Multi-Modal Adversarial Autoencoders for Recommendations of Citations and Subject Labels

Lukas Galke
Kiel University
Germany
lga@informatik.uni-kiel.de

Florian Mai
Kiel University
Germany
stu96542@informatik.uni-kiel.de

Iacopo Vagliano
ZBW – Leibniz Information Centre for Economics
Kiel, Germany
I.Vagliano@zbw.eu

Ansgar Scherp
Kiel University
Germany
asc@informatik.uni-kiel.de

ABSTRACT

We present multi-modal adversarial autoencoders for recommendation and evaluate them on two different tasks: citation recommendation and subject label recommendation. We analyze the effects of adversarial regularization, sparsity, and different input modalities. By conducting 408 experiments, we show that adversarial regularization consistently improves the performance of autoencoders for recommendation. We demonstrate, however, that the two tasks differ in the semantics of item co-occurrence in the sense that item co-occurrence resembles relatedness in case of citations, yet implies diversity in case of subject labels. Our results reveal that supplying the partial item set as input is only helpful, when item co-occurrence resembles relatedness. When facing a new recommendation task it is therefore crucial to consider the semantics of item co-occurrence for the choice of an appropriate model.

CCS CONCEPTS

• **Information systems → Recommender systems**; • **Computing methodologies → Neural networks**; *Learning from implicit feedback*;

KEYWORDS

recommender systems; neural networks; adversarial autoencoders; multi-modal; sparsity

ACM Reference Format:
Lukas Galke, Florian Mai, Iacopo Vagliano, and Ansgar Scherp. 2018. Multi-Modal Adversarial Autoencoders for Recommendations of Citations and Subject Labels. In *UMAP '18: 26th Conference on User Modeling, Adaptation and Personalization, July 8–11, 2018, Singapore, Singapore.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3209219.3209236

UMAP '18, July 8–11, 2018, Singapore, Singapore

ACM ISBN 978-1-4503-5589-6/18/07…$15.00
https://doi.org/10.1145/3209219.3209236

1 INTRODUCTION

Recent advances in autoencoders on images have shown that adversarial regularization can improve the performance of autoencoders [24]. The so-called adversarial autoencoders [24] are not only trained to reconstruct the input, but also to match the code with a selected prior distribution. We hypothesize that the thereby imposed smoothness on the code aids autoencoders in reconstructing highly sparse item vectors for recommendation. The rationale is that smoothness is one of the criteria for good representations that disentangle the explanatory factors of variation [4]. In this paper, we analyze whether adversarial autoencoders can be applied to highly sparse recommendation tasks. We evaluate the effect of adversarial regularization with respect to the degree of sparsity and different input modalities on two exemplary tasks: citation and subject label recommendation.

Citation Recommendation More and more publishers decide to contribute to the Initiative for Open Citations[1], which aims to make citation metadata publicly available. This motivates us to consider the following scenario as a recommendation task. When writing a new paper, it is essential that the authors reference other publications which are key in the respective field of study or relevant to the paper being written. Failing to do so can be rated negatively by reviewers in a peer-reviewing process. However, due to increasing volume of scientific literature, even some critical paper are sometimes overlooked. Hence, in this paper we study the problem of recommending publications to consider as citation candidates, given that the authors have already selected some other references and assuming that the paper is close to completion, i. e., information such as the title (or a tentative title) of the paper is available.

Subject Indexing Apart from citation data, also subject labels, or tags, are publicly available for numerous domains, such as MeSH[2] for medicine or EconBiz[3] for economics. Subject indexing is a common task in scientific libraries to make documents accessible for search. New documents are annotated with a set of subjects by professional subject indexers. Fully-automated multi-label classification approaches to subject

[1] https://i4oc.org
[2] https://www.nlm.nih.gov/mesh/
[3] https://www.econbiz.de/

indexing are promising [29], even when merely the metadata of the publications is used [9]. Professional subject indexers, however, typically use the result of these approaches only as recommendations, so that the human-level quality can still be guaranteed. This circumstance motivates us to build a subject label recommender system that explicitly takes the partial list of already assigned subjects into account.

To unify these two scenarios, we take either the citations or the assigned subjects as implicit feedback for a considered recommendation task. In the former case, citations are known to resemble credit assignment [41], whereas in the latter case the subject labels are selected by respective professionals such that their relevance to the paper is guaranteed by human supervision.

Traditionally, the recommendation problem is modeled as the prediction of missing ratings in a $\boldsymbol{U} \times \boldsymbol{I}$ matrix with set of users $\boldsymbol{U}$ and set of items $\boldsymbol{I}$ (matrix completion). In our case, following McNee et al. [25], we view research papers themselves as users over their authors or the responsible subject indexers. The rationale is that one author may be involved in multiple papers of different domains but that all authors for a given paper should receive the same recommendations. Analogously, a given paper should receive the same recommendations for candidate subjects, independently of the current subject indexer in charge of annotating it.

We have transferred the approach of Makhzani et al. [24], which was applied to images, and extended it to our problem of a general recommendation task. By developing a novel interpretation of the adversarial autoencoder, we show how it can be applied to recommendation tasks and how multiple input modalities can be incorporated. We make use of this capability in our experiments by considering besides the ratings also additional metadata, namely the documents' title, as content-based features. We performed 408 experiments for our two recommendation tasks to study how adversarial autoencoders perform while exploiting titles along with the partial list of citations or the already assigned subjects, respectively. For a close investigation of the adversarial autoencoders' performance, we not only consider the adversarial autoencoder as a whole but also individually assess its components.

We further evaluate to which degree these models are robust to sparsity in the dataset. When conducting citation or research paper recommendation, it is not desirable that only already frequently cited papers get recommended and less frequently cited papers are ignored. Common pruning strategies comprise removing rarely cited documents and documents that cite too few other works [2]. This pruning step affects the number of considered items, and thus, the degree of sparsity. To gain a better understanding of how the pruning threshold affects the models' performance, we conduct experiments, in which the pruning threshold is a controlled variable.

Our results show that the partial list of items is more important for the citation recommendation task than it is for the subject labeling task. This is interesting because an inspection of the semantics of item co-occurrence may help researchers or practitioners to tackle new recommendation tasks, specifically to decide whether to supply the partial list of items as input. For citation recommendation, item co-occurrence implied relatedness, i. e., it is of high relevance which other works have been cited so far. For subject labels, in contrast, co-occurrence implies diversity: similar subjects are rarely used together for annotation of a single document. Thus, the title is more relevant than the already assigned subjects. All of the evaluated methods appeared similarly sensitive to data sparsity despite the differences in the number of parameters.

Due to the use of the titles, the adversarial autoencoders yields competitive performance to the baselines. On the subject label recommendation task, they outperform the baselines. A closer look at the individual components of the adversarial autoencoder revealed that the sole MLP decoder achieved better performance than the whole model on the subject labelling task, while its performance fell behind the whole model on the citation recommendation task.

In summary, our contributions are the following:

- We present an adaption of adversarial autoencoders as a novel approach for multi-modal recommendation tasks on scientific documents.
- We analyze this new method along with all of its components on citation and subject label recommendation tasks while varying the input modalities. We gain valuable insights on the interactions between input modalities and the task: when item co-occurrence resembles relatedness, multi-modal variants are preferable, otherwise solely content-based variants may be more suitable.
- We evaluate the autoencoder models in realistic scenarios, as we split the datasets on the time axis and consider different thresholds for pruning by minimum item occurrence. This is especially important for the citations task because only already existing papers can be cited and it is desirable that also less cited papers are recommended.

The remainder of this paper is organized as follows. In Section 2, we review previous work on citation and tag recommendation as well as recommendation approaches from the deep learning field. After formally stating the problem in Section 3, we introduce the employed models in Section 4, describe the citation and subject recommendation experiments in Section 5. We discuss the results in Section 6, before we conclude.

2 RELATED WORK

Research paper and subject label recommendation. An extensive survey [2] shows that research paper recommendation is a well-known topic. In this context, BibTip [11] and bX [5] are well-known recommender systems, which operate on the basis of citations harvested by CiteSeer [12]. Docear is a more recent research paper recommender system, which takes user profiles into account [3]. For citation recommendation specifically, Huang et al. distinguish between recommendations based on a partial list of references and recommendations based on the content of a manuscript [17]. While the former is suited for finding matching citations for a given statement during writing, the latter strives to identify missing citations on the broader, document level. Citation recommendation recently focuses on context-sensitive applications, in which concrete sentences are mapped to, preferably relevant, citations [2, 8, 17]. Instead, we revisit the reference list completion problem and we do not take into account the context of the citation, as the full text of a papers is rarely available in large-scale metadata sources. In 1973, Small started the field of co-citation analysis [37]. Co-citation analysis assumes that two papers are more related to each other,

the more they are co-cited. Following that idea, Caragea et al. relied on singular value decomposition as a more efficient and extendable approach [6]. We recognize the need for new methods that are not only based on item co-occurrence but also take supplementary metadata into account for these partial list completion problems.

Subject label recommendation is similar to tag recommendation, as in both cases the goal is to suggest a descriptive label for some content. Sen et al. propose algorithms that predict users' preferences for items based on their inferred preferences for tags [35]. Montañés et al. exploit probabilistic regression for collaborative tag recommendation [27], while Krestel et al. relied on Latent Dirichlet allocation [23]. Similarly, Sigurbjörnsson and van Zwol propose a tag recommender for Flickr to support the user in the photo annotation task [36], whereas Posch et al. predict hashtag categories on Twitter [32]. Dellschaft and Staab measure the influence of tag recommender systems on the indexing quality in collaborative tagging systems [7]. These works, however, focus on tags for social media, while we consider subject labels from a standardized thesaurus for scientific documents.

Recommendation and Link Prediction based on Deep Learning. Multiple recommender systems based on deep learning have been proposed. Wang et al. used deep learning for collaborative filtering [40]. Another recent collaborative-filtering approach explicitly takes side information into account for autoencoders [1]. We include a similar model in our comparison, as it is one component of the adversarial autoencoder. Additional techniques employ recurrent neural networks to provide session-based recommendations [33] or combine knowledge graphs with deep learning [30, 34]. To the best of our knowledge, only two approaches makes use of deep learning techniques for citation recommendation. However, both of them focus on context-sensitive scenarios [8, 18].

Citation networks are also considered in many studies on link prediction. By making use of the network structure, dedicated architectures learn representations of its nodes. One of the most prominent approaches is DeepWalk [31], together with its extension Node2vec [16]. These methods perform a random walk over the graph and feed the generated sequence into skip-gram negative sampling methods [26]. Kipf and Welling recently proposed Graph Auto-Encoders [21] and Graph Convolutional Networks [20]. However, all of these graph-based approaches assume that all nodes (research papers) are known during training. Hence, they are unable to deal with unknown nodes (new, unseen citing documents) at test time. Instead, we focus on a more realistic application scenario, where we need to predict citations for a paper which is being written and thus yet unknown. To simulate such practical settings, we ensure that all documents of the test set are unknown to the system during training. Such a scenario is challenging as it corresponds to a cold-start situation.

3 PROBLEM STATEMENT

In the following, we provide a formal problem statement for the considered recommender task. The documents can be considered users in a traditional recommendation scenario, while the items are either cited documents or subject labels, respectively.

Given a set of m documents $\mathbb{D}$ and a set of n items $\mathbb{I}$, the typical recommendation task is to model the spanned space, $\mathbb{D} \times \mathbb{I}$. We model the ratings as a sparse matrix $\boldsymbol{X} \in \{0,1\}^{m \times n}$, in which X_{jk} indicates implicit feedback from document j to item k. To simulate a real-world scenario, we split the documents $\mathbb{D}$ into m_{train} documents for training $\mathbb{D}_{\text{train}}$ and m_{test} documents for evaluation $\mathbb{D}_{\text{test}}$, such that $\mathbb{D}_{\text{train}} \cap \mathbb{D}_{\text{test}} = \emptyset$. More precisely, we conduct this split into training and test documents based on the publication year. All documents that were published before a certain year are used as training, and the remaining documents as test data. This leads to an experimental setup that is close to a real-world application. More details will be provided in Section 5.1. All models are supplied with the complete ratings of the users $\boldsymbol{X}_{\text{train}} = \mathbb{D}_{\text{train}} \bowtie \boldsymbol{X}$ along with the supplementary information $\boldsymbol{S}_{\text{train}} = \mathbb{D}_{\text{train}} \bowtie \boldsymbol{S}$ for training. In the present work, we use the title of the documents as supplementary information. Still, in theory, more sources of supplementary information may be considered. The test set $\boldsymbol{X}_{\text{test}}, \boldsymbol{S}_{\text{test}}$ is obtained analogously.

Table 1: Notation

Symbol	Description
$\mathbb{D}$	Set of m documents
$\mathbb{I}$	Set of n items
$\boldsymbol{X} \in \{0,1\}^{m \times n}$	Sparse ratings matrix
$\boldsymbol{S} \in \mathbb{R}^{m \times d}$	Supplementary document information
$\boldsymbol{x}, \boldsymbol{s}$	Row vectors of $\boldsymbol{X}$ or $\boldsymbol{S}$, respectively
$[\boldsymbol{x}; \boldsymbol{s}]$	Concatenation of vectors $\boldsymbol{x}$ and $\boldsymbol{s}$
$\bowtie$	Natural join (on document identifiers)
$\boldsymbol{I}$	Identity matrix

For evaluation, we remove randomly selected items in $\boldsymbol{X}_{\text{test}}$ by setting one non-zero entry in each row to zero. We denote the hereby created test set by $\tilde{\boldsymbol{X}}_{\text{test}}$. The model ought to predict values $\boldsymbol{X}_{\text{pred}} \in [0,1]^{m_{\text{test}} \times n}$, given the test set $\tilde{\boldsymbol{X}}_{\text{test}}$ along with the title information $\boldsymbol{S}_{\text{test}}$. Finally, we compare the predicted ratings $\boldsymbol{X}_{\text{pred}}$ with the true ratings $\boldsymbol{X}_{\text{test}}$ via ranking metrics. The goal is that those items, that were omitted in $\tilde{\boldsymbol{X}}_{\text{test}}$, are highly ranked in $\boldsymbol{X}_{\text{pred}}$.

In both scenarios, i. e., citation recommendation and subject label recommendation, we regard documents and items as a bipartite graph (see Figure 1). Considering citations, this point of view may be counter-intuitive since a scientific document is typically both a citing paper and a cited paper. Still, the out-degree of typical citation network datasets is so high that we cannot expect to have metadata for all cited papers available. For instance, the PubMed citation dataset we use for our experiments offers metadata of 224,092 documents that cite 2,896,764 distinct other documents. Therefore it is reasonable to rely only on the metadata of the citing documents itself as basis for recommendations.

4 MODELS

In the following, we describe the employed models. We start with two baselines based on item co-occurrence. Subsequently, we briefly introduce the multi-layer perceptron as a building block for the two autoencoder variants. We show how title information can be incorporated in undercomplete and adversarial autoencoders. We provide information on hyperparameters in the final paragraph of this section.

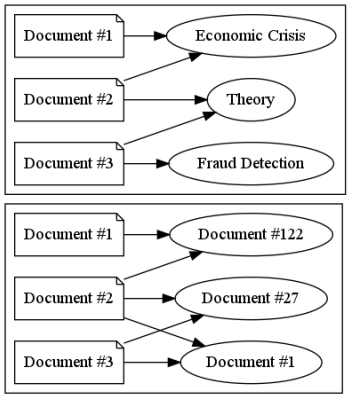

Figure 1: Exemplary bipartite graphs of documents annotated with subject labels (top) and citation relationships between documents (bottom). It becomes apparent how the two recommendation tasks share a similar structure.

Item Co-Occurrence. As a non-parametric yet strong baseline we consider the co-citation score [37] that is purely based on item co-occurrence. The rationale is that two papers, which have been cited more often together in the past, are more likely to be cited together in the future than papers that were less often cited together. Given training data $\boldsymbol{X}_{\text{train}}$, we compute the full item co-occurrence matrix $\boldsymbol{C} = {\boldsymbol{X}_{\text{train}}}^T \cdot \boldsymbol{X}_{\text{train}} \in \mathbb{R}^{n \times n}$. At prediction time, we obtain the scores by aggregating the co-occurrence values via matrix multiplication $\boldsymbol{X}_{\text{test}} \cdot \boldsymbol{C}$. On the diagonal of $\boldsymbol{C}$, the (squared) occurrence count of each item is retained to model the prior probability.

Singular Value Decomposition. Singular value decomposition (SVD) is an approach that factorizes the co-occurrence matrix of items $\boldsymbol{X}^T \cdot \boldsymbol{X}$. Caragea et al. showed that SVD can be successfully used for citation recommendation [6]. We therefore include SVD in our comparison and extend it by the capability of incorporating title information, which has already been proposed as future work by Caragea et al. [6]. We concatenate the textual features as TF-IDF weighted bag-of-words with the items and perform singular value decomposition on the resulting matrix. To obtain predictions, we only use those indices of the reconstructed matrix that are associated with items.

Multi-Layer Perceptron. A multi-layer perceptron (MLP) is a fully-connected feed-forward neural network with one or multiple hidden layers. The output is computed by consecutive applications of $\boldsymbol{h}^{(i)} = f(\boldsymbol{h}^{(i-1)} \cdot \boldsymbol{W}^{(i)} + \boldsymbol{b}^{(i)})$ with f being a nonlinear activation function. In the description of the following models, we abbreviate a two hidden-layer perceptron module by MLP-2. This MLP-2 module is not only used as a building block for subsequent architectures, but also as a full model that only operates on the documents' titles. In this case, we optimize binary cross-entropy $\text{BCE}(x, \text{MLP-2}(s))$, where the titles s are used as input and citations or subject labels x as target outputs. We chose to operate on an TF-IDF weighted embedded bag-of-words representation [10] for a fair comparison with the autoencoder variants, which are described below.

Undercomplete Autoencoders. The general concept of an autoencoder (AE) involves two components: the encoder enc and the decoder dec. The encoder transforms the input into a hidden representation (the code) $z = \text{enc}(x)$. Then the decoder reconstructs the input from the code $r = \text{dec}(z)$. The two components are jointly trained to minimize the loss function $BCE(x, r)$. To avoid learning to merely copy the input x to the output r, autoencoders need to be regularized. The most common way to regularize autoencoders is by imposing a lower dimensionality on the code (undercomplete). In short, autoencoders are trained to capture the most important explanatory factors of variation for reconstruction [4].

For both the encoder and the decoder we chose an MLP-2 module, such that the model function becomes $r = \text{MLP-2}_{\text{dec}}(\text{MLP-2}_{\text{enc}}(x))$. When the documents' title is available, we supply it as additional input to the decoder $r = \text{MLP-2}_{\text{dec}}([\text{MLP-2}_{\text{enc}}(x); s])$. We embed the textual features into a lower dimensional space by using pre-trained word embeddings [26]. The rationale here is that the rather low code dimension is not overwhelmed by the high amount of vocabulary terms. For a fair comparison of the models, also the MLP described above is supplied the same text representation as input. More precisely, we employ a TF-IDF weighted bag of embedded words representation which has proven to be useful for information retrieval [10]. The usage of title information in an undercomplete autoencoder is comparable to the approach by Barbieri et al. [1]. A minor difference is that we supply the side information (titles) only to the decoder, yet use two hidden layers for both the encoder and the decoder to enable a fair comparison to the adversarial variant, which is described below.

Adversarial Autoencoders. We extend the work of Makhzani et al. on adversarial autoencoders (AAE) [24], who combine generative adversarial networks [14] with autoencoders. The autoencoder component reconstructs the sparse item vectors, while the discriminator distinguishes between the generated codes and samples from a selected prior distribution (see Figure 2). Hence, the distribution of the latent code is shaped to match the prior distribution. We hypothesize that the latent representations learned by distinguishing the code from a smooth prior lead to a model that is more robust to sparse input vectors than undercomplete autoencoders. The rationale is that smoothness is a main criterion for good representations that disentangle the explanatory factors of variation [4].

Formally, we first compute $h = \text{MLP-2}_{\text{enc}}(x)$ and $r = \text{MLP-2}_{\text{dec}}(h)$ and then update the parameters of the encoder and the decoder with respect to binary cross-entropy $\text{BCE}(x, r)$. Hence, in the regularization phase, we draw samples $z \sim \mathcal{N}(0, \boldsymbol{I})$ from independent Gaussian distributions matching the size of h. The parameters of the discriminator $\text{MLP-2}_{\text{disc}}$ are then updated, to minimize $\log \text{MLP-2}_{\text{disc}}(z) + \log(1 - \text{MLP-2}_{\text{disc}}(h))$ [14]. Finally, the parameters of the encoder are updated to maximize $\log \text{MLP-2}_{\text{disc}}(h)$, such that the encoder is trained to fool the discriminator. As a result, the encoder is jointly optimized for matching the prior distribution and for reconstruction of the input [24].

To incorporate the documents' title, we once again concatenate on the code level. This scenario corresponds to the supervised case from the original work of Makhzani et al. on images, in which the purpose was to separate the style from the class. All information that cannot be reconstructed from the class is drawn from the

style (the code) [24]. We adapt this interpretation by supplying title information as additional input to the decoder. Hence, the model is optimized to exploit the title information when it is helpful for reconstruction but also take the partial item set into account. At prediction time, we perform one reconstruction step by applying one encoding and one decoding step.

Hyperparameters. The hyperparameters are selected by conducting pre-experiments on the citation recommendation dataset by considering only items that appear 50 or more times in the whole corpus. We chose this scenario because this aggressive pruning results in numbers of distinct items and documents that are similar to the ones of the subject label recommendation dataset. Considering the MLP-modules, we conducted a grid search with hidden layer sizes between 50 and 1,000, initial learning rates between 0.01 and 0.00005, activation functions Tanh, ReLU [28], SELU [22] along with dropout [38] (or alpha-dropout in case of SELUs) probabilities between .1 and .5 and as optimization algorithms stochastic gradient descent and Adam [19]. For the autoencoder-based models, we considered code sizes between 10 and 500, but only if the size was smaller than the hidden layer sizes of the MLP modules. In case of adversarial autoencoders, we experimented with Gaussian, Bernoulli, and Multinomial prior distributions, and with linear, sigmoid, and softmax activation on the code layer, respectively.

While we do not exclude that a certain set of hyperparameters may perform better in a specific scenario, we select the following, most robust, hyperparameters: hidden layer sizes of 100 with ReLU [28] nonlinearities and drop probabilities of .2 after each hidden layer. The optimization is carried out by Adam [19] with initial learning rate 0.001. The two autoencoder variants use a code size of 50. We further select a Gaussian prior distribution for the adversarial autoencoder. For SVD, we consecutively increased the number of singular values up to 1,000. Using higher amounts of singular values decreased the performance. We keep this set of hyperparameters fixed across all models and across all subsequent experiments to ensure a reliable comparison of the models' quality.

5 EXPERIMENTS

To evaluate adversarial autoencoders for recommendation tasks on scientific documents, we conduct a citation recommendation experiment as presented in Section 5.1 and a subject label recommendation experiment as presented in Section 5.2. Adversarial autoencoders are not only evaluated against the two baselines (item co-occurrence and SVD), but also against its own components: undercomplete autoencoders and multi-layer perceptrons.

5.1 Citation Recommendation

In this section, we describe our experimental setup which is designed to resemble a real-world application of missing citation recommendation.

Dataset. The CITREC[4] PubMed citation dataset [13] consists of 7,546,982 citations. The dataset comprises 224,092 distinct citing documents published between 1928 and 2011 and 2,896,764 distinct cited documents. The documents are cited between 1 and 3,247 times with a median of 1 and a mean of 2.61 (SD: 6.71). The citing documents hold on average 33.68 (SD: 27.49) citations to other documents (minimum: 1, maximum 2,242) with a median of 29.

[4]https://www.isg.uni-konstanz.de/projects/citrec/

Table 2: Dataset characteristics with respect to pruning thresholds on minimum item occurrence for the PubMed citation recommendation task.

pruning	cited documents	citations	documents	density
15	35,664	1,173,568	136,911	0.000240
20	20,270	878,359	121,374	0.000357
25	12,881	692,037	105,170	0.000511
30	8,906	568,563	96,980	0.000658
35	6,469	478,693	87,498	0.000846
40	4,939	413,746	79,830	0.001049
45	3,904	363,870	73,200	0.001273
50	3,185	324,693	67,703	0.001506
55	2,643	292,791	62,647	0.001768

Split on Time Axis. To simulate a real-world citation prediction setting, we split the data on the time axis of the citing documents. This resembles the natural constraint that publications cannot cite other publications that do not exist yet. Given a specific publication year T, we ensure that the training set $\boldsymbol{D}_{\text{train}}$ consists of all documents that were published earlier than year T and use the remaining documents as test data $\boldsymbol{D}_{\text{test}}$. Figure 3 shows the distribution of documents over the years along with the split into training and test set. We select the year 2011 for evaluation to obtain a 90:10 ratio between training and test documents.

Preprocessing and Dataset Pruning as Controlled Variable. For preprocessing the datasets, we conduct the following three steps:

(1) Build a vocabulary on the training set with items that received implicit feedback more than α times.
(2) Filter both the training and test set and retain only items from the vocabulary.
(3) Remove documents that are assigned to fewer than two of the vocabulary items.

The pruning threshold α is crucial since it affects both the number of considered items as well as the number of documents. Thus, we identify α as a controllable parameter and evaluate the models' performance with respect to different values for α. Table 2 shows the dataset characteristics with respect to the pruning threshold.

Evaluation Metric. For evaluation, certain items were omitted on purpose in the test set. For each document, the models ought to predict the omitted item as good as possible. Thus, we choose mean reciprocal rank as our evaluation metric. We are given a set of predictions $\boldsymbol{X}_{\text{pred}}$ for the test set $\tilde{\boldsymbol{X}}_{\text{test}}$. Hence for each row, we compute the reciprocal rank of the missing element from $\boldsymbol{x}_{\text{test}} - \tilde{\boldsymbol{x}}_{\text{test}}$. The reciprocal rank corresponds to one divided by the position of the omitted item in the sorted list of predictions $\boldsymbol{x}_{\text{pred}}$. We then average over all documents of the test set to obtain the mean reciprocal rank. To alleviate random effects of model initialization, training data shuffling, and selecting the elements to omit, we conduct three runs for each of the experiments. To allow a fair comparison, the

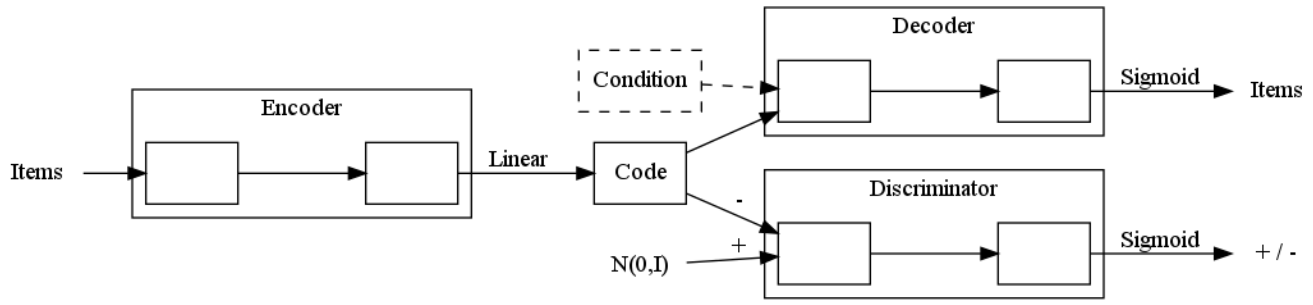

Figure 2: Adversarial autoencoder for item-based recommendations. Each edge resembles a parametrized mapping $f(\boldsymbol{W}\boldsymbol{x} + \boldsymbol{b})$ with activation function f and parameters W, b. When not labeled differently, the activation function is rectified linear followed by dropout.

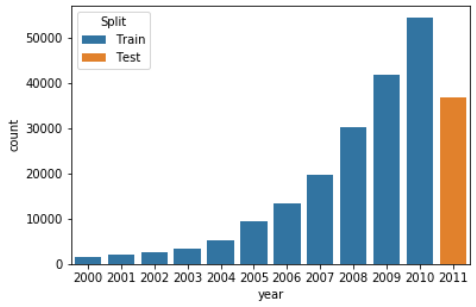

Figure 3: Count of documents by publication year starting with 2000 along with the split in training and test set for the PubMed citation dataset.

removed items in the test set remain the same for all models during one run with a fixed pruning parameter.

Results. Figure 4 shows the results for the models with respect to the pruning parameter that controls the number of considered items as well as the sparsity (see Table 2). We observe a trend that a more aggressive pruning threshold leads to higher scores among all models. When no title information is given, the item co-occurrence approach consistently yields the highest scores. When title information is available, adversarial autoencoders become competitive to the item co-occurrence approach and yield higher scores than all of their components.

5.2 Subject Label Recommendation

On the basis of our experience in multi-label classification [9, 15], we now consider a subject label recommendation task, which is close to how professional subject indexers work.

Dataset. The EconBiz dataset provided by ZBW — Leibniz Information Centre for Economics consists of 61,619 documents with label annotations from professional subject indexers [9, 15]. The

Table 3: Dataset characteristics with respect to pruning thresholds on minimum item occurrence for the EconBiz subject label recommendation task.

pruning	labels	assigned labels	documents	density
1	4,568	323,670	61,104	0.001160
2	4,103	323,060	61,090	0.001289
3	3,760	322,199	61,060	0.001403
4	3,497	321,213	61,039	0.001505
5	3,259	320,048	60,983	0.001610
10	2,597	314,738	60,778	0.001994
15	2,192	309,101	60,524	0.002330
20	1,924	303,693	60,272	0.002619

4,669 assigned labels are a subset of the controlled vocabulary Standardthesaurus Wirtschaft[5]. The number of documents to which a label is assigned ranges between 1 and 13,925 with mean 69 (SD: 316) and median 14. The label annotations of a document ranges between 1 and 23 with mean 5.24 (SD: 1.83) and median 5 labels.

Evaluation. The preprocessing steps and evaluation procedure for the subject label recommendation task is the same as in Section 5.1. We also conduct the split between training set and test set on the time axis (see Figure 5). This is challenging because label annotations suffer from concept drift over time [39]. We use the years 2012 and 2013 as test documents to obtain a train-test ratio similar to the scenario in Section 5.1. The dataset characteristics affected by dataset pruning are given in Table 3.

Results. Figure 6 shows the results for the models with respect to the pruning parameter that controls the number of considered items and therefore also the sparsity (see Table 3). When no title information is available, the adversarial autoencoder is competitive to the item co-occurrence approach. When title information is given, the adversarial autoencoder yields considerably higher scores than all models operating without this information. The sole decoder part (an MLP-2 module) of the adversarial autoencoder yields, however, consistently higher scores than the model as a whole.

[5]http://zbw.eu/stw/version/latest/about

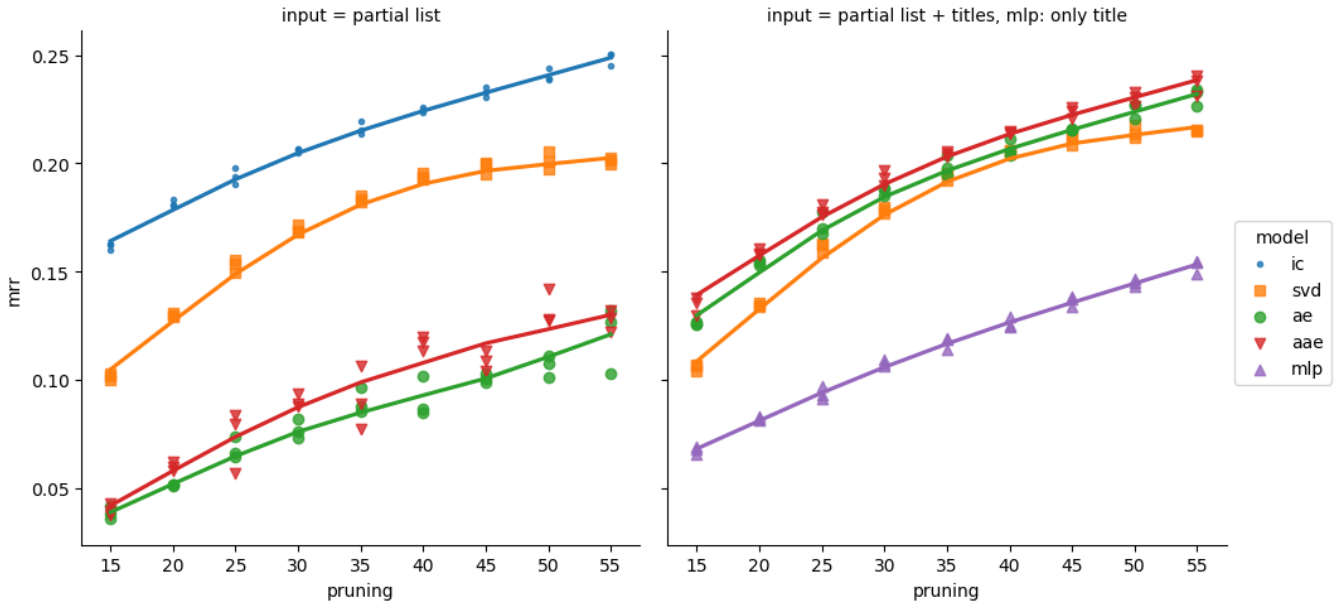

Figure 4: Mean reciprocal rank of missing citation on the test set with varying minimum item occurence (pruning) thresholds. Left: Only the partial list of items is given. Right: The partial list of items along with the document title is given, except for the MLP, which can only make use of the title.

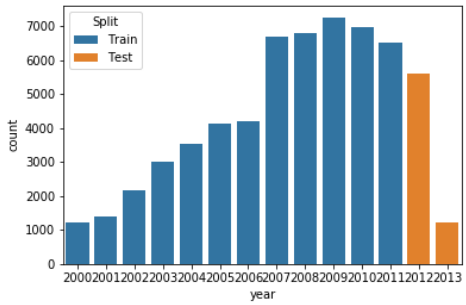

Figure 5: Count of documents by publication year starting with 2000 along with the split in training and test set for the Economics subject label dataset.

6 DISCUSSION

We have evaluated adversarial autoencoders for two different recommendation tasks on scientific documents with varying input modalities and varying numbers of considered items. Our results reveal relationships between the type of recommendation task and the input modalities. On the citation task, the partial list of citations is relevant to recommend potentially missing citations. For the subject label recommendation task, however, using solely the decoder on the title information yields even better performance than the whole model. Thus, our experiments show in which cases adversarial autoencoders are beneficial. On the citation recommendation task the title information enables adversarial autoencoders to become competitive to the strong baseline from co-citation analysis. The effect of the adversarial regularization component is marginal, yet leads to a consistent improvement over traditional, undercomplete autoencoders. By imposing different thresholds on minimum item occurrence, we varied the number of considered items and thus, the degree of sparsity. We observe that all considered models are similarly affected by the increased difficulty caused by higher numbers of considered items, despite the high amount of parameters.

Even though it is not surprising that co-citation count is highly relevant for citation recommendation [37], we have shown that adversarial autoencoders have a conceptual benefit: they offer the capability of exploiting additional information along with the partial list of citations. From the perspective of the model, it is of high importance to learn about the prior distribution of the data, which explains the strength of the item co-occurrence baseline. Autoencoders retain this benefit and may learn to put appropriate weights in the bias parameters if it is helpful for the overall objective. We envision that further types of information, such as the authors and publication year may further increase the overall performance.

Compared to item co-occurrence or singular value decomposition, all neural network approaches have a large number of learnable parameters as well as hyperparameters that require tuning. To assess the quality of the model itself, we used a fixed set of hyperparameters across all experiments and conducted multiple runs of the same experimental setup to alleviate random effects in initialization and shuffling.

On the subject recommendation task, we observed that the MLP decoder alone yields higher mean reciprocal rank scores than the adversarial autoencoder. Thus, already assigned subjects are less informative for a subject recommendation task than the titles are. This can be explained by a specific guideline for subject indexers

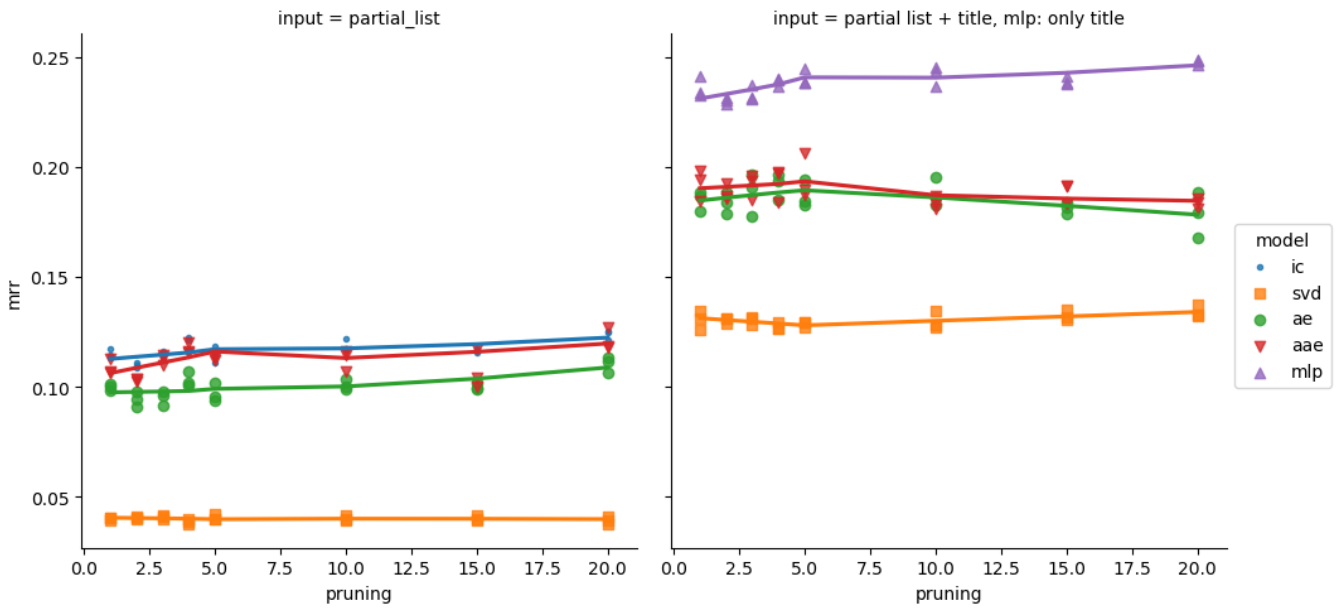

Figure 6: Mean reciprocal rank of missing subject label on the test set with varying minimum item occurence thresholds. Left: Only the partial list of items is given. Right: The partial list of items along with the document title is given, except for the MLP, which can only make use of the title.

working on the specific EconBiz dataset that we used for our experiments: when two or more subjects with a common ancestor in the hierarchical thesaurus of subjects match, it is preferred to assign the ancestor instead of the child subjects [15]. Thus, two subjects that are semantically related because they share a common ancestor are, because of the guideline, unlikely to co-occur in the annotations of a single document.

We conducted 408 experiments over two different recommendation tasks with different input modalities and varying degrees of sparsity. While it is a limitation that we only use one dataset per task, this enabled us to investigate the interactions across tasks, input modalities and the effect of sparsity. As a result, we can state that, on the one hand, there are tasks in which co-occurrence implies relatedness. On the other hand, there are recommendation tasks, in which co-occurrence of items rather implies diversity.

In the present work, we used one prototypical task for each of these two types of recommendations, i. e., citations, where it is known that co-citation reflects relatedness of the cited resources [2, 37], and subject labels, where the guidelines of subject indexers suggest that semantically related subjects are less likely to co-occur. We have carefully investigated the interaction between the semantics of item co-occurrence and supplying the partial list of items as input for a recommender system.

For practical recommender systems, the present work offers evidence that the aforementioned semantics of item co-occurrence is relevant for the decision, whether the partial list of items should be supplied to a recommendation model as input. We have shown that also on recommendation tasks, adversarial autoencoders consistently outperform their traditional, undercomplete counterpart and how additional information can be incorporated in such models. Our results show that both models with no learnable parameters and models with a high amount of learnable parameters are equally sensitive to the number of considered items, which we controlled by pruning the datasets with respect to minimum item occurrence.

7 CONCLUSION

We conclude that the different semantic interpretation of item co-occurrence in recommendation tasks highly affects the preferable input modalities. When item co-occurrence resembles relatedness such as in citations, supplying the list of already cited documents is beneficial for the overall performance. For subject recommendations, we observe that co-occurring subjects does not imply that these subjects are semantically similar. Rather, the document's subject needs to be described by multiple, diverse subject annotations. In such cases, we have shown that a single multi-layer perceptron component that operates only on the documents' titles is stronger than the whole adversarial autoencoder. We have shown that adversarial autoencoders consistently outperform undercomplete autoencoders, and that their capability of incorporating multiple input modalities offers a conceptual benefit.

Reproducibility. The source code for reproducing our experiments is openly available on GitHub[6].

ACKNOWLEDGMENTS

This work was supported by the German Research Foundation under project number 311018540 (Linked Open Citation Database) as well as by the EU H2020 project MOVING (contract no 693092).

[6]https://github.com/lgalke/aae-recommender

REFERENCES

[1] Julio Barbieri, Leandro G. M. Alvim, Filipe Braida, and Geraldo Zimbrão. 2017. Autoencoders and recommender systems: COFILS approach. *Expert Syst. Appl.* 89 (2017), 81–90.

[2] Joeran Beel, Bela Gipp, Stefan Langer, and Corinna Breitinger. 2016. paper recommender systems: a literature survey. *International Journal on Digital Libraries* 17, 4 (2016), 305–338.

[3] Jöran Beel, Stefan Langer, Bela Gipp, and Andreas Nürnberger. 2014. The Architecture and Datasets of Docear's Research Paper Recommender System. *D-Lib Magazine* 20, 11/12 (2014).

[4] Yoshua Bengio, Aaron C. Courville, and Pascal Vincent. 2012. Unsupervised Feature Learning and Deep Learning: A Review and New Perspectives. *CoRR* abs/1206.5538 (2012).

[5] Johan Bollen and Herbert Van de Sompel. 2006. An architecture for the aggregation and analysis of scholarly usage data. In *JCDL*. ACM, 298–307.

[6] Cornelia Caragea, Adrian Silvescu, Prasenjit Mitra, and C. Lee Giles. 2013. Can't see the forest for the trees?: a citation recommendation system. In *JCDL*. ACM, 111–114.

[7] Klaas Dellschaft and Steffen Staab. 2012. Measuring the influence of tag recommenders on the indexing quality in tagging systems. In *HT*. ACM, 73–82.

[8] Travis Ebesu and Yi Fang. 2017. Neural Citation Network for Context-Aware Citation Recommendation. In *SIGIR*. ACM, 1093–1096.

[9] Lukas Galke, Florian Mai, Alan Schelten, Dennis Brunsch, and Ansgar Scherp. 2017. Using Titles vs. Full-text as Source for Automated Semantic Document Annotation. In *K-CAP*. ACM, 20:1–20:4.

[10] Lukas Galke, Ahmed Saleh, and Ansgar Scherp. 2017. Word Embeddings for Practical Information Retrieval. In *GI-Jahrestagung (LNI)*, Vol. P-275. GI, 2155–2167.

[11] Andreas Geyer-Schulz, Michael Hahsler, and Maximilian Jahn. 2002. Recommendations for virtual universities from observed user behavior. In *Classification, Automation, and New Media*. Springer, 273–280.

[12] C. Lee Giles, Kurt D. Bollacker, and Steve Lawrence. 1998. CiteSeer: An Automatic Citation Indexing System. In *ACM DL*. ACM, 89–98.

[13] Bela Gipp, Norman Meuschke, and Mario Lipinski. 2015. CITREC: An Evaluation Framework for Citation-Based Similarity Measures based on TREC Genomics and PubMed Central. In *Proceedings of the iConference 2015*. Newport Beach, California. http://ischools.org/the-iconference/

[14] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron C. Courville, and Yoshua Bengio. 2014. Generative Adversarial Nets. In *NIPS*. 2672–2680.

[15] Gregor Große-Bölting, Chifumi Nishioka, and Ansgar Scherp. 2015. A Comparison of Different Strategies for Automated Semantic Document Annotation. In *K-CAP*. ACM, 8:1–8:8.

[16] Aditya Grover and Jure Leskovec. 2016. node2vec: Scalable Feature Learning for Networks. In *KDD*. ACM, 855–864.

[17] Wenyi Huang, Saurabh Kataria, Cornelia Caragea, Prasenjit Mitra, C. Lee Giles, and Lior Rokach. 2012. Recommending citations: translating papers into references. In *CIKM*. ACM, 1910–1914.

[18] Wenyi Huang, Zhaohui Wu, Liang Chen, Prasenjit Mitra, and C. Lee Giles. 2015. A Neural Probabilistic Model for Context Based Citation Recommendation. In *AAAI*. AAAI Press, 2404–2410.

[19] Diederik P. Kingma and Jimmy Ba. 2014. Adam: A Method for Stochastic Optimization. *CoRR* abs/1412.6980 (2014).

[20] Thomas N. Kipf and Max Welling. 2016. Semi-Supervised Classification with Graph Convolutional Networks. *CoRR* abs/1609.02907 (2016).

[21] Thomas N. Kipf and Max Welling. 2016. Variational Graph Auto-Encoders. *CoRR* abs/1611.07308 (2016). http://arxiv.org/abs/1611.07308

[22] Günter Klambauer, Thomas Unterthiner, Andreas Mayr, and Sepp Hochreiter. 2017. Self-Normalizing Neural Networks. In *NIPS*. 972–981.

[23] Ralf Krestel, Peter Fankhauser, and Wolfgang Nejdl. 2009. Latent dirichlet allocation for tag recommendation. In *RecSys*. ACM, 61–68.

[24] Alireza Makhzani, Jonathon Shlens, Navdeep Jaitly, and Ian J. Goodfellow. 2015. Adversarial Autoencoders. *CoRR* abs/1511.05644 (2015).

[25] Sean M. McNee, Istvan Albert, Dan Cosley, Prateep Gopalkrishnan, Shyong K. Lam, Al Mamunur Rashid, Joseph A. Konstan, and John Riedl. 2002. On the recommending of citations for research papers. In *CSCW*. ACM, 116–125.

[26] Tomas Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. 2013. Distributed Representations of Words and Phrases and their Compositionality. In *NIPS*. 3111–3119.

[27] Elena Montañés, José Ramón Quevedo, Irene Díaz, and José Ranilla. 2009. Collaborative Tag Recommendation System based on Logistic Regression. In *DC@PKDD/ECML (CEUR Workshop Proceedings)*, Vol. 497. CEUR-WS.org.

[28] Vinod Nair and Geoffrey E. Hinton. 2010. Rectified Linear Units Improve Restricted Boltzmann Machines. In *ICML*. Omnipress, 807–814.

[29] Jinseok Nam, Eneldo Loza Mencía, Hyunwoo J. Kim, and Johannes Fürnkranz. 2017. Maximizing Subset Accuracy with Recurrent Neural Networks in Multi-label Classification. In *NIPS*. 5419–5429.

[30] Enrico Palumbo, Giuseppe Rizzo, and Raphaël Troncy. 2017. entity2rec: Learning User-Item Relatedness from Knowledge Graphs for Top-N Item Recommendation. In *RecSys*. ACM, 32–36.

[31] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. 2014. DeepWalk: online learning of social representations. In *KDD*. ACM, 701–710.

[32] Lisa Posch, Claudia Wagner, Philipp Singer, and Markus Strohmaier. 2013. Meaning as collective use: predicting semantic hashtag categories on twitter. In *WWW (Companion Volume)*. International World Wide Web Conferences Steering Committee / ACM, 621–628.

[33] Massimo Quadrana, Alexandros Karatzoglou, Balázs Hidasi, and Paolo Cremonesi. 2017. Personalizing Session-based Recommendations with Hierarchical Recurrent Neural Networks. In *RecSys*. ACM, 130–137.

[34] Jessica Rosati, Petar Ristoski, Tommaso Di Noia, Renato De Leone, and Heiko Paulheim. 2016. RDF Graph Embeddings for Content-based Recommender Systems. In *CBRecSys@RecSys (CEUR Workshop Proceedings)*, Vol. 1673. CEUR-WS.org, 23–30.

[35] Shilad Sen, Jesse Vig, and John Riedl. 2009. Tagommenders: connecting users to items through tags. In *WWW*. ACM, 671–680.

[36] Börkur Sigurbjörnsson and Roelof van Zwol. 2008. Flickr tag recommendation based on collective knowledge. In *WWW*. ACM, 327–336.

[37] Henry Small. 1973. Co-citation in the scientific literature: A new measure of the relationship between two documents. *JASIS* 24, 4 (1973), 265–269.

[38] Nitish Srivastava, Geoffrey E. Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research* 15, 1 (2014), 1929–1958.

[39] Martin Toepfer and Christin Seifert. 2017. Descriptor-Invariant Fusion Architectures for Automatic Subject Indexing. In *JCDL*. IEEE Computer Society, 31–40.

[40] Hao Wang, Naiyan Wang, and Dit-Yan Yeung. 2015. Collaborative Deep Learning for Recommender Systems. In *KDD*. ACM, 1235–1244.

[41] Paulus Franciscus Wouters et al. 1999. *The citation culture*. Ph.D. Dissertation. Universiteit van Amsterdam. https://pure.uva.nl/ws/files/3164315/8231_13.pdf