be available for training. Knowledge-based methods and the baseline algorithm do not require ratings for training the algorithm, therefore they are evaluated on the *Given zero* method, i.e. the training set is empty and all ratings can be used for testing.

In each recommendation trial, at most the five resp. ten[2] highest ranked items are proposed to a user[3]. The cardinality of the intersection between the recommendation list and the testing set constitutes the set of hits (i.e. *hit_set*) for a given user. The Recall metric is computed as follows: $recall = \frac{|hit\_set|}{|testing\_set|}$. For each user, Recall is computed separately and the overall Recall is computed as the average of the Recall values for all users that received a recommendation from the algorithm. Therefore, we also indicate User Coverage that is defined as the share of users from the overall dataset that were recommended at least one item during the trial. Catalog coverage measures the novelty and serendipity of the produced recommendations. It is therefore expressed as the percentage of items in the catalog that are ever recommended to users [5].

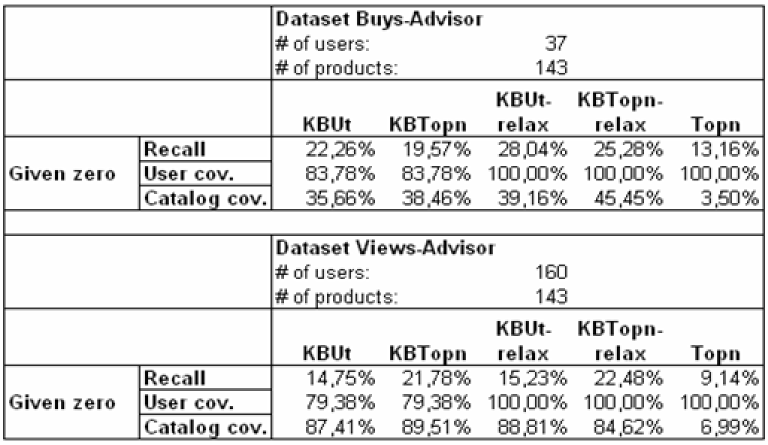
Results

In our experiments we are comparing two fundamentally different types of algorithms, each of them having specific strengths and weaknesses. On the one hand knowledge-based methods require a considerable amount of knowledge engineering from the very beginning, but do not suffer from cold-start problems; on the other hand learning-based algorithms require less effort at setup time but require a relatively high average number of ratings per user. These commonly known properties of these approaches became quite evident through our experimentation. Not surprisingly, knowledge-based algorithm variants clearly outperform learning-based methods in

[2] For the *Buys-All* and *Buys-Advisor* datasets we limited recommendation lists to five and for the larger *Views-All* and *Views-Advisor* datasets to ten.

[3] Depending on user neighbourhood, product characteristics and/or filtering rules algorithms might also recommend fewer than five items to a given user.

terms of User Coverage when applied on dataset *Buys-Advisor*, as they do not suffer from cold-start problems (see tables 1 and 2 for details).

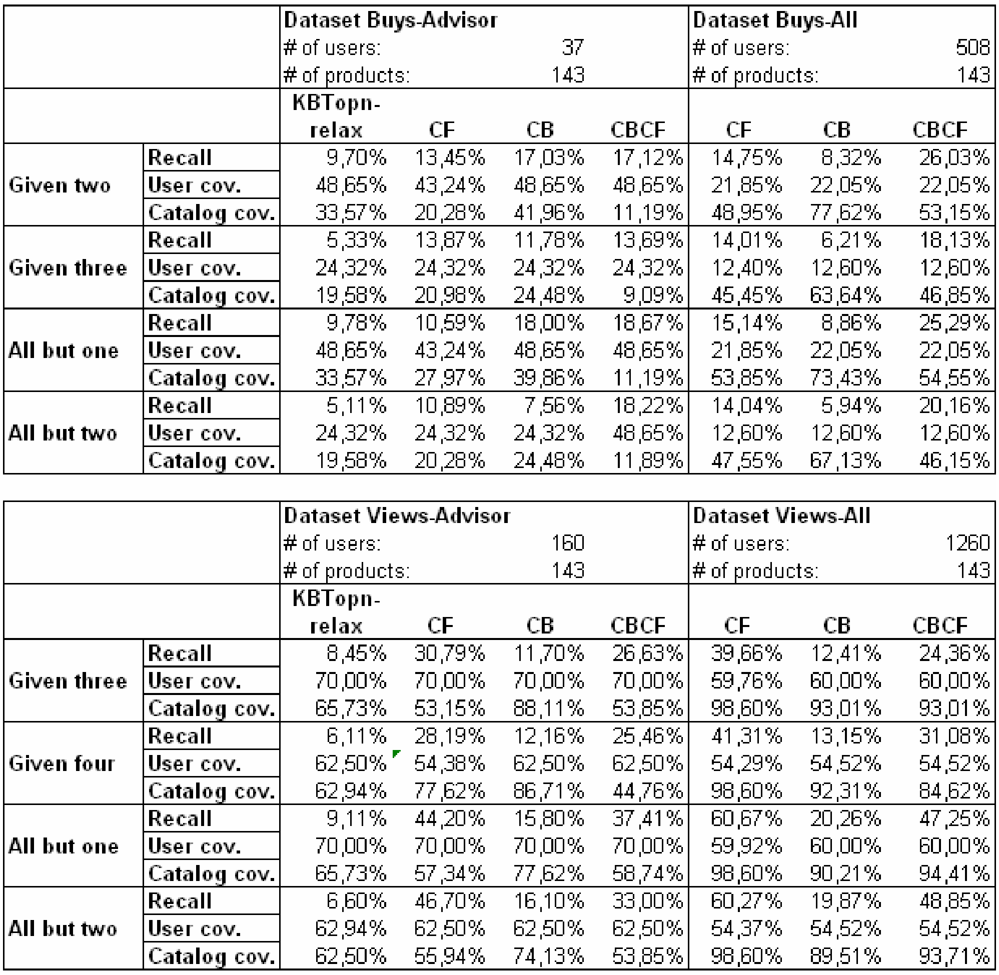


		Dataset Buys-Advisor				
		# of users:		37		
		# of products:		143		
		KBUt	KBTopn	KBUt-relax	KBTopn-relax	Topn
Given zero	Recall	22,26%	19,57%	28,04%	25,28%	13,16%
	User cov.	83,78%	83,78%	100,00%	100,00%	100,00%
	Catalog cov.	35,66%	38,46%	39,16%	45,45%	3,50%
		Dataset Views-Advisor				
		# of users:		160		
		# of products:		143		
		KBUt	KBTopn	KBUt-relax	KBTopn-relax	Topn
Given zero	Recall	14,75%	21,78%	15,23%	22,48%	9,14%
	User cov.	79,38%	79,38%	100,00%	100,00%	100,00%
	Catalog cov.	87,41%	89,51%	88,81%	84,62%	6,99%

Table 1: Results on knowledge-based hybrids

In addition, knowledge-based methods can keep up with collaborative filtering with respect to Catalog Coverage. This clearly contradicts the common argument that solely collaborative filtering systems may achieve good results in terms of novelty and serendipity of recommendations. If applied to a specific domain where domain heuristics are available for deriving recommendations, also knowledge-based RS may achieve a considerable good Catalog Coverage (close to 90% on the *Views-Advisor* dataset). Furthermore, the difference in Catalog Coverage between evaluation results on *Buys-Advisor* and *Views-Advisor* datasets has to be attributed to the different lengths of recommendation lists.

When taking a closer look on the different variants of knowledge-based hybrids (compare table 1), we can state that they all outperformed the impersonal *Topn* recommender, whose Catalog Coverage remains below 10% as it was constantly proposing the list of the five respectively ten topseller items.



		Dataset Buys-Advisor # of users: 37 # of products: 143				Dataset Buys-All # of users: 508 # of products: 143		
		KBTopn-relax	CF	CB	CBCF	CF	CB	CBCF
Given two	Recall	9,70%	13,45%	17,03%	17,12%	14,75%	8,32%	26,03%
	User cov.	48,65%	43,24%	48,65%	48,65%	21,85%	22,05%	22,05%
	Catalog cov.	33,57%	20,28%	41,96%	11,19%	48,95%	77,62%	53,15%
Given three	Recall	5,33%	13,87%	11,78%	13,69%	14,01%	6,21%	18,13%
	User cov.	24,32%	24,32%	24,32%	24,32%	12,40%	12,60%	12,60%
	Catalog cov.	19,58%	20,98%	24,48%	9,09%	45,45%	63,64%	46,85%
All but one	Recall	9,78%	10,59%	18,00%	18,67%	15,14%	8,86%	25,29%
	User cov.	48,65%	43,24%	48,65%	48,65%	21,85%	22,05%	22,05%
	Catalog cov.	33,57%	27,97%	39,86%	11,19%	53,85%	73,43%	54,55%
All but two	Recall	5,11%	10,89%	7,56%	18,22%	14,04%	5,94%	20,16%
	User cov.	24,32%	24,32%	24,32%	48,65%	12,60%	12,60%	12,60%
	Catalog cov.	19,58%	20,28%	24,48%	11,89%	47,55%	67,13%	46,15%

		Dataset Views-Advisor # of users: 160 # of products: 143				Dataset Views-All # of users: 1260 # of products: 143		
		KBTopn-relax	CF	CB	CBCF	CF	CB	CBCF
Given three	Recall	8,45%	30,79%	11,70%	26,63%	39,66%	12,41%	24,36%
	User cov.	70,00%	70,00%	70,00%	70,00%	59,76%	60,00%	60,00%
	Catalog cov.	65,73%	53,15%	88,11%	53,85%	98,60%	93,01%	93,01%
Given four	Recall	6,11%	28,19%	12,16%	25,46%	41,31%	13,15%	31,08%
	User cov.	62,50%	54,38%	62,50%	62,50%	54,29%	54,52%	54,52%
	Catalog cov.	62,94%	77,62%	86,71%	44,76%	98,60%	92,31%	84,62%
All but one	Recall	9,11%	44,20%	15,80%	37,41%	60,67%	20,26%	47,25%
	User cov.	70,00%	70,00%	70,00%	70,00%	59,92%	60,00%	60,00%
	Catalog cov.	65,73%	57,34%	77,62%	58,74%	98,60%	90,21%	94,41%
All but two	Recall	6,60%	46,70%	16,10%	33,00%	60,27%	19,87%	48,85%
	User cov.	62,94%	62,50%	62,50%	62,50%	54,37%	54,52%	54,52%
	Catalog cov.	62,50%	55,94%	74,13%	53,85%	98,60%	89,51%	93,71%

Table 2: Comparative results on learning- and knowledge-based methods

When applying the same experiment design learning-based recommendation strategies reached significantly better results with respect to Recall than the knowledge-based *KBTopn-relax* variant on all datasets (compare Table 2). With higher number of average ratings per user learning-based methods massively improve their lead (compare results on *Views-Advisor* vs. *Views-All*). But admittedly, we have to note that the experiment designs of Table 2 penalize the knowledge-based algorithms. Users typically interacted several times with the recommender systems and their requirements varied during the evaluation period. However, we used only those user requirements entered during her/his final interaction. Context-aware experiment designs that consider the last validly entered user requirements at the point of time of a transaction would be required here.

The content-boosted collaborative filtering hybrid [7] was performing quite poor as already noticed in [10]. Interestingly, for smaller and sparse datasets the content-boost adds some improvements, but for larger dataset it deteriorates results.

In general, we can state that the reported results on accuracy - Recall roughly in the range of 10% to 50% and Precision below 15% - are significantly worse than results typically reported in evaluation reports of recommender systems. This leads us to an additional conclusion from this evaluation exercise, namely that research on recommendation algorithm will still have a long way to go for optimizing methods and algorithms that will then show their strengths on sparse commercial datasets.

Conclusions

Most of today's research efforts concerned with the evaluation of a recommender system's performance (measured e.g., by means of accuracy or coverage) are based on publicly available (movie) recommendation datasets. Thus, the possibilities of gaining new insights by evaluating these datasets are limited to some extent, as these data stem from a rather specific type of recommender application. In this paper, we have therefore particularly focused on an evaluation in a real-world commercial context. Based on a real-world dataset in the domain of fine cigars, a comparative evaluation of the two fundamentally different algorithm types (knowledge- vs. learning-based) has been conducted. By making our datasets publicly available, we aim at stimulating and fostering further research especially within the context of commercial shopping platforms.

References

[1] Balabanović, M. and Shoham, Y. Fab: content-based, collaborative recommendation. *Communications of the ACM*, 40, 3 (1997), pp. 66 – 72.

[2] Burke, R. Hybrid recommender systems: survey and experiments. *User-Modeling and User-Adapted Interaction*, 12, (2002), pp. 331-370.

[3] Sarwar, B.; Karypis, G.: Konstan, J. and Riedl, J. Analysis of Recommendation Algorithms for E-Commerce. *2nd ACM Conference on Electronic Commerce (EC-00)*, 2000, pp. 158-167.

[4] Cotter, P. and Smyth, B. PTV: Intelligent Personalised TV Guides. *12th International Conference on Innovative Applications of Artificial Intelligence,* (2000), pp. 957–964.

[5] Herlocker, J.; Konstan, J.; Terveen, L.; and Riedl, J. Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems*, 22, 1 (2004), pp. 5–53.

[6] Jannach, D. Advisor Suite–A knowledge-based sales advisory system. *16th European Conference on Artificial Intelligence - Prestigious Applications of AI*, (2004), pp. 390–402.

[7] Melville, P.; Mooney, R. and Nagarajan, R. Content-Boosted Collaborative Filtering for Improved Recommendations. *Proceedings of the Eighteenth National Conference on Artificial Intelligence*(AAAI-2002), Edmonton, Canada, pp. 187-192.

[8] Resnick, P.; Iacovou, N.; Suchak, M.; Bergstrom, P. and Riedl, J. GroupLens: An Open Architecture for Collaborative Filtering of Netnews. *ACM Conference on Computer Supported Cooperative Work*, Chapel Hill, NC: ACM, 1994, pp. 175–186.

[9] Ricci, F.; Venturini, A.; Cavada, D.; Mirzadeh, N.; Blaas, D. and Nones, M. Product Recommendation with Interactive Query Management and Twofold Similarity. 5*th International Conference on Case-Based Reasoning.* Trondheim, Norway, 2003, pp. 479-493.

[10] Tso, K.; Schmidt-Thieme, L. Attribute-aware Collaborative Filtering. *29th Annual Conference of the German Classification Society.* Magdeburg, Springer, 2005.