Investigating Biases in News Recommendation

Fakultät für Elektrotechnik und Informatik

Institut für Data Science

Leibniz Universität Hannover

Masterarbeit

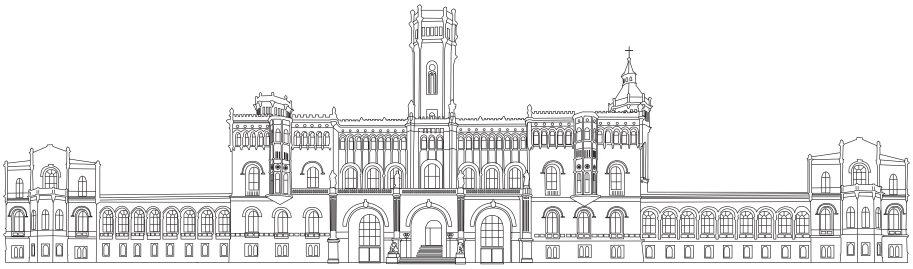
submitted for the degree of

Master of Science (M. Sc.)

by

Florian Langenhagen

Matriculation Number : 2957330



First Examiner: Prof. Dr. Wolfgang Nejdl

Second Examiner: Dr. Sowmya S. Sundaram

Supervised By: M. Sc. Jonas Wallat

January 2, 2023

Erklärung der Selbständigkeit

Hiermit versichere ich, dass ich diese Arbeit selbstständig verfasst habe, keine anderen als die angegebenen Quellen und Hilfsmittel benutzt wurden, alle Stellen der Arbeit, die wörtlich oder sinngemäß aus anderen Quellen übernommen wurden, als solche kenntlich gemacht sind, und die Arbeit in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegt habe.

Hannover, January 2, 2023

Florian Langenhagen

Abstract

With increasing automation and the continuous development of machine learning, modern algorithms are now used in almost all areas to improve and simplify workflows. Recommendation Systems (RS) are one group of these algorithms. They enable automated suggestions of items based on the interests of the user. In this work, we will focus on the investigation of biases in recommendation systems for news. News Recommendation Systems (NRS) provide a way to suggest targeted news to users according to their needs. As news is the primary source of information, it is imperative that it is presented fairly and free from bias. Thus, in addition to good recommendations for the user, novelty and diversity should be ensured by the NRS. For this purpose, several experiments are conducted with the MIND dataset, which has collected 1,000,000 users' data on MSN. This work gives an overview of the different biases in the feedback loop of news recommendation systems. In particular, data and model biases are examined and related to other user biases. The research should enable a template for bias modeling. All tests are presented in a Github repository `https://github.com/LaKin314/Investigating_biases_in_News_Rec`.

Contents

List of Tables

List of Figures

Acronyms

BiRNN Bidirectional Recurrent Neural Network. 16

CNN Convolutional Neural Network. 7, 8

DNK Deep Knowledge-Aware Network for News Recommendation. 7, 33, 37, 57

i.i.d. independent and identically distributed. 10, 16

LSTUR Neural News Recommendation with Long- and Short-term User Representations. 7, 37, 57

MAUP Modifiable Areal Unit Problem. 23

NAML Neural News Recommendation with Attentive Multi-View Learning. 7, 33, 37, 51, 57, VIII

NPA Neural News Recommendation with Personalized Attention. 7, 33, 37, 57

NRMS Neural News Recommendation with Multi-Head Self-Attention. 2, 7, 32, 33, 37, 44, 47, 51, 53, 57

NRS News Recommendation Systems. 1, 2, 5, 6, 9, 32, 34, 57, V

RNN Recurrent Neural Network. 8, 15, 16

RS Recommendation Systems. 1, 2, 6, 31, 32, 34, 38, V

1 Introduction

Recommendation Systems (RS) are a major sub-field of artificial intelligence on the internet. For example, websites like Amazon are trying to learn the interests of their users and suggest targeted products to increase sales. Spotify tries to suggest new songs using popular songs, and Netflix identifies similar movies for users. Another big area is news recommendations. Reading news on the internet is increasingly becoming the primary medium and is slowly replacing traditional sources such as television or newspapers. An empirical study by Eurostat [30] gives evidence for this. Filistrucchi [31] also shows in his paper that in Italy, for example, traditional newspapers offer online formats, which leads to a decline in the newspaper medium. Newspapers like the New York Times[1] make their content available and sites like Google[2] or MSN[3] aggregate articles from different providers. It allows providers to track and profile their users [18]. The data is then used to learn the interests of users. However, pure interest cannot be extracted from a user's previous behavior. This is because users have a distorted perception of the information, the so-called cognitive biases. This means, for example, that a user looks at a news item because it looks appealing to him and not because he is interested in the topic [129]. Another example is the change of interest of a user. Previously interesting news may be less interesting at a later time [79]. These biases are implicitly picked up during data collection. Biases must be considered in the data collection or modeled by the underlying model. Otherwise, the NRS learns based on skewed data, leading to worse or unfair recommendations. As NRS in particular focuses on informing, it is important that news are diverse [48, 15] and new [95] despite tailored interests.

The subject matter of this paper is roughly divided into two parts.

First of all, it is about the study of biases. On the user side, we speak of cognitive biases. They influence people's decisions. Therefore, this topic is strongly influenced by psychological aspects such as subjective perception [63]. The topic is mainly researched through user studies in psychology, especially marketing psychology (e.g. [89, 29]).

In contrast, biases in data and models are a topic of computer science, especially data research. Thus, this research is distinct from psychological user bias. Data biases also arise from users and are manifested in data (e.g. [78]). This changes the way models deal with the data. It leads to misconceptions and losses in the performance of models [21]. There-

[1] www.nytimes.com

[2] www.news.google.com

[3] www.msn.com/de-de/nachrichten

fore, research is being conducted to describe these biases, detect them and eliminate them through debiasing methods [20, 103].

Current research already provides some overviews of bias in machine learning [79] and RS [21]. The topic of fairness is addressed in many papers [73, 5, 16, 33]. These primarily focus on specific biases. For example, Ma and Dong [73] provides a way to mitigate the popularity bias generated by conformance and the resulting unfairness for providers. Qi et al. [94] also propose a way of enabling provider fairness. However, this is not always verifiable. News aggregators like Google News are private companies and do not make their models publicly available for viewing. This reinforces the desire for fairness. Since the recommendations of NRS directly impact users and their view of society, ethical aspects also play a role. These are highlighted by Milano et al. [82].

Starting with our work, we define mathematically what a NRS is, which stages it has and which different forms exist. To get a basic understanding of NRS, we explain the differences in the architectures of content-based and collaborative RS. In addition, the most important basics for the neural networks behind it are explained. All potential biases are listed and explained once to get an overall view. These are divided into the three stages of NRS. After explaining the basics, we conduct experiments to identify potential biases between data-model and model-user. In this work, we use the MIND dataset by Wu et al. [126] and the Recommenders[4] library from Microsoft. MIND is a collection of user actions on the news site MSN[5] with 1,000,000 user entries. This data is examined for data bias. Neural News Recommendation with Multi-Head Self-Attention (NRMS) [120] is used mostly, but other models are also briefly presented and compared. It creates news and user embeddings using multi-head self-attention to learn contextual information. Embedded candidates are then compared with the user embedding, which results in a click prediction. We use it for the investigation of possible model biases.

Since experiments are based on the collected data from MIND, some statistics are presented on the data, which helps to interpret the results better. The list of biases combines current research and review papers for general RS and cognitive biases. These are explained again in the following section. In addition, our work aims to clarify to what extent the size of the user history affects the model's performance. To do this, we look at different sizes of user histories and check whether these users have the same chance of getting good recommendations. We call this issue history-length unfairness. It provides a generalization of the extreme form of the user-based cold-start problem [42], which describes the reduction in performance for users without history. Although the experiments are limited to data and model biases, we discuss potential user biases in detail and draw connections to data biases. Finally, a list of all biases experimented with and discussed will be presented in our work.

In summary, this work aims to clarify the extent and in what form biases can occur in NRS.

[4] www.github.com/Microsoft/Recommenders

[5] https://www.msn.com/en-us

Unlike other research, we present a comprehensive checklist of the potential user, data, and model biases. To illustrate this, we will test the theoretical findings using the MIND database provided by Microsoft. This provides insight into the problem of bias with implicit feedback.

2 Related Works

2.1 Biases Investigation

Originally, bias research comes from pure psychology and refers to the cognitive biases of human [19]. Therefore it often serves as a basis for research in other fields. For example, Caverni et al. [19] provides an introduction to the psychological side of the issue. A frequently cited article by Haselton et al. [44] provides an overview of various cognitive biases. In addition, a very detailed list of cognitive biases and psychological effects is provided [66]. This is illustrated and can be viewed in Figure A.1.

More research is being done in other areas. For example, the extent to which cognitive biases change clinical decisions in medicine is being researched frequently [101, 116, 85, 87]. Saposnik et al. [101], for example, addresses the problems of anchoring and the framing effect. It causes physicians to be fixed on the first symptoms they see, leading to wrong diagnoses and patient treatments. This shows a strong interest in the topic of medicine. Other fields such as marketing and economics also greatly interest this topic. For example, Otuteye and Siddiquee [89] and Dowling et al. [29] explain how investment decisions are influenced by bias. Guiso et al. [41] are dedicated to the very popularising topic of cultural bias in business. It shows unfairness towards other cultures.

In machine learning, the work of Mehrabi et al. [79] offers an excellent insight into the topic. They describe the feedback loop of user interactions, data, and algorithms and list different biases at each level. It gives an additional idea of biases according to data and model.

In this master thesis, we refer to many biases of the paper, such as longitudinal data fallacy [13] or exposure bias [68]. Many kinds of research directly address specific biases. Heckman [47] examines the need for random sampling and defines the resulting selection bias. Li and Vasconcelos [62] provides a framework called REPAIR that addresses forms of representation bias. Arazo et al. [10] discusses confirmation bias in semi-supervised learning.

Specifically for recommender systems and news recommendation systems, research is somewhat more advanced. Here, too, a survey paper provides a good basic structure for our research. The paper by Chen et al. [21] deals with biases in recommendation systems and shows some debiasing methods, following the structure of the three stages in NRS and divide their work accordingly. They also describe the feedback loop amplification and possible performance losses with debiasing methods. Among other things, data, selection on rating [78, 90], exposure [68, 67], position [90, 25] and popularity bias [73] are described. Ma and

Dong [73] draw a link between popularity bias and conformity bias. They speak of conformity in selecting items in terms of popularity, which entails a stronger unfairness towards providers of the items. Thus, in their definition, conformity bias is not limited to explicit feedback, contrary to the definition of Krishnan et al. [61] which show that conformity exists concerning rating systems. They also refer to it as social influence bias.

Mansoury et al. [77] uses a graph-based method to reduce exposure bias to care for provider fairness referring to their earlier paper [4] on multi-sided exposure bias. The method is done post-processing and can be used in addition to an existing RS. It takes recommendations of specific length and resizes them, ending up with fewer but refined recommendations. The results increase fairness via mitigating exposure bias, but on a loss of accuracy. In a different paper, they also show that there is an amplification of these biases due to the feedback-loop [75]

Olteanu et al. [88] provide a list of biases for social data datasets such as Facebook[1]. They show how biases occur and provide a framework for detecting these in a theoretical way.

Ovaisi et al. [90] describes the position and selection bias for implicit feedback and is able to mitigate the issue using a "Learning-To-Rank System" with a relevancy score. Wu et al. [121] also use a debias model to minimize position bias. They use a generative adversarial network[37] for this purpose. It consists of a bias-aware and an unbiased models, which generates random positions for items. The user representation, which is learned with the help of a transformer model [111], is improved with adversarial learning.

To reduce selection bias Liu et al. [71] use the successful contrastive learning [22, 50] method to learn debiased representations. These are learned by self-supervised learning.

Ji et al. [54] have a look at whether loyal users benefit from better recommendations. To do this, they define loyalty in three ways: number of accumulated interactions, active time, and recency. The results show that the opposite is the case and that recency is the most important factor for good recommendations.

Since most News Recommendation Systems use message titles and other textual information to predict interest, Alam et al. [7] check news for potential biases in sentiment and stance. To do this, they use a Transformer BERT classifier [111, 28] for classification. They show a slight tendency towards negativity on sentiment. For the stance sentiment, they looked at the opinion on refugee policy. Showing a small tendency to stance against it.

Qi et al. [94] show that there is a provider unfairness in news models. They present a solution in which unbiased representations are learned using adversarial learning Goodfellow et al. [38]. It uses a content model to learn news representations and back-propagates the adversarial loss. On the other side, a provider discriminator is trained to decide whether the content is fair or not. It is trained on the back-propagated discrimination loss.

Finally, Heitz et al. [48] tested the impact of a diversity-aware NRS on users utilizing a user study. They built three groups of users using a news app. Depending on the group,

[1] www.facebook.com

they received news based on optimal accuracy, optimal diversity, and chronology. The users were ranked before and after the tests according to political leanings. The results show that users with diversity-optimized news deviate more from their usual interests and thus become politically depolarised.

2.2 News Recommendations Systems

News Recommendation Systems are now based on modern machine-learning approaches and are improving with research successes in this field. Many models have moved from classical collaborative filter models to hybrid model architectures. These make use of content information or knowledge graphs. Many models leverage success in Natural Language Processing and use embedding such as Word2Vec [80] and GloVe [92] to encode news [120, 113, 118]. These allow similar news to be displayed close together in the embedding space. The latest models with pre-trained Transformer [28] models with Attention [111] offer the best performance. These enable additional contextual information to be captured among the messages. Currently, the model of Wu et al. [124] achieves the best results on the leaderboard[2] of the MIND [126] dataset. It uses a transformer structure with additive attention.

Other models like NRMS [120] use encoders for news and users. Pre-trained word embeddings are used as the basis for the news encoder. Only the title of a news item is used as information. They use multi-head self-attention and additive word attention to store special aspects in the representations. Candidates are then compared with the user representation. Unlike NRMS, NAML [118] also uses information from the category and body of the article. These are also initialized with a word or category embedding. A CNN filter is applied to the embeddings on the next layer and again provided with an attention layer. A Dense layer is used for the categories. The result is a weighted sum of the representations. The user is represented analogously to NRMS. For this, the previously clicked news is used as the basis for the user representation.

LSTUR [9] uses the classic RNN structure with long-short-term memory GRU [23] for the user encoder. The user history serves as input and is represented in this way. The individual news encodings are concatenations of topic, sub-topic, and news title embeddings. The latter are learned using word embeddings and reinforced by word attention.

NPA [119] uses personalised attention for user and news encoding. In contrast to non-personalized attention, attention weighting depends on the user's interests. The rest of the architecture is similar to NAML using word embedding and a CNN layer.

DNK [113] offers an approach via knowledge graphs. Entities are used that are linked to each other. These are embedded and filtered with a CNN layer.

Most models are based on deep architectures. This can be seen in the overviews of Raza and Ding [96] and Amir et al. [8]. The work of Raza and Ding [96] is dedicated to the broad

[2] see `https://msnews.github.io/`

picture of NR, listing current challenges, metrics used, and datasets. Amir et al. [8] refers only to the deep News Recommendation Systems. Most investigated models are time-aware, use the news content for predictions, and have a CNN or RNN architecture. They show that the research on offline systems is clearly in the foreground. Only one cited paper on online systems by Wu et al. [123] is proposed in their survey. This reinforces the work of Wu et al. [126]. The MIND dataset provides the largest free dataset with implicit feedback to date. In their paper, they present the most important statistics and show the performance of current models on the data.

Other architectures address specific issues and adapt their architecture accordingly. Since many polarising topics are widely read, some providers take advantage of this. They create fake news. Patankar et al. [91] try to counteract this problem with a bias-aware model.

3 Foundations

This chapter serves as the basis for the entire thesis. We start with the definition of a recommendation system and continue with basic to state-of-the-art machine learning methods for recommendation tasks. The second part presents a definition of bias and a general overview in NRS.

3.1 News Recommendation

3.1.1 Recommendation Systems

A *recommendation* or *recommender system* is an (automatic) tool that is used to provide items to users. This task is mostly bound to a specific field, like news recommendations or product suggestions. The major goal of a good recommendation system is to recognize the interests of a user and match these interests to suitable items.
Formally, the recommendation task is to learn a function

$$f : \mathbf{U} \times \mathbf{I} \longrightarrow \mathbf{R}, \tag{3.1}$$

where $\mathbf{U}$ is the set of users, $\mathbf{I}$ is the set of items and $\mathbf{R}$ is the feedback of a user. In practice, users and items are represented by their characteristics. For users, this can be previous preferences, personal attributes, or access timestamps. Items can be, for instance, represented with textual information or timestamps[21].
There are two different types of feedback. Rating-based feedback is called *explicit* feedback. It lets users the opportunity to provide a rating of an item. In this case, the recommendation system uses this feedback to predict items for the user. An example of explicit feedback is the five-stars rating system by Amazon[1]. This means $\mathbf{R} = \{1, 2, 3, 4, 5\}$.
An *implicit* feedback is only bound to two classes (i.e. $\mathbf{R} = \{0, 1\}$). It reflects the user's disliking (0) or liking (1) for an item. Examples of implicit feedback are purchases on shopping sites, website visits in search engines, or clicks on news articles.
Even if there are theoretically only these two types of feedback, recommendation systems may use both. For instance, Amazon can predict user interests by rating and purchasing behaviors. This is sometimes referred to as *hybrid* feedback [6].

[1] www.amazon.com

3.1.2 Machine Learning in Recommendation Systems

There are a lot of approaches to learning good recommendation systems. For different subfields, different approaches may apply. This chapter will introduce these approaches in a top-down manner, starting with the recommendation system approaches and ending with the machine learning foundations used. Before getting into touch with the different architectures, Westart with an objective that allows finding a *good* recommendation system for arbitrary uses.

Overall, a (parameterized) recommendation system f_W is called *optimal* in its parameters W if it minimizes the *true risk*:

$$\min_W L(f_W) = \min_W \mathbb{E}_{(u,i,r)\sim p*}[\mathrm{err}(f(u,i),r)], \tag{3.2}$$

where $(u,i,r) \sim p^*(u,i,r)$ denotes that u,i, and r are sampled from the underlying data distribution and $\mathrm{err}(f(u,i),r)$ is a metric measuring the distance between the actual rating of r and the predicted rating $f(u,i)$.

Since there is no real way to learn from the ideal underlying distribution, we instead use the *empirical risk*, which estimates the true risk using a training set D:

$$\hat{L}(f_W) = \frac{1}{|D|} \sum_{(u,i,r)\in D} \mathrm{err}(f(u,i),r) \tag{3.3}$$

This estimate is unbiased for an i.i.d. and large enough training set (i.e. $\mathbb{E}[\hat{L}(f_W)] = L(f_W)$) [21, 46].

In the upcoming section, we follow the structure of the book *Recommender Systems: An Introduction* by Jannach et al. [53] to give an overview of the different architectures in recommendation systems.

3.1.2.1 Collaborative Recommendation Systems

The *collaborative* recommendation system is characterized by its usage of behavioral data. On the one hand, it uses data collected from previous interactions with the system. On the other hand, it uses data from users with similar interests. The main goal of a collaborative recommender system is to represent the similarity of user interests in a good way.

The result of a collaborative recommender system can be item suggestions with a certain probability that the user likes it or the *top-K* most promising items.

Collaborative Filtering

An example of this technique is *collaborative filtering.* Consider only two users, user A and user B. User A has been given implicit feedback (like clicking or purchasing) to the items

$\{i_1, i_2, i_3\}$. User B has been given implicit feedback to $\{i_1, i_2\}$. The recommendation system now tries to give a good recommendation to user B. Since user A and user B have a high overlap ($\{i_1, i_2\}$) in their items of interest, it is convenient that the recommendation system suggests $i3$ to user B.

So the main idea is to filter the most promising items for multiple users with high conformity in their interests. A traditional approach for finding a similar user is to use the unsupervised *user-based K-nearest-neighbors* classifier. It treats the given feedback for each item as a vector in a $|\mathbb{I}|$-dimensional space. After quantifying the users' proximity, the classifier chooses a prediction close to the most similar user. This approach is a simple introduction to this type of method. However, most feedback data is too big to be compared in a matrix with the shape $|\mathbb{U}| \times |\mathbb{I}|$. A similar approach is *item-based* filtering. Instead of using user vectors to identify a similarity, it uses an item vector with different user feedback.

These two classifiers are categorized as *memory-based* classifiers. This is because they directly use the data saved in the memory to predict feedback. Another classifier is the *model-based* classifier. Instead of using the whole data, it is trained before usage. This is usually done in a supervised manner. The classifier then predicts with the learned model. Traditional model-based approaches are *Matrix factorization/latent vector models* which we want to discuss before getting in touch with more state-of-the-art approaches. Note that these approaches are used in collaborative recommendations and all types of recommendation systems (e.g., content-based and knowledge-based).

Matrix Factorization/Latent Vector Models

The term of *matrix factorization* encodes a bunch of techniques to create a latent space of vectors for the original input. The main goal is to focus on the most important features of the data. Thus the latent space has (primarily) less dimension than the original space. The reason is that strongly correlated attributes are combined in a sense. Similar to collaborative filtering, recommendations are made by comparing the similarity of an item to a user vector. In these approaches, the user vector is generated by some matrix factorization.

There are many traditional techniques to calculate a latent space of essential factors like *single value decomposition* [27] or *principal component analysis (PCA)* [2]. Both techniques are already known for a long time in mathematics and create a lower dimensional space of factors. With the rise of neural networks, a more modern solution for the factorization problem is using an *autoencoder.* Autoencoder started as a tool to apply PCA [60]. Now there are powerful tools in which autoencoders play a significant role, like *variational autoencoders* [58].

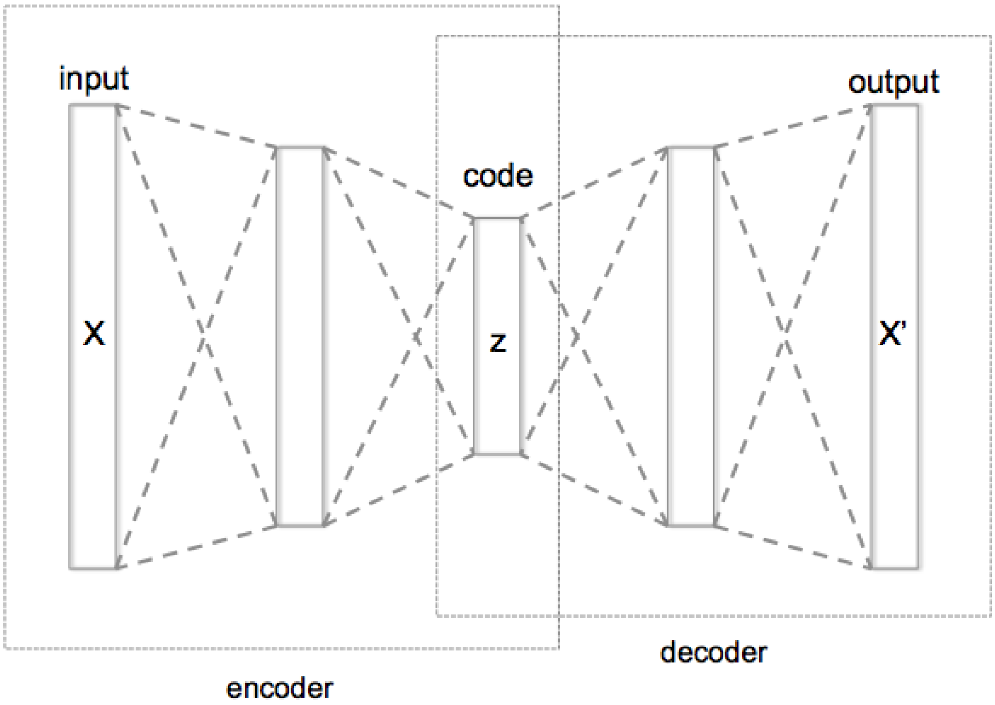


Figure 3.1: A multi-layer autoencoder (Image by Chervinskiwe- Own work, CC BY-SA 4.0, https://commons.wikimedia.org/w/index.php?curid=45555552)

Autoencoder

An *autoencoder* is a specific type of neural network (see Figure 3.1). As mentioned in the previous section, its purpose is to learn an input's hidden/latent representation (referred to as *code*). It is an unsupervised approach thus, it needs no labels for ground truth information. However, it is trained like a supervised neural network with labels equal to the input. Thus it is also referred to as a *self-supervised* model. The architecture has two basic components - the encoder and the decoder. The encoder is a function $e : \mathbf{X} \rightarrow \mathbf{Z}$, where $\mathbf{X}$ is the input space and $\mathbf{Z}$ is the latent space. Note that the size of the latent space can be chosen and is usually set $|\mathbf{Z}| \leq |\mathbf{X}|$ (called *bottleneck*). The decoder is a function $d : \mathbf{Z} \rightarrow \mathbf{X}'$, where $\mathbf{X}'$ is the reconstruction of $\mathbf{X}$. The simplest version of an autoencoder uses only one layer (*shallow*), but there can be advantages like exponentially less needed training data and less computational afford for deeper autoencoders [36]. The objective to train an autoencoder model is

$$\min_{U,W} L(d_W(e_U(x)), x),$$

where U and W are the parameters of the encoder/decoder, and L is a loss function to quantify the distance between the prediction and the actual input.
Usually, the loss function is some function like the (empirically) *least-squares* $L(d_W(e_U(x)), x) = \frac{1}{n}\sum_{i=1}^{n} ||(x_i - d_W(e_U(x_i)))||_2^2$ for $x_1, .., x_n \in X$ and the *steepest descent* optimizer.
There are more collaborative recommendation approaches like the *probabilistic* recommendation system.
We continue with the content-based models.

3.1.3 Content-based Recommendation Systems

Even if collaborative filtering is reasonable, it lacks semantic information. It only uses the behavioral information of the community. When there is no community or only a small one, the model is not able to infer good recommendations.
Thus *content-based recommendation systems* are built on the contextual information of an item. It is convenient to use attributes like a news title in news recommendations or a book genre in book recommendations. The main idea is to use this information, encode them in some sense and match user profiles to new items. In the upcoming section, we will focus on the problem of content representations and especially on text/word representations.

Content Representation

In most cases, attributes about the recommended items are saved. These attributes of an item are referred to as *features.* Considering news recommendations, this could be a table with information about the category, the keywords, or the author of a news article. With this information or features, it is possible tocreate a preference profile for each user and match interesting items.
The basic idea for such a profile is saving the information to another table. Each feature then contains a list of liked values. An item will be recommended if the feature values have a significant overlap. However, this brings problems, especially with textual features like keywords, genres, or titles. For instance, each word would be equally important. Another problem is ambiguities and synonyms in words. To ensure a better representation of textual features, we want to present some options for this task. To do so, we will come back to the example of keywords in a news article.
Usually, we want to have numerical and not textual data. But most of the contextual data is in text form. To get a numerical representation of a text, we may introduce methods to create a vector representation of a word. There is a basic approach called *one-hot encoding.* A numerical text representation yields an n-dimensional vector where n is the number of all possible words. Thus each dimension represents a word. If the word is at this dimension, it has a 1(*hot*) as value and otherwise a 0. The representation gives us a way to calculate with

words, like representing a sentence with the sum of its one-hot encoded words. Note that these vectors are very sparse, and no contextual information is stored. A variation of this representation is the *TF-IDF* encoding [24]. Instead of using only boolean values on the encoding, it scales them according to the word's relevance. The relevance (and therefore the scaling) is calculated with the *term frequency* and the *inverse document frequency.* The term frequency is the ratio of term (word) occurrences to a document's total number of words. A higher term frequency means a higher relevance. The inverse document frequency is the inverted number of term occurrences in all documents. Note that it is counted only once per document. A higher document frequency means lower relevance. Thus it gets inverted. Overall, to create a better encoding, there are multiple pre-processing steps to consider, like adding *stop words*, *stemming*, *lemmatization* and *normalization.*

In the following section, we will mostly follow the structure of the book *Representation Learning for Natural Language Processing* [70] and *Hands-on Question Answering Systems with BERT: Applications in Neural Networks and Natural Language Processing* [99].

3.1.3.1 Distributed Word Representation

There are more advanced encoding options to catch the syntax of a sentence and the relatedness between words. *Word2Vec* [80] and *GloVe* [92] create word *embeddings* that are used widely in NLP tasks, including recommendation systems with content-based information. They are powerful because similar/related words are close to each other in the embedding space. This means a recommendation model using word embedding may be able to suggest titles similar to the title the user liked earlier.

These types of methods are called *distributed word representations.*

Word2Vec

Word2Vec [80] is a tool introduced by Google. There are two different models to create an embedding for each word in a dictionary, continuous-bag-of-words and skip-grams (see Figure 3.2).

Continous-bag-of-words (short: CBOW) embeddings are trained on the task to predict a center word in a surrounded context called context window. This ensures that embeddings learn a similarity between words. Mathematically, CBOW does predict the probability

$$P(w_i|w_{j:|i-j|\leq l, i\neq j}) = \text{softmax}(\mathbf{M}(\sum_{j:|i-j|\leq l, i\neq j} w_j)), \tag{3.4}$$

where w_k is the kth word, l is the context window size which denotes the number of words left and right to the center word, and $\mathbf{M}$ is the weight matrix.

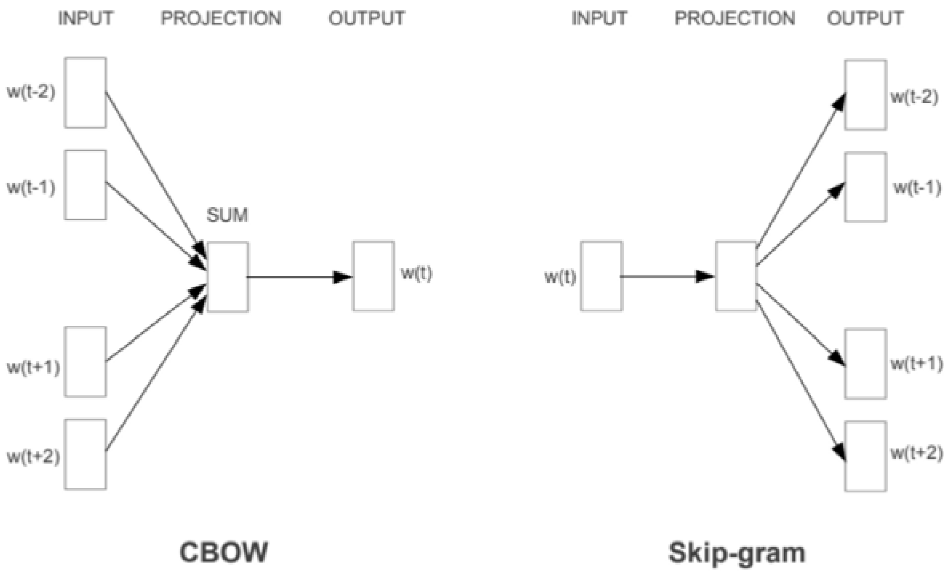


Figure 3.2: A comparison between CBOW and Skip-Gram representation (image taken from Mikolov et al. [81]

Example: "Alice$_1$ has$_2$ a$_3$ big$_4$ __$_5$ with$_6$ a$_7$ huge$_8$ garden$_9$"

With $l = 2$ and $i = 5$ the model would try to predict $P(w_i|$"a", "big", "with", "a").

Attention Models

There is still a major shortcoming with distributed word representation: they are context-independent. This means that sequential inputs are not seen as a big picture but as little independent pieces.

In 2014, Bahdanau et al. [12] redesigned traditional Sequence-to-Sequence learning with RNN-based encoder-decoder architecture via context-based weighting, namely attention. The output can relate to the importance of specific inputs. In the original paper, a context vector c_i is calculated for each target y_i for a sequence $\{x_1, ..., x_n\}$. Each c_i is then computed via

$$c_i = \sum_{k=1}^{T} \alpha_{i,k} h_k, \tag{3.5}$$

where $h_j \in \{h_1, ..., h_n\}$ are so-called annotations, being the output of the encoder for an input $x = \{x_1, ..., x_n\}$ of an RNN and $\alpha_{i,k}$ the attention weights. So the encoded input is weighted by attention.

The attention weights must be trained in the process and should be probabilities. Since the relation is initially unknown, some feedforward network f is trained to find optimal $\alpha_{i,k}$.

This leads to the formula

$$\alpha_{i,k} = softmax(e_{i,k}) = \frac{exp(e_{i,k})}{\sum_{j=1}^{T} exp(e_{i,j})} \tag{3.6}$$

where

$$e_{i,k} = f(s_{i-1}, h_k).$$

The function f is called the alignment model and is trained to learn the relation between the last hidden state (decoder output) of a RNN s_{i-1} and the annotation h_k. With this probabilistic modeling, a context vector c_i is the expected annotation with a certainty of $\alpha_{i,k}$ [12]. The original paper is limited to RNN/BiRNN and sequential data in general. Later, a general approach to attention was proposed with transformer attention [111]. Also, there are different alignment score functions [111, 72].

3.2 Biases in Recommendation Systems

3.2.1 Statistical Bias

Although there are many different types of biases, they all refer to one general definition:

> *A (statistical) bias is a systematic tendency towards or against someone or something, which causes a gap between statistical results and the truth.*

Mathematically, a bias is a difference between your expected estimate and the truth:

$$\text{bias}(\hat{\theta}) = \mathbb{E}(\hat{\theta}) - \theta, \tag{3.7}$$

where $\hat{\theta}$ is the estimator and θ is the true value. A (point) estimator/statistic is a function that predicts a certain quantity of interest. This can be a descriptive statistical value, a vector of weights for a machine learning model, or a function/function estimator. Therefore, an estimator takes n independent and identically distributed data points $x_1, x_2, ..., x_n$ and estimates

$$\hat{\theta} = f(x_1, ..., x_n). \tag{3.8}$$

It is to be observed that an estimator with $\text{bias}(\hat{\theta}) = 0$ is called *unbiased* estimator [36].

3.2.1.1 Cognitive Bias

In psychology, the term *cognitive bias* indicates a misinterpretation of human perceptions that causes distorted conclusions. Often, this refers to self-delusion in the process of thoughts [114]. Since recommendation systems interact with users, many cognitive biases are present in the collected behavioral data. There is a large number of psychological effects causing biases. John Manoogian designed an overview (see Appendix Figure A.1). In this overview, the different effects are categorized into four different groups. The first group lists every effect related to the human memory process. For instance, the *negativity bias* is caused by the effect that we remember bad events more intensely than good things. In rating-based (explicit-feedback) recommendation systems, this may overall generate a more negative rating.

The second group is all effects related to the tendency to act fast. An example of these effects is the *less-is-better effect*. It states that low-value options are chosen more often than high-value options [52]. The other two groups are the effects of an overdose of information and, on the other hand, not enough meaningful information. An easily understandable example of an overdose of information is that flashy and bizarre information shown is better memorable than common information. This effect is noted as *bizarreness-effect* [11].

However, when the human brain is fed less information, it tends to fill in meaning with prior knowledge. For instance, we tend to believe people with high authority more than people with less authority (*authority bias*) [83]. Note that each effect may be part of multiple groups. Also, definitions of effects and biases can overlap. There are more than this categorization options, like dividing the biases into conscious and unconscious biases.

3.2.1.2 Inductive Bias

Another term of bias is introduced when speaking about machine learning, the *inductive bias*. An inductive bias is caused by the assumptions made when choosing a model. Most modern algorithms, especially neural networks, have to make some assumptions when learning to generalize an unknown function and distribution of data. These assumptions might be the type of neural network architecture or the number of parameters needed to fit the data.

It shows that not every bias is harmful and needs to take care of [84].

Finding biases and debiasing is a big topic in recommendation systems. There are many different types of bias. However, not every bias is as relevant as others in news recommendations.

This section will give an overview of the different biases. Firstly, it is shown more generally on recommendation systems and in the particular case of news recommendations. We will follow the survey paper by Chen et al. [21] and Mehrabi et al. [79] in terms of terminology and definitions. In general, there are three stages in which biases can occur. The first stage is the **collection** of user data. On the recommendation task, these data represent the user behaviors on the system, like click histories or given ratings. The second stage is the process of **learning** a recommendation model with the data collected. And the final step is to **serve** new recommendations to the user by the trained model. This sequence of steps is executed repeatedly. This is called the **Feedback loop** (see the left side of Figure 3.3).

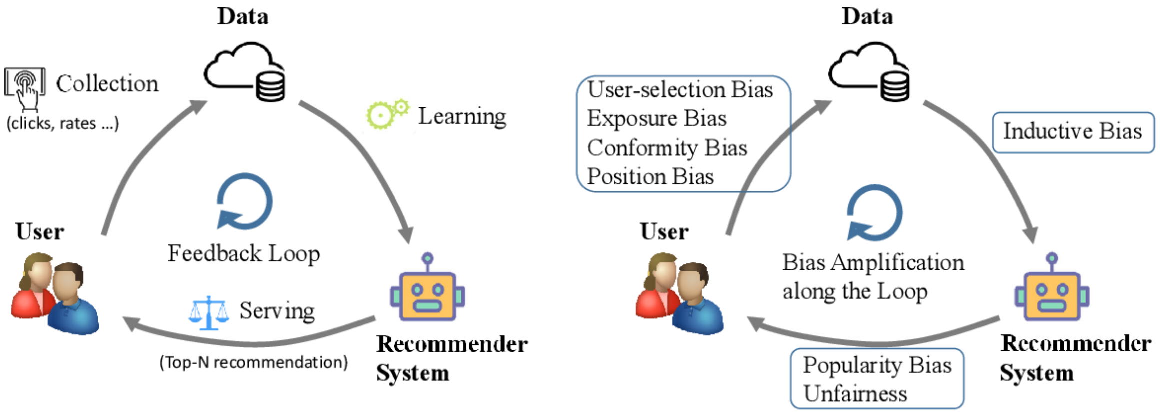


Figure 3.3: Stages of the recommendation task (left) and according to biases (right). Taken from Chen et al. [21]

3.2.2 User-to-Data Bias

The User-to-Data biases are closest to the cognitive biases mentioned before. Users interact with the recommender system and then choose/click/rate an item. This item is then added to the learning data.

3.2.2.1 Selection Bias

Generally, the *selection bias* occurs when the data is not sampled randomly. In the environment of a user making explicit feedback, this feedback might not be random. This happens since users are willing to rate an item they particularly like or dislike [78, 21].
Another problem is that a recommender system only shows a specific number of items, even if there are more recommended items. It forces the user to choose an item that prevents a random selection. Observe that this bias is related to a positional bias [90].

3.2.2.2 Position Bias

When showing items to the user, they can not be presented equally. However, items in a higher position are more likely to be clicked by a user. It causes a *positional bias* since the position is not generally related to the relevance of an item [26].

3.2.2.3 Context Bias

Recently, the term *context bias* was introduced by Zheng et al. [129] It describes the contextual relation between multiple items and their attributes. An attribute might be the

position, the modality, or the title of the item.
As an example, let us consider two news items. The first one is an article positioned higher than the second one. However, the second one is a video with a flashy title. In this context, it is hard to imagine which news is more likely to be clicked, even with the knowledge of a potential positional bias and modality bias. This is what context bias captures.

3.2.2.4 Exposure Bias

The *exposure bias* results from ambiguity between unobserved and unliked items by the user. Considering the case of implicit feedback, the recommendation system cannot distinguish between an unclicked and a not seen item. Again, this is related to a positional bias since lower positions are more likely to be unseen.
Exposure bias is also called *previous model bias* in some literature [68, 67]. This is because the last recommendation models and their policies generate the exposed items.

3.2.2.5 Conformity Bias

Conformity bias is caused by the influence of society and groups on the individual. Therefore, users may not interact with an item due to their preferences but to a conformity propensity. [73]
Conformity bias is more relevant in explicit feedback systems since visible feedback from other users affects user opinions.
It is also referred to as social bias in research. [79]

3.2.2.6 Behavioral Bias

The *behavioral bias* generally describes the effect that people tend to act differently in different situations. For instance, writing an e-mail to a friend might be less formal than an e-mail to an employer. [88]

3.2.2.7 Temporal Bias

A *temporal bias* is the tendency of changes in behaviors over time. Since society is constantly changing and systems may change, the usage of a system is affected by it. This causes a trend of worse generalization when the system is not updated. [88]

3.2.2.8 Anchoring Effect

The *anchoring effect* is a cognitive bias due to an influential reference point called *anchor*. People tend to support their thoughts to this point even if there is no relation.
A referenced anchor can be numerical and non-numerical. [128]

A typical example of the anchoring effect comes from the manipulation of customers in marketing. Products are marked with an old price and a new, reduced price (e.g. 6.99€ to 2.99€). The old price is an anchor which is supposed to suggest a good deal on the product. Customers are more likely to buy this product because of the old price.

3.2.2.9 Framing Effect

When context, like a piece of information, is proposed in a certain way, it is called a *frame*. The *framing effect* is the shift of the perceived impact of information according to the frame. There are two types of framing, loss and gain framing. The loss frame is a view from the bad side of information, whereas the gain frame is viewed from the good side of information. [109]
The framing effect and the non-numerical anchoring effect are close. This is because a frame can be viewed as an anchor.
For instance, there was a train accident with 1000 people and four injured people due to a crash. A newsletter is writing an article about that crash. A loss framing for the title of this article would be "Four people got injured at a train accident," whereas "Almost no injuries at a train accident" would be a gain framing.

3.2.2.10 Weber-Fechner Law

The *Weber-Fechner law* describes the relative sensual perception of humans. It states that the human perception of an increasing number of visual impressions is different. Linear growth of visual impressions is received as a logarithmic increase instead [97].

3.2.2.11 Semmelweis Reflex and Conservatism

The *Semmelweis reflex* describes the effect that new information is more likely rejected if they conflict with old information, norms, or paradigms. It is a cognitive bias produced by the overestimation of the own beliefs. [51] Therefore it is closely related to *conservatism* and the *egocentric bias* [102]. The term "reflex" comes from the impulse to directly disagree on this conflicting new information.
A famous example is Galileo Galilei proposed his controversial observations to the church, which did not believe his studies that the earth was not the center of the universe.

3.2.2.12 Attributional Bias

An *attributional bias* is the systematic fallacy when evaluating the behaviors of oneself or others. There are some well-known subgroups of attributional biases.

Group Attribution Error/Stereotyping is the fallacy that a certain group has the same attribution or opinion. On the other hand, it is a fallacy that a certain group member is representative of the whole group in terms of attribution. [74]
Ultimate attribution error is the tendency to underestimate situational and overestimate personal factors in terms of someone's behaviors outside of a group. This lead to an over-negative view of a person outside a group and an over-positive view of a person inside a group. For example, this group can be gender, race, or profession. [49]

3.2.2.13 Automation Bias

An *automation bias* is the tendency of humans to underestimate their beliefs and decision power and overestimate the correctness of decisions made by machines. This causes humans to prefer a decision by a device to a decision made by themselves. [35]

3.2.2.14 Reactive Devaluation

Reactive devaluation is the tendency to devalue a recommendation made by someone or something that the person dislikes or that is an antagonist.
An example was proposed by Ross and Stillinger [98] in 1991. They asked pedestrians from the US if they would agree that a general bilateral reduction of nuclear arms would be desirable for Russia and the US. 90% agreed when the questioner added the information that the proposal came from the president of the US. 80% agreed when the questioner added the information that the proposal came from a group of experts. And only 44% agreed when the questioner added the information that the proposal came from Michael Gorbachev, the former leader of the Soviet Union.

3.2.2.15 Historical Bias

Finally, a bias is referred to as *historical bias.* It describes an intrinsic bias due *to the world as it is.* To evaluate a historical bias, it is required to have a retrospective on the world and how it was and now is.
As an example: According to Player et al. [93] there are only 5% of women in the top 500 CEOs of the world in 2018. In 2018, searching for *CEO* would have led to only a fraction of women shown. This is because there is a historical bias against women. However, this effect enhances the thinking that it is more likely that a man can be a CEO than a woman. Later, Google changed the number of images shown to be fairer. [108]

3.2.3 Data-Model Bias

In this section, we present the biases in picking up data for the algorithmic part of the model.

3.2.3.1 Measurement Bias

Measurement biases are a group of biases. They occur in different types of measurements mostly due to wrong choice or usage of measurements. Suresh and Guttag [108] stated three measurement problems that result in biased data.

1. **Different groups in data are measured in a different frequency**. For example, let us say one group of participants in a study is much more often monitored. This led to an increasing failure in this group. In a feedback loop, this leads to even more monitoring. [14]
2. **Different groups may have a different quality in their measurement**. This is often the effect of systematic discrimination.
3. **Labels and features may be designed in an oversimplified way**. This led to an inaccurate and biased view.

In the often criticized *COMPAS* model [108, 79, 57] for classifying criminality, the "riskiness" attribute has a significant measurement bias.
In *COMPAS*, the riskiness of a person is measured by the times this person or their close contacts has been arrested. This is a biased measurement since afro-Americans are discriminated against in a way that they get arrested much more frequently.

3.2.3.2 Omitted Variable Bias

An *omitted variable bias* occurs when the used model omits certain important variables for inferring a prediction. This can be featured in data sets that ignored or did not consider external factors.
Internal omitted variable: An image classifier is limited to a certain input size. Therefore images have to be compressed. This leads to an information loss. This is closely related to an inductive bias.
External omitted variable: A classifier tries to predict a person's pregnancy probability. However, the training data set has no attribute "gender".

3.2.3.3 Aggregation Bias

When making a wrong conclusion about an individual seeing the whole population, it is called an *aggregation bias*. This may occur when the model has not connected certain conditions to make a prediction.

In medical tools, there are shown that some diseases have different types of symptoms in different types of subgroups like gender. Therefore it might be hard for a single model to predict a disease for the whole population but for a specific subgroup. [108, 107]
There are multiple subcategories of the aggregation bias like the *Simpson-paradox* [104] and the *Modifiable Areal Unit Problem (MAUP).*
When a total random sample of some population is divided into sub-sample and the sub-samples unexpectedly tend to be correlated, it is called Simpson-paradox [117].
MAUP is a bias when aggregating point data with different boundaries, which then leads to a different perception and inference [34] (see Figure 3.4).

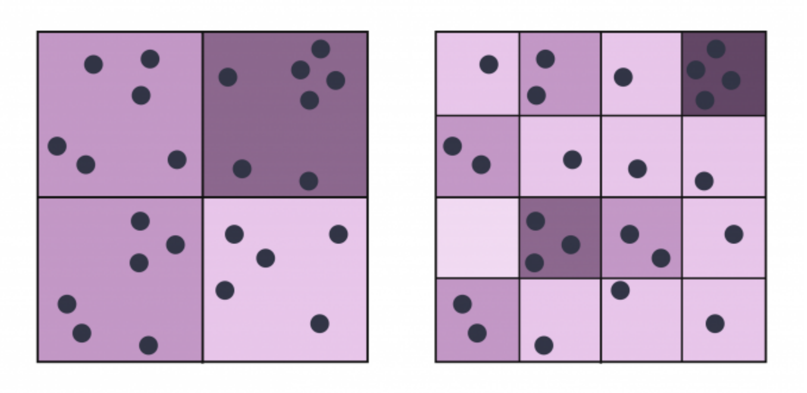


Figure 3.4: Modifiable Areal Unit Problem. Scaling of boundaries changes the perception of the dots (from `https://gisgeography.com/maup-modifiable-areal-unit-problem`).

3.2.3.4 Sampling/Representation Bias

A *sampling bias* is the problem of non-random sampling inside subgroups. This leads to an unrepresentative population and worse generalization when dealing with new data.

3.2.3.5 Longitudinal Data Fallacy

The *Longitudinal data fallacy* is a temporal, statistical bias that occurs when the data is collected as cross-sectional. Still, predictions are made like it is a longitudinal study. This is often because longitudinal studies are much more time intensive. Fallacies happen because the same cohorts tend to be biased in some sense, but cohorts from different points in time are unbiased. A study cross-sectional study has only a single point of time where the data is collected. In contrast, a longitudinal study has multiple points where the data is collected. Barbosa et al. [13] proposed an example of user behavior data on Reddit[2]. They hypothesized that the length of comments on Reddit would decrease over time using a cross-

[2] www.reddit.com

sectional study. The study strengthened their belief. However, using the actual longitudinal data shows a different trend.

3.2.3.6 Linking Bias

The *linking bias* is fixed to network attributes. It occurs when connections or node attributes are misleading and not representative.

3.2.3.7 Redundancy Ambiguity

Redundant data is identical or close to similar data in the overall data set. Usually, redundant data will be removed in the data cleaning process, but it is not always desired. Consider the example of someone making a post on Facebook multiple times. If it was posted accidentally, the redundant ones should be dropped from the data set. When someone wants to state something numerous times, this information would be more important to them, and the data should be kept. This lead to ambiguity and a potential bias on the importance of information.

3.2.4 Model/Learning Bias

The second stage where biases occur is the learning stage. In this stage, the model makes assumptions about the data, which causes an *inductive bias* mentioned before (see subsubsection 3.2.1.2).

3.2.5 Resulting/Serving Bias

In the final step, the model produces recommendations which are the model's results. These results are served to the users, which then completes the circle of the feedback loop. This introduces biases in the served recommendations.

3.2.5.1 Popularity Bias

Even though *popularity bias* is mentioned in the section of serving biases, it is a problem that originated in data and in the algorithm used to train the model. In recommendation systems, data is often long-tailed (e.g., see) in terms of the population of items. This means a few things appear as recommendations very often for the users, and the majority of items are not as popular and appear lesser [4].

Another prevalent example of this phenomenon is that the 10% of the wealthiest people on earth own about 85% of the global household wealth. This results in another common fact that the richer people get even richer. [110] The same applies to the recommendation task: The most popular recommendations are recommended even more in the feedback loop, which is empirically proven by Abdollahpouri et al. [5].

3.2.5.2 Stance/Sentiment Bias

The *sentiment bias* is a specific bias in the field of NLP and news recommendation. It is the tendency for recommendation models, including language models, to have a more negative sentiment. Similarly, the *stance bias* indicates a tendency to certain viewpoints. Both effects lead to reduced diversity in the proposed information. [7]

3.2.5.3 Cold Start Problem

A well-known issue in recommendation tasks is the *cold-start problem.* The problem occurs with new users. The user has no interaction/rating behavior when first interacting with the system. The underlying model cannot match promising items for the user, and the prediction is more likely to be unsuited. Overall, these new users may get worse recommendations. This is a problem in collaborative filtering and content-based recommendations. [42, 112]

3.2.5.4 Unfairness

The last thing to mention is *unfairness.* It might be the most well-known bias. It describes systematic discrimination against someone or something. In recommendation systems, this might be users with certain characteristics like gender or race or news coming from a particular website.

3.2.6 Feedback Loop/Bias Amplification

When information is first proposed to users, they interact with the system and pick items from impressions they like. The system then picks this information up and improves its recommendation on the following proposed items. This is called a *feedback loop* in recommendation tasks (see Figure 3.3). This feedback loop produces the problem of *bias amplification.* It means that certain biases getting more present when the feedback loop is repeated over and over again.

For instance, Mansoury et al. show in their paper on bias amplification that there is an amplified popularity bias and a trend of reduced diversity in each feedback loop. [76]

3.2.6.1 Emergent Bias

A big problem seen in examples like Microsoft chatbot *Tay* [55] is the *Emergent bias.* It is caused by the biased society interacting with the model repeatedly. *Tay* was a chatbot launched in 2016 on Twitter[3]. After only 24 hours of interactions with the community, it got racist and posted conspiracy theories and other delicate topics. Microsoft then decided to shut it down.

This was due to the emergent bias caused by interacting with groups of people sharing these opinions. The tendency is limited to models with feedback loops. It creates a window for attacks that contaminate the model.

There are two types of emergent bias for different kinds of models. An online model that still learns about the process of new interactions might learn this biased information. For an offline model exclusively trained on collected data, there is also a problem of emergent bias. *Tay* searched the internet for information when it interacted with people. For specific suggestive questions, it replicated opinions from the searched website. This caused a bias without actually learning what it was writing. [55]

[3] Social media platform : www.Twitter.com

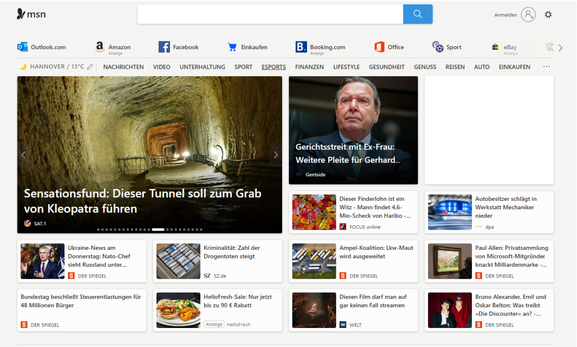
4 Experimental Setup

In this chapter, we want to talk about the experimental setup. There are some limitations in the dataset, used models, and bias analysis for this work.

4.1 The MIND-Dataset

In this work, we focus on *MIND*, which was proposed by the team of Microsoft and Tsinghua University. MIND is an acronym for **Mi**crosoft **N**ews **D**ataset. It is an aggregation of user click behaviors on the Microsoft News platform[1]. An example of a user impression is shown in Figure 4.1.

Wu et al. [126] have collected data for 1 million users on the platform, randomly sampled from the total population. These users have a history of at least 5 click interactions between the 12. October and the 22. November 2019. The collected behaviors are due to implicit feedback and therefore do not have a form of rating across the news. The first 4 weeks of collecting data are used as the training data. The fifth week is for the validation set and the last is for the test set. There are three different sizes made available for the dataset. The *MIND-large* is the original dataset, containing all 1 million users and their behaviors. For test purposes, the *MIND-small* and *MIND-demo* were proposed.



Sensationsfund: Dieser Tunnel soll zum Grab von Kleopatra führen
SAT.1

Gerichtsstreit mit Ex-Frau: Weitere Pleite für Gerhard...
Gentside

Dieser Finderlohn ist ein Witz - Mann findet 4,6-Mio-Scheck von Haribo -...
FOCUS online

Autobesitzer schlägt in Werkstatt Mechaniker nieder
dpa

Ukraine-News am Donnerstag: Nato-Chef sieht Russland unter...
DER SPIEGEL

Kriminalität: Zahl der Drogentoten steigt
SZ.de

Ampel-Koalition: Lkw-Maut wird ausgeweitet
DER SPIEGEL

Paul Allen: Privatsammlung von Microsoft-Mitgründer knackt Milliardenmarke -...
DER SPIEGEL

Bundestag beschließt Steuerentlastungen für 48 Millionen Bürger
DER SPIEGEL

HelloFresh-Sale: Nur jetzt bis zu 90 € Rabatt
HelloFresh

Diesen Film darf man auf gar keinen Fall streamen
WELT

Bruno Alexander, Emil und Oskar Belton: Was treibt »Die Discounter« an? -...
DER SPIEGEL

Figure 4.1: Microsoft news homepage from 10. November 2022.

[1] https://www.msn.com/de-de

Due to a reduced computation time, the experiments are limited to **MIND-small** dataset, which contains 50,000 users and their behaviors. These users are randomly sampled from the original dataset. The data contains 156,965 impressions indicating clicked and not clicked news.

4.1.1 Structure of MIND

A news article is characterized by a *News ID*, a *Category*, *Subcategory*, *Title*, *Abstract*, *URL* and a dictionary of *Title Entities* and *Abstract Entities.* A description of the columns can be found seen in Table 4.1. The *Abstract* and *Title Entities* are referenced to the WikiData[2] knowlegde graph (see Table B.1. Note that *MIND* does not have any information about the actual body but the URL to the news article.

The rows of behavioral data are mainly attributed to an impression proposed to a user and the user's history. To indicate whether a candidate in an impression is clicked or not, the authors concatenated a 1 for a click and a 0 for no click. Additionally, the timestamp of the impression is recorded. In summary, an Impression is characterized by a unique *Impression ID*, an *User ID*, a *Time*, a *History*, and *Impressions.* Short descriptions can be viewed in Table 4.2.

Column	**Description**
News ID	Unique identifier of the news entry
Category	Main topic of the news e.g. sports
Subcategory	Sub topic of the news e.g. soccer
Title	Header shown to the user
Abstract	Small description of the news
URL	Site link to the news
Title Entities	Key words of the title
Abstract Entities	Key words of the abstract

Table 4.1: Column description of the news.

Column	**Description**
Impression ID	Unique identifier of the impression
User ID	User identified by his unique ID who the impression was shown to
Time	Timestamp when the impression was shown
History	List of news which were clicked by the user in the past
Impressions	Choice of news which were shown to the user with concatenated 0 or 1 for not clicked/clicked news

Table 4.2: Column description of the behaviors.

[2] `https://www.wikidata.org/wiki/Wikidata:Main_Page`

4.1.2 MIND Statistics

In this subsection, we want to give an overview of some descriptive statistics according to MIND news and behaviors. Since most of the tests are running on the small dataset, we will present the statistics of the training set of the small dataset. A short description of the large dataset can be viewed in Table B.2. Most of the shown distributions and statistics represent the large dataset very well.

The small MIND is only splitted into a training and a validation set. A test set is not proposed. Both, the validation and training sets contain a total of 50,000 users. The training set contains about 68% of the total behaviors and the other about 32% in the validation set.

4.1.2.1 News Statistics

The news file contains $51,282$ different news in the training set and $42,416$ in the validation set. Briefly, we present statistics on each piece of information about the news.

News Categories

Each news is categorized into one of 17[3] topics/categories manually set by the author of the news article. The absolute frequencies of each category can be viewed in Figure 4.2. The major share of news categories is *news* with about 30.76% and *sports* with about 28.3%. The next highest category in terms of frequency is the *finance* category with about 6% of share.

Each of the categories has multiple subcategories. The frequency and the top subcategories are presented in Table B.3.

News Titles

News titles are different in their syntax and semantics. However, it is hard to measure the semantics of a text. There are some ways proposed in the experimental part where we classify the sentiment of each news. It shows that there is a tendency for more negative than positive news in terms of sentiment. In the semantic view, we give an overview of the lengths of news titles.

The title lengths are in a form of a gaussian distribution (see Figure 4.3). Its values are around the mean of 66.25 according to characters and 10.77 according to words in the title. The standard deviation for characters is 19.22 and 3.29 for words. That means, about 95% of all titles have between 4 and 17 words in them and 27 to 104 characters, assuming a gaussian distribution.

[3] In the large dataset is a total of 20 categories

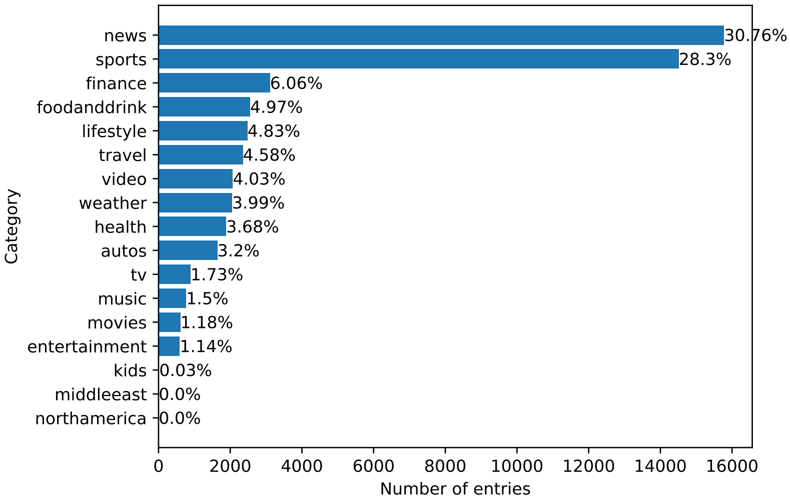


Figure 4.2: Frequencies of each category in the news dataset.

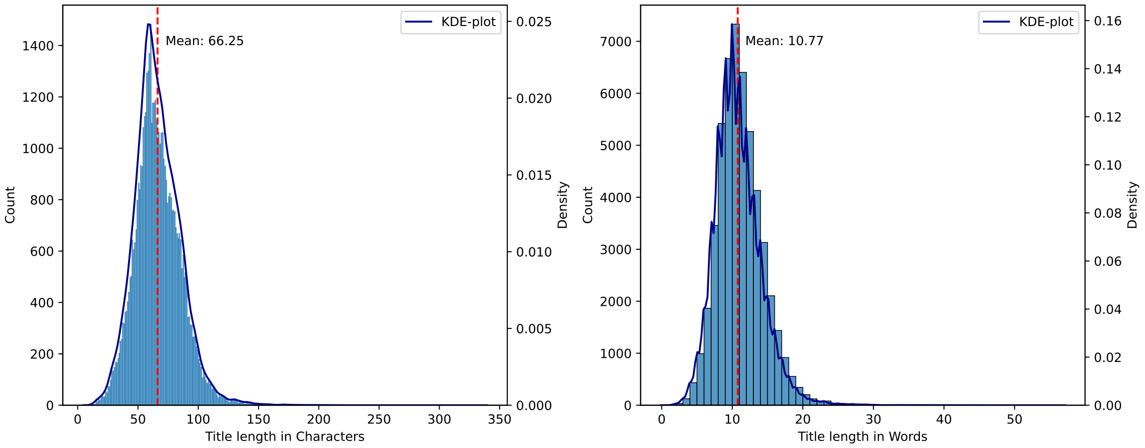


Figure 4.3: Histogram/KDE plot of each title lengths according to characters (left) and words (right).

4.1.2.2 Behaviors Statistics

In the behaviors dataset are impressions and histories of $50,000$ users randomly sampled from the total dataset. The history of a user is always the same but the impressions change. The total number of impressions in the dataset is $156,964$. Since the data is collected over a time period of five weeks, there is no necessity in investigating the time stamps. So we focus on the histories and the impressions.

Histories

A user's history is a list of clicked news articles in the past. They are represented by their *News ID*. Since there are a lot of different users in terms of their activity, the sizes of

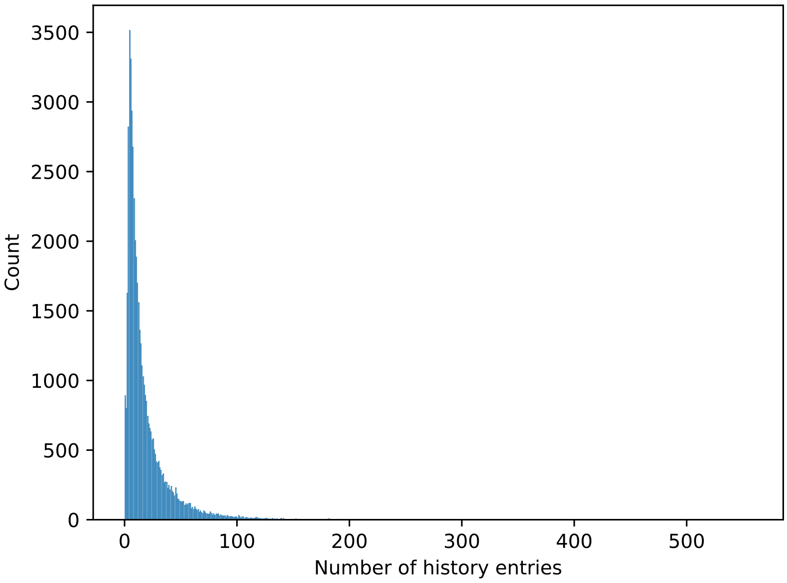


Figure 4.4: Long-tail histogram of the history length of each user.

the histories vary a lot. The histogram of the lengths is long-tailed. This can be seen in Figure 4.4. Most users have a history with four, five, or six news. The median is 11 news, meaning that 50% of all user histories have a size lower or equal to 11. 90% of users have a history size less than 42 and 99% a size less than 116. Note that the total counts sum up to $50,000$. So each history only counts once.

Impressions

Impressions are the recommendations by the RS to the user that the user interacted with. A news article represented by its ID is noted as clicked or not. A news article in the impression is called *candidate*. There is always at least one candidate clicked in an impression.
The number of candidates varies on each impression. The absolute frequencies are long-tailed, and smaller impressions occur much more than bigger impressions (see the left graph of Figure 4.5). 50% of the data has an impression size of less or equal to 24 candidates. The average click rate on an impression is about 0.109. It is measured as

$$CR(\mathbf{D}) = \frac{1}{|\mathbf{D}|}\sum_{i\in\mathbf{D}} \frac{|\{c : (c \in i) \text{ and } (\text{c is clicked})\}|}{|i|},$$

where $\mathbf{D}$ is the dataset of behaviors, $i = \{c_1, .., c_{|i|}\} \in \mathbf{D}$ are the impressions and c_j for $j \in \{1, ..., |i|\}$ is a candidate with a click-indication.
Overall, 72.56 of the impressions have only one clicked candidate. Less than 1.5% of the impressions have a higher click count than 5 candidates(see the right graph of Figure 4.5). A truncated table is presented in Figure B.1.

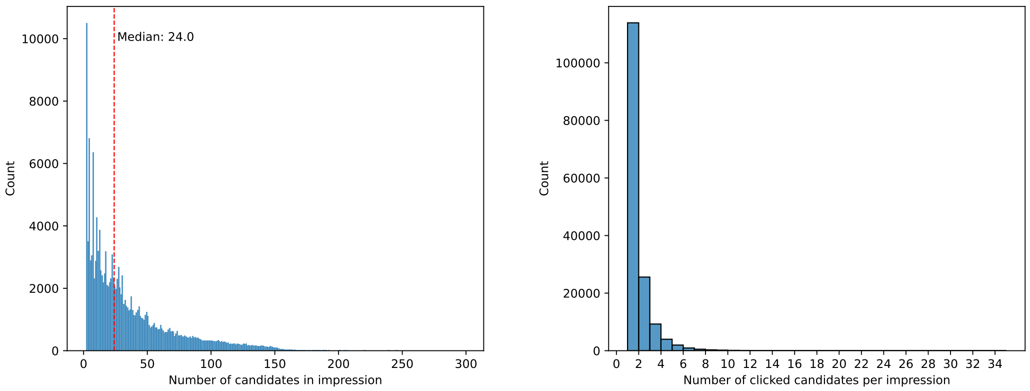


Figure 4.5: Histogram of impression sizes (left) and clicked candidates per impression(right).

4.2 Microsoft Recommenders

In this work, we use *Microsoft Recommenders* to load implementations of modern NRS. *Microsoft Recommenders* is an open-source library to get in touch with RS. The library contains five subsections of utilities: Recommender algorithms, Datasets, Evaluation, Utils and Hyperparameter tuning[4]. Recommender algorithms implement different recommendation models but are not limited to news recommendations. These contain traditional models like the TF-IDF approach and modern embedding-based and deep model approaches. The Datasets subsection proposes an interface for loading datasets like MIND or MovieLENS. The Evaluation section offers metrics and evaluation techniques to quantify the results of a recommendation model. Hyperparameter tuning gives some aid in automatic optimization in terms of hyperparameters. Utils has some valuable tools for the usage of the data. We mainly use methods in Recommender algorithms, evaluation, datasets, and evaluation. Hyperparameter tuning is not a part of this work. When using the recommendation models, we stick to the recommended hyperparameters.

4.2.1 Recommender Algorithms

The experimental part starts with an overview of performances on MIND using models from the *Recommenders library*. We will introduce the used models in this section to understand these methods. A table of used hyperparameters can be viewed in Table C.1. All model-related metrics are plotted with WandB [115].

NRMS

NRMS is an acronym and stands for *Neural News Recommendation with Multi-Head Self-Attention*. It was introduced by Wu et al. [120] in 2019 .

[4] see `https://microsoft-recommenders.readthedocs.io/en/latest/`

NRMS is a content-based recommendation model. It has a neural architecture with two encoders (see Figure 4.6) to identify contextual similarities in and between news using multi-head attention.

The *news encoder* is producing word embeddings e_i for each word in a news title. It then uses a multi-head self-attention layer to learn contextual connections of the word. The output is denoted as h_i^w and concatenates all heads. Afterward, another layer of additive word attention is used to select the most important words in a news title. The words are representations learned in the second layer. The output of the news encoder is a weighted sum of these representations.

The user encoder acts analogously. Users are represented as the history of their clicked news titles. The news encoder embeds each news title in the history. Again, the embedded news is fed into a multi-head attention layer to learn contextual relatedness between this news. Afterward, the additive news attention weights news in history by importance. The dot product of embedded candidate news and the user representation measures the prediction of a news click.

The main tests in this work are exemplified with NRMS. This is because it has an easy-to-understand structure and yet high performance. For instance, in the paper by [121], it can outperform other models like DNK, NAML and NPA.

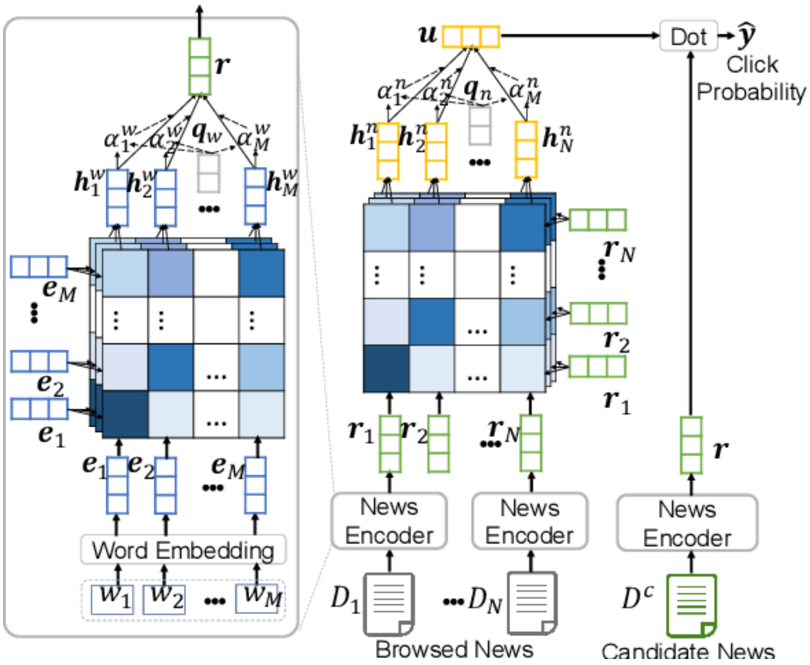


Figure 4.6: Framework of NRMS using a News Encoder (right) and a User Encoder (left) with multi-head attention. Image originates from *Neural News Recommendation with Multi-Head Self-Attention* [120].

All used models use a news encoder and a user encoder with an attention mechanism to learn the latent representation of the information.

4.2.2 Metrics

To evaluate the performance of recommendation models, we want to briefly introduce the used metrics in Recommenders.

Group Area-under-Curve

Area-under-Curve (AUC) is a function that quantifies the rate of *True Positive* and *False Positive* prediction. The *True Positive Rate* (also called *Sensitivity*) is defined as

$$TPR = \frac{TP}{TP + FN} \tag{4.1}$$

and the *False Positive Rate* (also referred to as 1−*Specificity*) is defined as

$$FPR = \frac{FP}{FP + TN}, \tag{4.2}$$

where TP is the number of correct positive recommendations, FN is the number of incorrect negative recommendations, FP is the number of incorrect positive recommendations, and TN is the number of correct negative recommendations. Both rates are bound to the interval of $[0, 1]$. Usually, a positive or negative recommendation is made due to a threshold (e.g., $p(click|user, news) > 0.5 \Rightarrow$ positive recommendation and $p(click|user, news) \leq 0.5 \Rightarrow$ negative recommendation). To identify a good threshold, for each threshold, a point is plotted with its False Positive Rate on the x-axis and True Positive Rate on the y-axis. When connecting these points, the result is called a *ROC curve*. **AUC calculates the area under the ROC curve**.

Note that a good threshold has a TPR close to 1 and an FPR close to 0. It is the trade-off between both rates. AUC gives a summarized performance of the used classification model. [39]

There is a weakness for AUC, especially in personalized RS like NRS. Usually, a RS recommends the top K items in terms of click probability $p(click|user, news)$. But not every group has similar probabilities. Thus the probabilities can not be calculated all at once. **Group AUC calculates the ROC AUC for each group by a weighted sum.** [56] In this work, it is convenient to group by user impressions. Thus the group AUC is calculated as

$$GAUC = \frac{1}{|Impressions|} \sum_{i \in Impression} AUC(labels_i, predictions_i). \tag{4.3}$$

nDCG@K

The **n***ormalized* **d***iscounted* **c***umulative* **g***ain* is a metric to quantify the relevance of ranked recommended items.
The discounted cumulative gain is defined as

$$DCG@K = \sum_{i}^{K} \frac{2^{rel_i} - 1}{\log_2(i + 1)} \tag{4.4}$$

where

$$rel_i = \begin{cases} 1 : \text{i'th item was clicked} \\ 0 : \text{else} \end{cases}$$

referring to the definition of Burges et al. [17]. To achieve a consistent cumulative relevance for each recommendation, the sum is normalized by the *ideal discounted cumulative gain* (iDCG). This is the DCG@K of the by relevance sorted recommendations. This ensures a maximal outcome. So nDCG@K is calculated as

$$nDCG@K = \frac{DCG@K}{iDCG@K}. \tag{4.5}$$

It is bound to $0 \leq nDCG@K \leq 1$.
In our experiments, we use the popular nDCG@5 and nDCG@10 metrics.

Mean MRR

Like in GAUC, Mean **M***ean* **R***eciprocal* **R***ank* is the slight variation of MRR that calculates the user impressions MRR and then averages over these results.
The Mean Reciprocal Rank for an impression is calculated as

$$MRR = \frac{1}{|I|} \sum_{i \in I}^{|} I| \frac{1}{rank(i)} \tag{4.6}$$

where $rank(i)$ is the i-th news actual relevant proposed news ordered by click probability.

5 Experiments

5.1 Model Performance Overview

We evaluated five modern news recommendation models to give an overview of modern approaches and their performance. These models are NRMS, LSTUR, NAML, NPA and DNK. All these models are trained for 10 epochs on the small dataset of *MIND*. For a faster and more consistent result, the batch size is set to 32. All models are optimized using an ADAM optimizer [59]. The list of relevant hyperparameters can be viewed in Table C.1. The table also indicates the structure of the models.

Results

After training for ten epochs on the small MIND dataset, NAML shows the best results in all four metrics (see Table 5.1). Each model's loss improves on each step (see Figure C.1). Except for DNK, all models produce similar results on the metrics. The results of group AUC lay in an interval of **0.6293-0.642**. The results of Mean-MRR are in an interval of **0.2856-0.3029**. The results of nDCG@K are in an interval of **0.3116-0.3352** for $K = 5$ and **0.3807-0.3975** for $K = 10$. DNK underperforms with with a Group AUC of **0.5762.**, a Mean-MRR of **0.1925**, a nDCG@5 of **0.1879** and a nDCG@10 of **0.2626**. The progression of the results can be viewed in Figure C.1 to Figure C.5.

Name	Group AUC	Mean MRR	ndcg@10	ndcg@5
DNK	0.5762	0.1925	0.2626	0.1879
NPA	0.6328	0.2874	0.3832	0.3165
NRMS	0.6293	0.2856	0.381	0.3116
NAML	**0.6563**	**0.3029**	**0.3975**	**0.3352**
LSTUR	0.6326	0.291	0.3807	0.3157

Table 5.1: Final metrics after 10 epochs of training on small MIND. Top performances are highlighted. NAML is showing off the best performance.

5.2 Investigating Biases

To follow the structure of the foundation's section, we divide the investigation of the biases into three parts, user-data, data-model, and model-user biases. Since our data is from MIND and relies on implicit feedback, not every bias mentioned in the foundations can be investigated in this part.
A complete list of all investigated biases can be seen in Table 5.2. A brief description, the type of feedback, whether there is a connection to other biases, the type of measurement, and the research basis with debiasing methods, if available, are presented. Biases that need to be shown by a user study are mentioned and discussed but beyond this paper's experimental scope.

5.2.1 User-Data Bias

5.2.1.1 Position Bias/Selection Bias

Selection bias and positional bias are closely related to implicit feedback. In both scenarios, we evaluate if users pick items shown earlier in the impressions.
For a user u there is no positional bias if

$$(p_u(n_k)) \approx \frac{1}{|I_u|} \tag{5.1}$$

where $p_u(n_k)$ is the probability that the kth news in an impression is clicked, I_u is the proposed impression, and $\{n_1, ..., n_{|i|}\} \in I$. The probability that a user picks news from a specific position is completely random or uniform. Otherwise, the data is positional biased. To evaluate (5.1), we slightly changed the equation. Instead of calculating the bias for a user, the general bias for all users is calculated. Since the number of news per impression differs, impressions with less than 5 items are skipped. There is also an assumption that each position's relevance is equal. The data is collected with unknown RS. However, there will be some relevant scores for the candidates.

Results

Figure 5.1 shows that the number of clicks per position decreases strongly. More than 4% of all clicks are items in the first position. Higher positions are less clicked.
The graph shows the clicked positions in an unweighted manner. This means positions are treated equally for every impression. But news clicked in a large impression should be more relevant than news clicked in a small impression. Therefore we introduce an exponential decay function to weigh the clicks. The weights are calculated via

$$f(x) = 1 - e^{-\frac{x+6}{15}} \in [0, 1] \tag{5.2}$$

Bias	Description	Feedback/ in MIND	Related to	Measurability in	Research	Debias
Inductive Bias	Model assumptions influence results	Both / ✓	-	Model;	[45]	-

Selection Bias	User rating is not random	Explicit/ x	Negativity bias;
			Anchoring;
				User study;
				Data;
					[78, 47]
					[78]
						[71, 103]

First match Bias (Selection bias)	Potential interesting news are not seen	Implicit / ✓	Exposure bias;	User study; Data	-	-
Position Bias	Higher positioned news are favoured by users	Both / ✓	Weber-Fechner-Law;	Data;	[90, 25] [26, 68]	[20, 121]
Context Bias	Interests are blurred by context/modality	ind./ x	Framing;	User study; Data	[129]	[129]
Exposure Bias	Ambiguity between not clicked and not liked news	ind. / n.i.	First match bias;	User study;	[68]	[127, 77]

Conformity Bias	User interests are manipulated;
	Skewed rating distribution
		Explicit* / x	Anchoring;
			Framing;
			Reactive Devaluation
				User study;
				(Data;)
					[61, 73]	[73, 69]

Behavioral Bias	User interests differ for different situations	ind. / n.i.	Conformity bias;	User study;	[29, 88]	-

Temporal Bias	User interests change over time	ind. / n.i.	-	User study;
				Data;
					[88]	-

Anchoring/ Framing	Interests are manipulated by reference points	Explicit* / x	-	User study;	[128]	-
Semmelweis Reflex/ Conservatism	New information are more likely to be denied	ind. / n.i.	-	User study;	[51, 102]	-
Attributional Bias	Fallacy that individuals have same interests as other individuals due to the same grouping	ind. / n.i.	Discrimination unfairness;	Model;	[74, 49]	-
Automation Bias	Proposed news are chosen because of trust to the machine	ind. / n.i.	-	User-study;	[35]	-

Reactive Devaluation	News are not clicked because of bad experiences with some entity like provider, users, author	Explicit* / n.i.	Personal experience;
			Unfairness;
				User-study;	[98]	-

Measurement Bias	Modelled measurement are chosen or used wrong	ind. / x	Inductive Bias;	Model;	[14, 108]	-

Omitted Variable Bias	Variables that not taken into account when collecting data or used by model	Both / ✓	Inductive bias;	Data;
				Model;
					[47]	-

Aggregation Bias	Fallacy about individuals due to wrong conditioning	ind. / n.i.	Attributional bias;	Model;	[107, 108]	-
Sampling/ Representation bias	Unrepresentative population on sampling inside subgroups	Both / n.i.	-	Data;	[62, 47] [108]	-
Longitudinal data fallacy	Fallacies due to cross-sectional study instead of longitudinal study	ind. / n.i.		Data;	[13]	-
Redundancy Ambiguity	Multiple same user interactions can be treated as very important or redundant	Both* / x	-	Data;	[88]	-
Popularity Bias	News are even more presented as their population suggests	Both/✓	Unfairness;	Data;Model;	[4, 73, 5]	[3]
Stance/Sentiment Bias	Reduced diversity due to overpopulation of negative or positive news	ind. / ✓	-	Data;Model;	[7, 65]	[122]

Recency	Recent interests are more relevant	ind. / ✓	Temporal bias;	Modell;
				Data;
					[54]	[86]

Cold Start Problem	New users get worse recommendations	ind. / ✓	Unfairness;	Model;	[42, 112] [64]	[64, 40]
Interests blurring (History length unfairness)	Worse recommendations for users with large news history	ind./ ✓	Unfairness;	Model;	-	-
Unfairness	Recommendation system systematically prefers/discriminates certain entities	ind. / ✓	-	Model;	[73, 5]	[16, 33] [94, 125] [77]
Emergent Bias	Model learns a certain preferential treatment/discrimination interacting with users in the feedback loop	Both[1] n.i.	Unfairness;	Model;	[55, 32]	-

Table 5.2: Overview for potential biases in news recommendation systems and the experiment. Feedback can be implicit, explicit, both, or feedback-independent (ind.). Comment-based feedback is noted with a star (*). Measurability shows at which stage the bias can be measured. Debias gives methods that mitigate the problems.

where x is the length of an impression. The regularized graph (Figure 5.2b) can be seen in comparison to the naive approach (Figure 5.2a) in Figure 5.2. The exponential decay counting reduces the gap between higher and lower positions. Both graphs show a positional bias assuming no differences in relevance. Using a Kolmogorov–Smirnov test to identify if the data is uniformly distributed, the result shows there is no chance that the data is uniformly distributed. **The visual and calculated results show a positional bias.**

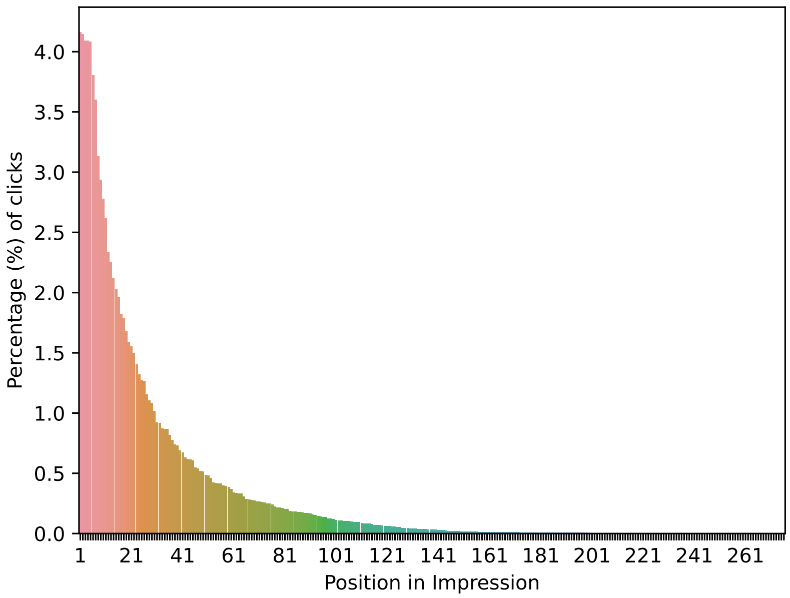
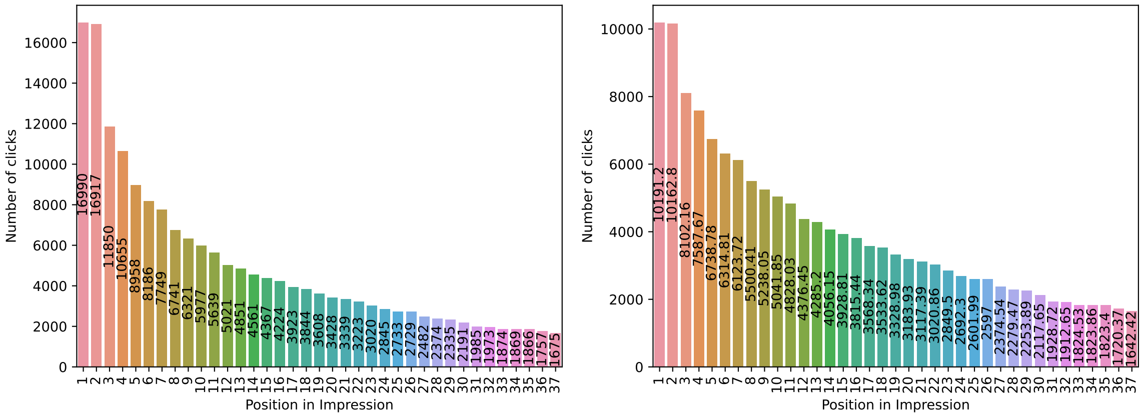


Figure 5.1: Percentages of each clicked position in MIND small. Impressions with less than 5 elements are shrunk.



(a) Simple Counting

(b) Exponential decay counting

Figure 5.2: Comparison of simple counting and exponential decay counting up to the mean of impression lengths 37.

5.2.1.2 Sentimental Bias

The data is classified by a pre-trained binary Destillbert [100] model to evaluate the potential of a sentimental bias. For each news in the dataset, the model predicts a *negative* or a *positive* sentiment. If the model is uncertain about a sentiment with a certainty of less than 0.5, the news is classified as *neutral*. A neutral sentiment will not be part of the evaluation.

The overall news sentiment proposed is slightly negative (see Figure 5.3). Thus there are about 41.84% of positive and 58.16% of negative classified sentiments in the news dataset. This means a factor of about 1.39 more negative than positive news.

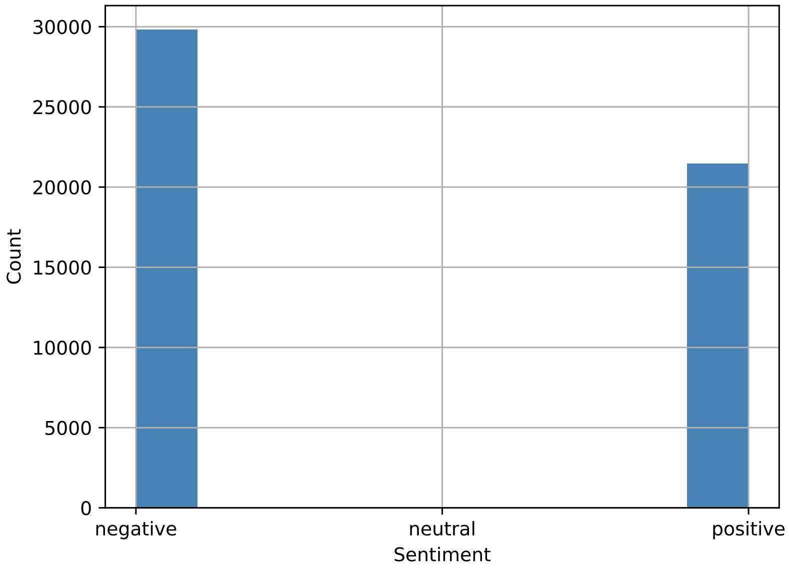


Figure 5.3: Overall sentiment in news dataset of MIND small

A sentimental bias is evaluated due to a potential user bias tending to pick a particular sentiment more likely and the over sentiment of impressions proposed to the user. The average user sentiment is calculated by

$$\text{sent}_u = \frac{1}{|H_u|} \sum_{n_k \in H_u} sent(n_k) \tag{5.3}$$

where H_u is the history of the user with news entries $\{n_1, ..., n_{|H_u|} \in H_u$ and

$$sent(n_k) = \begin{cases} 1 & : n_k \text{ has a positive sentiment} \\ -1 & : n_k \text{ has a negative sentiment} \end{cases}$$

The average impression sentiment for a user is calculated by

$$\text{sent}_{I_u} = \frac{1}{|I_u|} \sum_{i \in I_u} \sum_{n_k \in i} \frac{sent(n_k)}{|i|} \tag{5.4}$$

where are I_u are the impressions proposed to the user and $\{n_1, ..., n_{|i|}\} = i \in I_u$ are the news in impression i.
The sentimental bias is then the absolute deviation of (5.2) and (5.3)

$$sb = \text{sent}_u - \text{sent}_{I_u}. \tag{5.5}$$

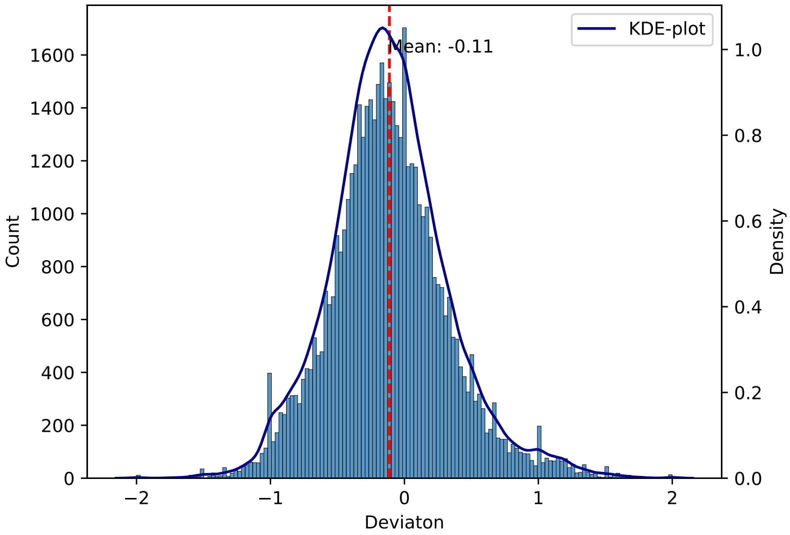


Figure 5.4: Histogram/KDE-plot of the point-wise deviations of behavior and impression sentiments

The data is unbiased regarding sentiment if $sb \approx 0$.

Results

Behaviors and impressions are averaged for each user. Both sentiments are, on average, more negative than positive (see Table 5.3 and Figure 5.5). The sentiment of the user behaviors is, on average, 17.13% positive and 73.15% negative. This factor is 4.27 more negative than positive news clicked by users in the past. The impressions proposed to the users have an average sentiment of 18.13% positive and 77.45% negative. In comparison, it is slightly higher negativity in the sentiment. The impressions sentiment has a mean of about -0.16 but less standard deviation than the behaviors sentiment. Therefore it is closer to neutral sentiment.

The point-wise deviation of each user's history to its impression proposed can be viewed in Figure 5.4. The deviation is the difference between the user's behavior and the proposed impression by the model. A negative deviation means that the behavior is more negative and a positive deviation means that the behavior is more positive in terms of sentiment. The deviation is bound to the interval $[-2, 2]$.

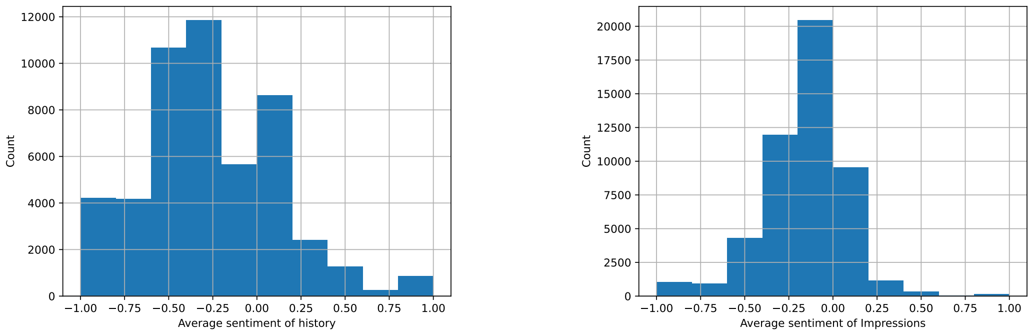
The sentiment bias is equal to the mean, which is

$$sb = -0.11. \tag{5.6}$$

Thus **the system proposed more positive news to the user** than the user used to click.

	Avg. positive	Avg. negative	Mean	Stand. Dev.
Behavior sentiment	8565/50000 (17.13%)	36573/50000 (73.15%)	-0.271557	0.397828
Impression sentiment	9063/50000 (18.13%)	38725/50000 (77.45%)	-0.159119	0.240844

Table 5.3: Comparison of the behavior sentiment and the impression sentiment.



(a) History/Behaviors

(b) Impressions

Figure 5.5: Comparison of impression and behavior sentiment

5.2.2 Model-User Bias

5.2.2.1 Popularity Bias

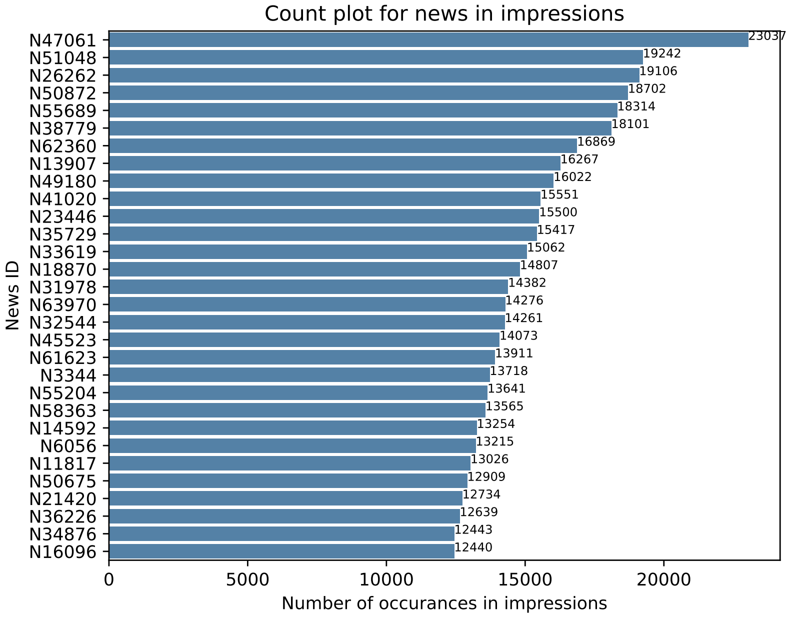


Figure 5.6: Top 30 news in terms of occurrences in impressions.

To evaluate a popularity bias in MIND, we looked at the frequencies of suggested news. To give evidence, the frequencies are compared with the frequency distribution of the histories, which can be seen as the relevance of news.

Title	Category	Subcategory	Show Ratio
Black Friday Deals You Can Start Shopping Today	lifestyle	shop-holiday	14.677%
Rep. Tim Ryan endorses Biden in Democratic primary	news	elections-2020-us	12.259%
Celebrity plastic surgery transformations	entertainment	entertainment-celebrity	12.172%
50 amazing gifts for every type of person and budget	lifestyle	shop-holidays	11.915%
Charles Rogers, former Michigan State football, Detroit Lions star, dead at 38	sports	football_nfl	11.668%

Table 5.4: Top most presented news in impressions.

Results

To give an overview, there are **20288 different news** shown to the users in MIND small. The ratio of new news is about 0.13. The most shown news is shown 23037 times (see Figure 5.6) whereas 11471/20288 news (56.5%) are shown less or equal 10 times. The data is longtailed, and a small number of articles is presented very often.
Comparing these results to users' behaviors, there is an over popularity in *lifestyle* and especially in *shop-holiday* subcategories.

5.2.2.2 History Length Unfairness

The hypothesis is that users with different histories get different recommendations in history length unfairness. In this unfairness bias, we used an *NRMS* model pre-trained for 5 epochs on the training set and a modified training set. The tests are set up for five user groups: users with 1, 10, 25, 50, and 100 history entries. The reason for stopping at a history size of 100 is that there are not enough users with more than 100 entries. In the small dataset are only about 6.1% of histories greater than 100 and only 2.2% greater than 150. Otherwise, there would be an underrepresentation. To achieve this pre-setting, the training set is modified for every user group in the training set to hold at least the number of history entries investigated. For instance, the user group with a history length of 50 is trained on the training set with users of at least 50 behaviors. The history is then shrunk to the exact number of entries to have an exact comparison. Thus only the top recent entries are used.
Every user group has its trained *NRMS*-model. For evaluation, the metrics *Group AUC*, *nDCG@5,10*, and *Mean MRR* are used. For consistency purposes, the model is trained on three seeds, and the results are averaged over the single results of a seed. This also ensures reproducibility. MIND only proposes a training and validation set. The models are evaluated on a custom test set to assess the generalization better. The test set is randomly

sampled from unseen data in the large dataset of MIND.
Note that a hyperparameter for history size had to be set before training the models. It ensures that the model can pay attention to this number of entries.

Results

To see the differences between the influence of users with different history sizes, see the figures and table below. Except for nDCG@5, the best results are on a History size of $|H| = 10$ or $|H| = 25$. The results for the training with a modified version of the training set are much closer together than with training on the basic training set. Only the nDCG@5 evaluation is slightly better on the modified training set in terms of training and the evaluation on the test set afterward. A visualization of nDCG@10 can be seen in Figure 5.7.

Metric	**|H| = 1**	**|H| = 10**	**|H| = 25**	**|H| = 50**	**|H| = 100**
Mean MRR	0.2705 0.2462	0.3072 0.2476	0.3078 0.2462	0.2747 0.244	0.2524 0.2418
Group AUC	0.6113 0.5526	0.6506 0.5574	0.6506 0.5538	0.6138 0.5499	0.5852 0.5495
nDCG@5	0.2935 0.2621	0.3362 0.2618	0.3376 0.2589	0.2992 0.2568	0.2744 0.256
nDCG@10	0.3608 0.3222	0.4012 0.3236	0.4011 0.3223	0.3646 0.3205	0.339 0.319
Mean MRR (Test)	0.2698 0.2467	0.3055 0.2481	0.3058 0.2469	0.2739 0.2446	0.2516 0.2423
Group AUC (Test)	0.6111 0.5522	0.651 0.5568	0.6505 0.5533	0.6133 0.5493	0.5839 0.5487
nDCG@5 (Test)	0.2926 0.263	0.3342 0.263	0.3355 0.2601	0.2984 0.2579	0.2735 0.257
nDCG@10 (Test)	0.3598 0.3231	0.3997 0.3246	0.399 0.3234	0.3635 0.3212	0.3377 0.3199

Table 5.5: Table of the performance on different history sizes. Compare the standard training set (upper value) and the modified training set (lower value). The top results are underlined.

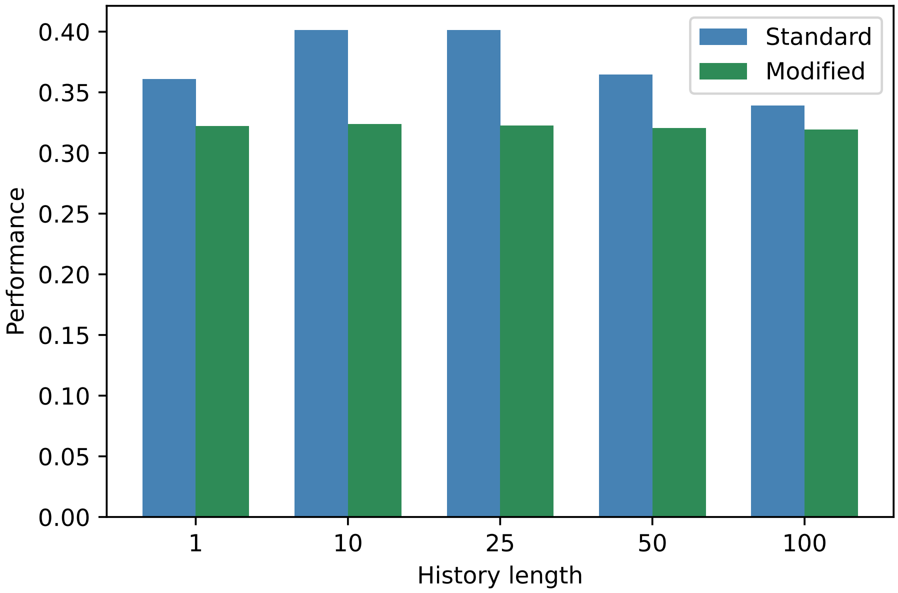
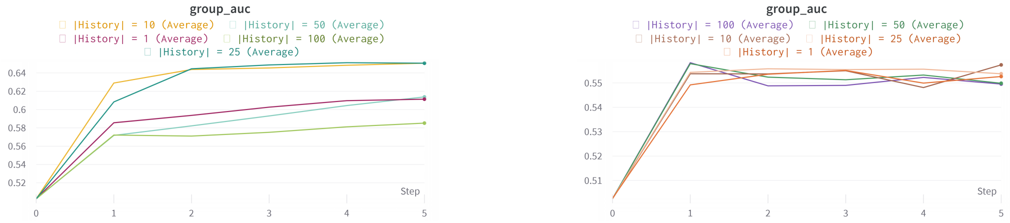


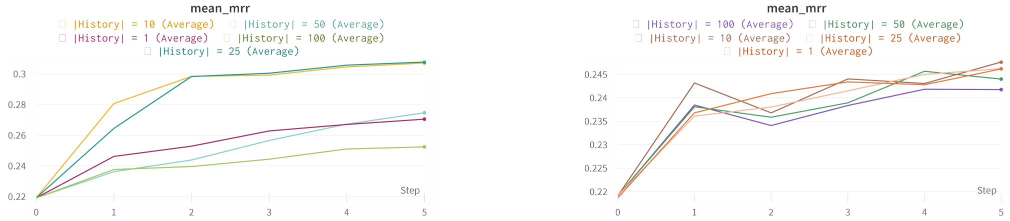
Figure 5.7: Performance in nDCG@10 for different history sizes comparing the standard and modified training set.



(a) Standard training set

(b) Modified training set

Figure 5.8: Comparison of Group AUC



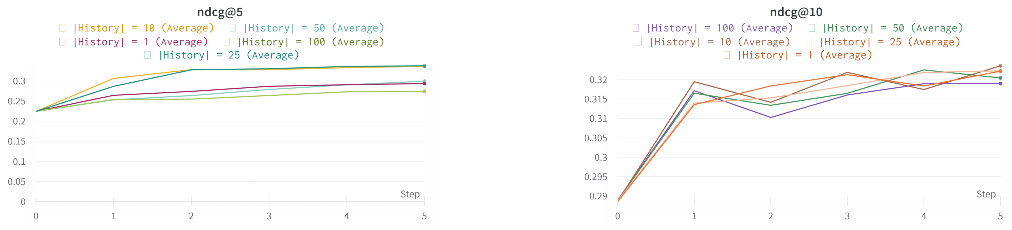
(a) Standard training set

(b) Modified training set

Figure 5.9: Comparison of Mean MRR

5.2.3 Recency Bias

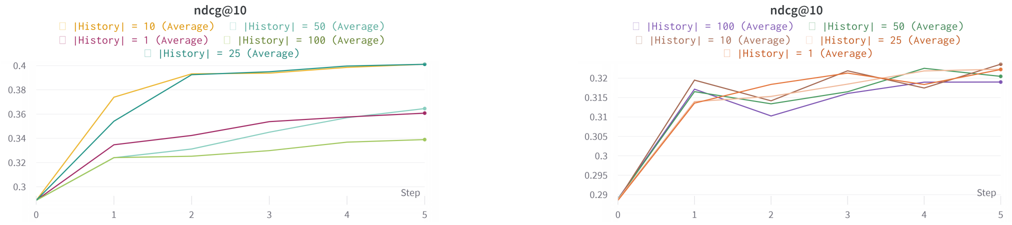
The recency bias in news recommendations is the tendency for more recent news to have more impact on the recommendations. Thus recent news in user behaviors has a stronger correlation to user interests. This test was originally a continuation of the tests for history



(a) Standard training set

(b) Modified training set

Figure 5.10: Comparison of nDCG@5



(a) Standard training set

(b) Modified training set

Figure 5.11: Comparison of nDCG@10

length unfairness. To evaluate this bias, we looked at the attention weights after the 5 epochs of training. Using *NRMS* as a model, there are three types of attention. Here, we focus on the attention of news in users' history. This is related to the history size since attention is fixed. A recency bias is present if there is a tendency for attention weights to get higher over time.

Results

Usually, we deal with multi-head attention. Therefore we took the sum over all heads

$$a_k(u) = \sum_i^h a_i^h$$

, where $h \in H$ is the hth head in H and $k \in B_u$ is the kth news in the History of a user u.

The results show that attention weights for news in user behaviors tend to decrease. There is no real monotony. The most recent values are about 2x as important as the oldest ones. These results remain consistent for different sizes of histories.

5.2.4 Potential User and Explicit Feedback Biases

Context bias. Since no modalities such as video or image content are specified in MIND, a context bias is not detectable in this work. However, verification in news recommendation

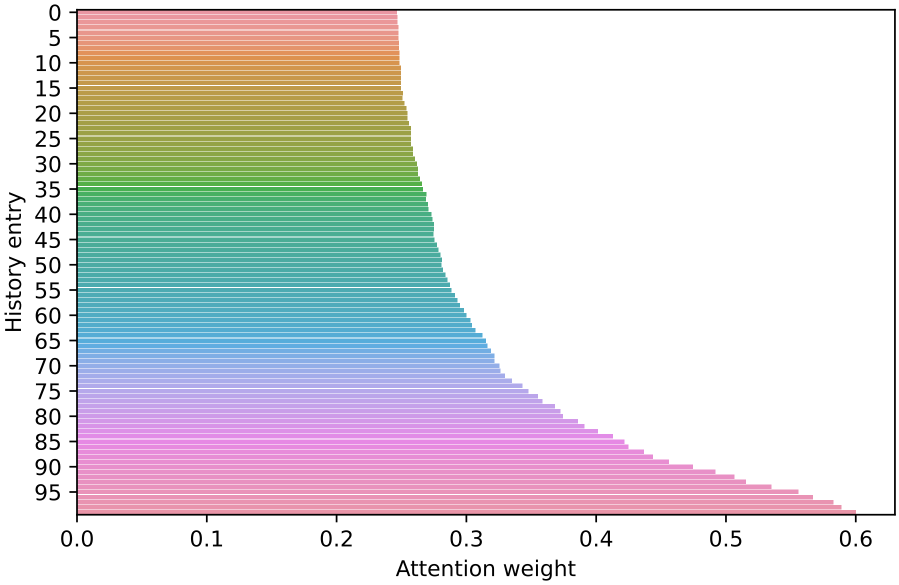


Figure 5.12: Attention weights for each entry in the user history (here with 100 entries). Higher entry means more recent.

systems with given modalities is quite possible. Context bias is closely related to the framing effect. Both describe a shift in perception due to the context of an environment.

Conformity bias is also not measurable. Conformity biases are based on the evaluations of other users. This can be found in explicit feedback. Other sources of conformity can also be comments. Many news providers allow users to comment on the news they read. This influence, however, also refers only to explicit feedback. The opinion and ratings of other users can be seen as an anchor. Therefore, conformity bias is closely related to anchoring and framing. The counterpart to conformity is reactive devaluation. Instead of pursuing similar interests, the opposite is done on principle.

Behavioural and temporal biases are difficult to measure for news without a user study and are also not assessed. However, new interests can be found in the data. As we have seen, more recent news is more relevant than older news. The recency bias is a direct subproduct of the temporal bias.

Semmelweis reflex and conservatism. The Semmelweis reflex and conservatism are effects that tend to make users reject new information. Modeling this behavior is of particular importance for news recommendations. For example, new information in a news item could lead users to not click on news items, rate them less favorably, or comment negatively.

Attributional bias. Attributional bias is a threat to performance, especially in collaborative filtering. Since users are matched with similar users, it can lead to a false conclusion of similar interests.

Measurement and omitted variable bias are a subcategory of inductive biases. Since only clicks were collected in MIND, measurement errors are unlikely. With the use of NRMS in our experiment, we have a restriction on the user's history. Omitted variables on the model side (internal) are the time and the body that can be accessed via URL. External omitted variables are none on the old MSN page. However, when data is collected again on the current MSN page, variables such as user comments and ratings could be included for additional user information. However, it is more challenging to maintain the anonymity of the users, as this information is public. Aggregation biases are difficult to measure because they arise from incorrect model conditioning. Explainable models or methods of interpretability are necessary for this.

Representation bias and longitudinal data fallacy. Due to time constraints, the check for a sampling bias was not carried out. The data from MIND were collected over six weeks and is therefore influenced by the current state of the world during this time. For example, the most commonly suggested message was an overview of Black Friday deals. This event is annual and therefore very unrepresentative for the whole year. So a longitudinal study is potentially more useful to get a more accurate picture. A longitudinal data fallacy is therefore also possible and related to this. To get a good assessment of this fallacy, however, the data would have to be collected again as a longitudinal study.

Redundancy ambiguities are not a problem in MIND, as there is simply no redundant data. Other and especially comment-based datasets may contain this problem. For example, multiple viewing or multiple commenting can be interpreted as particularly relevant news or as a mishap.

Unfairness. There is already a lot of research on unfairness, including unbiasing. We have seen that there is an unjustified frequency of marketing lifestyle news. Other tests could include assessing whether there is provider unfairness of the model as studied by Qi et al. [94].

6 Discussion

6.1 Performance Tests

Contrary to expectations, in the performance test, NAML performed better than NRMS. The expectation that NRMS is the best performer is due to the better results in, for example, [121, 126]. The reason for our results may be the restriction to a single run. For time reasons, we did not take an average of several runs. The result could not be representative.

6.2 Investigating and Discussion of Biases

In our experiments, we were able to see some potential biases in the data and model. In this chapter, we want to discuss these findings, referring to our intuitions and results by other researchers. The experiments are limited to MIND, which has collected data from implicit feedback. An overview is shown in the Table 5.2. It shows all biases we experimented with, potential biases, and biases for further investigation.

For a **positional bias**, the tests show that there is a strong tendency of users to pick news that are positioned higher in the impression than lower (see Figure 5.2). Although, there is no measurement for relevance in the proposed items. The data in MIND is captured from an already existing news recommendation system. According to the authors of the MIND-paper [126], the underlying system uses state-of-the-art approaches for news recommendations. These include embeddings for news and users, time awareness, and personalization. The actual model is hidden from the public. So the predictions, which give a measurement of certainty for a proposed item, are not accessible. We introduced the weight function to give more credit to picking a particular position from a bigger impression (see Equation 5.2). This function is only for intuition. There is no mathematical derivation for it. It is a function of exponential decay clipped to $[0, 1]$. A picked item at position k should be more relevant if picked from many items. However, there has to be a saturation limit. Users may tend to have a sensory overload. This is also closely related to cognitive biases like the *Weber-Fechner law* (see 3.2.2.9). It states that the human perception of a countable stimulus is logarithmic. So it suggests that this effect occurs in news recommendations.

Using this information to build a weight function and assume an equivalent relevance for every position, the data is positional biased. There are two major problems with it. User behaviors may have less relevant news, and models learn from this information. Also, there is unfairness in the news. When clicking news in lower positions is more likely, there has to be a fair chance for news to get shown in these positions. There is a relation to a user-sided selection bias. Especially users looking for a certain source may only interact with the first matching item. In MIND, about 72.56% of impressions only have one clicked news (see right graph of Figure 4.5). More than 90% have less than 3 clicks. This means higher-positioned items have a better chance of being clicked.

Figure 5.3 shows a tendency for more negative than positive news. A pre-trained BERT model classified this. To the best of our knowledge, there is no researched reason for this tendency for negativity. Looking at the sentiment in the user behaviors, there is an average of 73.15% of news sentiments. The mean of the average sentiment is at about -0.272. So the users tend to click more negative news. This is supported by Soroka [105], who has shown a negativity bias in news attention. Later, it was shown that there is a stronger psychological effect of negative news by Soroka et al. [106]. These results might affect the overall news sentiment. MIND data is collected from an intact feedback loop. Because users are more willing to click negative views, news providers might also produce more negative news. This systematic negativity bias would produce an unequal distribution of positive and negative news.

The impression sentiment shows a very different distribution of average sentiments. Figure 5.5b shows a left-skewed normal distribution with a mean at about -0.16. The average sentiment of impressions is 0.112 for the behaviors. Also, the spread of average sentiments is less. The results differ from those proposed by Alam et al. [7]. In this work, we showed that the impressions proposed to the users are, on average, more positive than the user behaviors. Even though there are more negative sentiments overall. The news recommendation system makes impressions. These results may not be unusual to follow an underlying normal distribution with less extreme values close to the edge. The users' behaviors have much more values close to the extreme. Many users have very negative interests and a view of very positive news in terms of sentiment. The extremes are not modeled well by the underlying model. The data has sentimental bias and proposes **on average, more positive impressions.**

We trained a model on different history sizes in the history length unfairness experiment. The results showed differences in performances. The intuitional hypothesis was that a more extensive user history leads to better recommendations. The reason is that there is more information about a user. However, the hypothesis is rejected. The best performances are achieved with a **user's history length of** 10-25. Even users with only a single item in their history performed better on all metrics than users with 100 items in their history. The **interests seem be blurred** due to to much news entries in the history. Many researchers like Lika et al. [64] show that recommendation tasks have cold-start problems. It means new users get worse recommendations. In our experiments, users with only 1 item in the history are cold-start users. The recommendation system is not able to generalize the interests. This explains the worse performance. On the other hand, users with a big history length have the worst performance and get outperformed by users with 10-25 entries. This is contrary to the recommended size of 50 history entries in most of the modelsTable C.1. To the best of our knowledge, there is no researched reason for this. Looking at the attention weights proposed in the recency bias section (see Figure 5.12), the attention is nearly monotonically decreasing over time. So later viewed news are more relevant than earlier ones. These results are supported by the research of Ji et al. [54]. They show that active users get better recommendations. This is because of a temporal bias in users' preferences. User preferences change over time, so recommendation systems must model this behavior. We see a **recency bias for our attention-based model**. It gives a potential explanation for the problem with large user behaviors. **Older news seems to blur the interests of users**. It leads to worse recommendations. The sweet spot for the best amount of user behaviors seems to lie between 10 and 25 entries.

Technically, there is a popularity bias in the dataset. Some news are shown very often in users' impressions (see Table 5.4), but more than 56% of news are shown less or equal to 10 times. The most shown news is 23.037 times shown to users, which means it was shown in 14.677% of all impressions. It is hard to find an explanation for these extreme frequencies. Taking a look at the title gives an intuition of the reason. The most popular news is titled "*Black Friday Deals You Can Start Shopping Today*". This suggests that the popularity bias is systematic. It is a sort of advertisement and might be explicitly set inside an impression. This would mean no popularity bias in the recommendation model but a systematic unfairness by the provider.

6.3 Limitations of the work

This work has some limitations in the experiments. *MIND* is a collection with implicit feedback from users. This means that interests are only encoded binary, interested and not interested in news. In contrast, explicit feedback is more diverse and gives more gradation in terms of user interests. In this work, we focus only on implicit feedback. A big reason for this is that there are only way smaller datasets with ratings. This changes the investigation of biases, as well. Biases like anchoring effect, reactive devaluation, and conformity bias are more applicable to explicit feedback. Furthermore, the model biases are only evaluated on NRMS. NRMS has a good performance and uses a modern but simple approach to provide content-based recommendations. There is big research in news recommendation systems with plenty of new models every year. We picked this one to show how to evaluate these different biases in news recommendations.

Another limitation is the consideration of only two of the three stages of biases in the feedback loop. User biases are mainly cognitive biases. These are psychological effects that change users' opinions on news items. To quantify cognitive biases in news recommendation systems, we must conduct user studies that are very human and time resource expensive. However, user biases end up in data, and some data biases can be supplied to user biases. This can be a temporal bias causing the model to favor more recent news or a negativity bias causing an overall more negative sentiment. Knowing these biases and giving a complete overview of them is necessary.

7 Future works

The limitations of this work already give an intuition of future works on bias investigation in news recommendation. Since *MIND* gives only implicit feedback, more experiments on explicit feedback are possible. There are datasets like Bing News[1] and NewsReel[2] for explicit feedback evaluation. The dimension of these datasets is way smaller than MIND. This makes it harder to train models. Therefore, current research focuses on different recommendation models like movie recommendation with MovieLens [43]. Currently, news recommendation with feedback lacks a large-scale dataset. There is research with sufficient datasets like BING news, but it is not released to the public by Microsoft.

There are some potential biases in explicit feedback. User-sided selection bias is shown for rating-based collaborative filters by Marlin et al. [78]. Again, the showcase of their work is limited to Yahoo, Netflix, and MovieLens data. Another bias to investigate in explicit feedback is the anchoring effect. Proposing different scales of ratings has a different effect on the users. For instance, since the collection of MIND, MSN[3] has changed its news page to allow users to like or dislike news with a binary rating system. But ratings on news providers are still uncommon.

The anchoring effect is, in another sense, closely related to the conformity bias. The scale of rating is a provider-based anchoring. At the same time, different user ratings can be seen as user-based anchoring. It is the same as conformity bias, which influences different user opinions. For these types of biases, a user study would be appropriate. Further investigation could be done on reactive devaluation. Feedback data can be collected to see if you systematically devaluate certain use providers or news topics.

[1] https://github.com/hwwang55/DKN/tree/master/data/news

[2] https://www.newsreelchallenge.org/dataset/

[3] https://www.msn.com/de-de/nachrichten

8 Conclusion

In this master thesis, we investigated the different biases occurring in NRS. The first step was to look at the current state of research to identify challenges, gaps, and existing research on biases. In this way, a large list of different biases could be identified. This research comes mainly from psychology and general recommendation systems. The biases were divided into three stages, i.e., user, data, and model. For our experimental work, the MIND dataset was used, in which data was collected from 1,000,000 users who accessed MSN news. The experimental work is carried out based on this data. It shows an insight into bias for implicit feedback, which is much more common in news recommendations than explicit feedback. Moreover, the experiment was limited to data and model biases. To give a more general overview a table in which all possible biases are provided.

To get an overview of current models, the models NRMS, DNK, NAML, LSTUR, and NPA were trained and tested for their performance at the beginning of the experiment. The results of the performance tests showed that the tested models all performed similarly until DNK. This achieved the worst results. NAML achieved the best result in all the metrics tested. In the main experiment, we have seen a bias in data and model and potential biases for users. Showing, there is a position bias in the displaying of news. News displayed higher up is potentially clicked more often than news displayed further back. The clicks were weighted with an exponential decay function that depended on the impression size. The bias could only be shown assuming that the news items displayed are equally relevant. The average sentiment of the user history and impression was compared for the sentiment bias. The results show that users prefer negative news more than positive news. The compared values provided similar results. Nevertheless, a negative sentiment bias could be shown. The system suggests more positive than negative news than the user history of news. We also saw that a possible popularity bias exists. Since many over-popularised news items are advertisements, it seems reasonable to assume that they are systematically inserted. In the test with different user history lengths, the most extensive histories of 50 and 100 performed significantly worse than those of 10 and 25. Upon further investigation, we showed that the model emphasizes more current interests. They become blurred as the size of the user history increases. At the other extreme, the model cannot perform well on new news using a single news item in history.

The results in this and other works show the need for investigations on biases. Many biases are based on users' perceptions, causing skewed data. Also, models still struggle to deal with these issues leading to popularity and recency biases. Our table with the list of biases shall serve as a checklist for upcoming research for data collection and model creation.

Bibliography

[1] 2023. URL: https://upload.wikimedia.org/wikipedia/commons/6/65/Cognitive_bias_codex_en.svg.

[2] Hervé Abdi and Lynne J. Williams. "Principal component analysis". In: *WIREs Computational Statistics* 2.4 (2010), pp. 433–459. DOI: https://doi.org/10.1002/wics.101. eprint: https://wires.onlinelibrary.wiley.com/doi/pdf/10.1002/wics.101. URL: https://wires.onlinelibrary.wiley.com/doi/abs/10.1002/wics.101.

[3] Himan Abdollahpouri, Robin Burke, and Bamshad Mobasher. "Controlling popularity bias in learning-to-rank recommendation". In: *Proceedings of the eleventh ACM conference on recommender systems*. 2017, pp. 42–46.

[4] Himan Abdollahpouri and Masoud Mansoury. "Multi-sided Exposure Bias in Recommendation". In: *CoRR* abs/2006.15772 (2020). arXiv: 2006.15772. URL: https://arxiv.org/abs/2006.15772.

[5] Himan Abdollahpouri, Masoud Mansoury, Robin Burke, and Bamshad Mobasher. *The Unfairness of Popularity Bias in Recommendation*. 2019. DOI: 10.48550/ARXIV.1907.13286. URL: https://arxiv.org/abs/1907.13286.

[6] Zahra Ahmad. "Recommender Systems: Explicit Feedback, Implicit Feedback and Hybrid Feedback". In: *Analytics Vidhya* (2021). URL: https://medium.com/analytics-vidhya/recommender-systems-explicit-feedback-implicit-feedback-and-hybrid-feedback-ddd1b2cdb3b (visited on 10/17/2022).

[7] Mehwish Alam, Andreea Iana, Alexander Grote, Katharina Ludwig, Philipp Müller, and Heiko Paulheim. "Towards Analyzing the Bias of News Recommender Systems Using Sentiment and Stance Detection". In: *Companion Proceedings of the Web Conference 2022*. ACM, Apr. 2022. DOI: 10.1145/3487553.3524674. URL: https://doi.org/10.1145%5C%2F3487553.3524674.

[8] Nabila Amir, Fouzia Jabeen, Zafar Ali, Irfan Ullah, ASIM JAN, and Pavlos Kefalas. "On the current state of deep learning for news recommendation". In: *Artificial Intelligence Review* (May 2022). DOI: 10.1007/s10462-022-10191-8.

[9] Mingxiao An, Fangzhao Wu, Chuhan Wu, Kun Zhang, Zheng Liu, and Xing Xie. "Neural News Recommendation with Long- and Short-term User Representations". In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Florence, Italy: Association for Computational Linguistics, July 2019, pp. 336–345. DOI: 10.18653/v1/P19-1033. URL: https://aclanthology.org/P19-1033.

[10] Eric Arazo, Diego Ortego, Paul Albert, Noel E. O'Connor, and Kevin McGuinness. "Pseudo-Labeling and Confirmation Bias in Deep Semi-Supervised Learning". In: *2020 International Joint Conference on Neural Networks (IJCNN)*. 2020, pp. 1–8. DOI: 10.1109/IJCNN48605.2020.9207304.

[11] L. Backman and L. Nyberg. *Memory, Aging and the Brain: A Festschrift in Honour of Lars-G ran Nilsson*. Psychology Press Festschrift Series. Taylor & Francis, 2009. ISBN: 9780203866665. URL: https://books.google.de/books?id=0w1F7YlJ3RcC.

[12] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. *Neural Machine Translation by Jointly Learning to Align and Translate*. 2014. DOI: 10.48550/ARXIV.1409.0473. URL: https://arxiv.org/abs/1409.0473.

[13] Samuel Barbosa, Dan Cosley, Amit Sharma, and Roberto M. Cesar. "Averaging Gone Wrong". In: *Proceedings of the 25th International Conference on World Wide Web*. International World Wide Web Conferences Steering Committee, Apr. 2016. DOI: 10.1145/2872427.2883083. URL: https://doi.org/10.1145%5C%2F2872427.2883083.

[14] Solon Barocas and Andrew D Selbst. "Big data's disparate impact". In: *Calif. L. Rev.* 104 (2016), p. 671.

[15] Abraham Bernstein et al. "Diversity in News Recommendation". en. In: (2021). DOI: 10.4230/DAGMAN.9.1.43. URL: https://drops.dagstuhl.de/opus/volltexte/2021/13745/.

[16] Asia J Biega, Krishna P Gummadi, and Gerhard Weikum. "Equity of attention: Amortizing individual fairness in rankings". In: *The 41st international acm sigir conference on research & development in information retrieval*. 2018, pp. 405–414.

[17] Chris Burges, Tal Shaked, Erin Renshaw, Ari Lazier, Matt Deeds, Nicole Hamilton, and Greg Hullender. "Learning to Rank Using Gradient Descent". In: *Proceedings of the 22nd International Conference on Machine Learning*. ICML '05. Bonn, Germany: Association for Computing Machinery, 2005, pp. 89–96. ISBN: 1595931805. DOI: 10.1145/1102351.1102363. URL: https://doi.org/10.1145/1102351.1102363.

[18] Claude Castelluccia. "Behavioural Tracking on the Internet: A Technical Perspective". In: *European Data Protection: In Good Health?* Ed. by Serge Gutwirth, Ronald Leenes, Paul De Hert, and Yves Poullet. Dordrecht: Springer Netherlands, 2012, pp. 21–33. ISBN: 978-94-007-2903-2. DOI: 10.1007/978-94-007-2903-2_2. URL: https://doi.org/10.1007/978-94-007-2903-2_2.

[19] J-P Caverni, J-M Fabre, and Michel Gonzalez. *Cognitive biases*. Elsevier, 1990.

[20] Olivier Chapelle and Ya Zhang. "A Dynamic Bayesian Network Click Model for Web Search Ranking". In: *Proceedings of the 18th International Conference on World Wide Web*. WWW '09. Madrid, Spain: Association for Computing Machinery, 2009, pp. 1–10. ISBN: 9781605584874. DOI: 10.1145/1526709.1526711. URL: https://doi.org/10.1145/1526709.1526711.

[21] Jiawei Chen, Hande Dong, Xiang Wang, Fuli Feng, Meng Wang, and Xiangnan He. *Bias and Debias in Recommender System: A Survey and Future Directions*. 2020. DOI: 10.48550/ARXIV.2010.03240. URL: https://arxiv.org/abs/2010.03240.

[22] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. *A Simple Framework for Contrastive Learning of Visual Representations*. 2020. DOI: 10.48550/ARXIV.2002.05709. URL: https://arxiv.org/abs/2002.05709.

[23] Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. *Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation*. 2014. DOI: 10.48550/ARXIV.1406.1078. URL: https://arxiv.org/abs/1406.1078.

[24] Hans Christian, Mikhael Pramodana Agus, and Derwin Suhartono. “Single document automatic text summarization using term frequency-inverse document frequency (TF-IDF)”. In: *ComTech: Computer, Mathematics and Engineering Applications* 7.4 (2016), pp. 285–294.

[25] Andrew Collins, Dominika Tkaczyk, Akiko Aizawa, and Joeran Beel. “Position bias in recommender systems for digital libraries”. In: *International Conference on Information.* Springer. 2018, pp. 335–344.

[26] Andrew Collins, Dominika Tkaczyk, Akiko Aizawa, and Jöran Beel. “A Study of Position Bias in Digital Library Recommender Systems”. In: *CoRR* abs/1802.06565 (2018). arXiv: 1802.06565. URL: http://arxiv.org/abs/1802.06565.

[27] Scott Deerwester, Susan T Dumais, George W Furnas, Thomas K Landauer, and Richard Harshman. “Indexing by latent semantic analysis”. In: *Journal of the American society for information science* 41.6 (1990), pp. 391–407.

[28] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. *BERT: Pretraining of Deep Bidirectional Transformers for Language Understanding.* 2018. DOI: 10.48550/ARXIV.1810.04805. URL: https://arxiv.org/abs/1810.04805.

[29] Katharina Dowling, Daniel Guhl, Daniel Klapper, Martin Spann, Lucas Stich, and Narine Yegoryan. “Behavioral biases in marketing”. In: *Journal of the Academy of Marketing Science* 48.3 (2020), pp. 449–477.

[30] Eurostat. *Consumption of online news rises in popularity.* Aug. 2022. URL: https://ec.europa.eu/eurostat/web/products-eurostat-news/-/ddn-20220824-1.

[31] Lapo Filistrucchi. “The impact of Internet on the market for daily newspapers in Italy”. In: (2005).

[32] Batya Friedman and Helen Nissenbaum. “Bias in Computer Systems”. In: *ACM Trans. Inf. Syst.* 14.3 (July 1996), pp. 330–347. ISSN: 1046-8188. DOI: 10.1145/230538.230561. URL: https://doi.org/10.1145/230538.230561.

[33] Sahin Cem Geyik, Stuart Ambler, and Krishnaram Kenthapadi. “Fairness-Aware Ranking in Search Recommendation Systems with Application to LinkedIn Talent Search”. In: *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery.* ACM, July 2019. DOI: 10.1145/3292500.3330691. URL: https://doi.org/10.1145%5C%2F3292500.3330691.

[34] GISGeography. *MAUP - Modifiable Areal Unit Problem - GIS Geography.* July 2015. URL: https://gisgeography.com/maup-modifiable-areal-unit-problem/.

[35] Kate Goddard, Abdul Roudsari, and Jeremy C Wyatt. “Automation bias: a systematic review of frequency, effect mediators, and mitigators”. In: *Journal of the American Medical Informatics Association* 19.1 (June 2011), pp. 121–127. ISSN: 1067-5027. DOI: 10.1136/amiajnl-2011-000089. eprint: https://academic.oup.com/jamia/article-pdf/19/1/121/5911703/19-1-121.pdf. URL: https://doi.org/10.1136/amiajnl-2011-000089.

[36] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning.* http://www.deeplearningbook.org. MIT Press, 2016.

[37] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. "Generative Adversarial Nets". In: *Advances in Neural Information Processing Systems.* Ed. by Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K.Q. Weinberger. Vol. 27. Curran Associates, Inc., 2014. URL: https://proceedings.neurips.cc/paper/2014/file/5ca3e9b122f61f8f06494c97b1afccf3-Paper.pdf.

[38] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. *Explaining and Harnessing Adversarial Examples.* 2014. DOI: 10.48550/ARXIV.1412.6572. URL: https://arxiv.org/abs/1412.6572.

[39] Google Developers. *Klassifizierung: ROC-Kurve und AUC | Machine Learning | Google Developers.* 2022. URL: https://developers.google.com/machine-learning/crash-course/classification/roc-and-auc (visited on 11/21/2022).

[40] Jyotirmoy Gope and Sanjay Kumar Jain. "A survey on solving cold start problem in recommender systems". In: *2017 International Conference on Computing, Communication and Automation (ICCCA).* 2017, pp. 133–138. DOI: 10.1109/CCAA.2017.8229786.

[41] Luigi Guiso, Paola Sapienza, and Luigi Zingales. "Cultural Biases in Economic Exchange?*". In: *The Quarterly Journal of Economics* 124.3 (Aug. 2009), pp. 1095–1131. ISSN: 0033-5533. DOI: 10.1162/qjec.2009.124.3.1095. eprint: https://academic.oup.com/qje/article-pdf/124/3/1095/5377740/124-3-1095.pdf. URL: https://doi.org/10.1162/qjec.2009.124.3.1095.

[42] Guibing Guo. "Resolving data sparsity and cold start in recommender systems". In: *International Conference on User Modeling, Adaptation, and Personalization.* Springer. 2012, pp. 361–364.

[43] F. Maxwell Harper and Joseph A. Konstan. "The MovieLens Datasets: History and Context". In: *ACM Trans. Interact. Intell. Syst.* 5.4 (Dec. 2015). ISSN: 2160-6455. DOI: 10.1145/2827872. URL: https://doi.org/10.1145/2827872.

[44] Martie G Haselton, Daniel Nettle, and Paul W Andrews. "The evolution of cognitive bias". In: *The handbook of evolutionary psychology* (2015), pp. 724–746.

[45] David Haussler. "Quantifying inductive bias: AI learning algorithms and Valiant's learning framework". In: *Artificial Intelligence* 36.2 (1988), pp. 177–221. ISSN: 0004-3702. DOI: https://doi.org/10.1016/0004-3702(88)90002-1. URL: https://www.sciencedirect.com/science/article/pii/0004370288900021.

[46] David Haussler. *Probably approximately correct learning.* University of California, Santa Cruz, Computer Research Laboratory Santa ..., 1990.

[47] James J. Heckman. "Sample Selection Bias as a Specification Error". In: *Econometrica* 47.1 (1979), pp. 153–161. ISSN: 00129682, 14680262. URL: http://www.jstor.org/stable/1912352 (visited on 12/22/2022).

[48] Lucien Heitz, Juliane A. Lischka, Alena Birrer, Bibek Paudel, Suzanne Tolmeijer, Laura Laugwitz, and Abraham Bernstein. "Benefits of Diverse News Recommendations for Democracy: A User Study". In: *Digital Journalism* 10.10 (2022), pp. 1710–1730. DOI: 10.1080/21670811.2021.2021804. eprint: https://doi.org/10.1080/21670811.2021.2021804. URL: https://doi.org/10.1080/21670811.2021.2021804.

[49] Miles Hewstone. "The 'ultimate attribution error'? A review of the literature on intergroup causal attribution". In: *European Journal of Social Psychology* 20.4 (1990), pp. 311–335. DOI: https://doi.org/10.1002/ejsp.2420200404. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/ejsp.2420200404. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/ejsp.2420200404.

[50] R Devon Hjelm, Alex Fedorov, Samuel Lavoie-Marchildon, Karan Grewal, Phil Bachman, Adam Trischler, and Yoshua Bengio. *Learning deep representations by mutual information estimation and maximization.* 2018. DOI: 10.48550/ARXIV.1808.06670. URL: https://arxiv.org/abs/1808.06670.

[51] Jonathan Howard. "Semmelweis reflex". In: *Cognitive Errors and Diagnostic Mistakes.* Springer, 2019, pp. 467–500.

[52] Christopher K. Hsee. "Less is better: when low-value options are valued more highly than high-value options". In: *Journal of Behavioral Decision Making* 11.2 (1998), pp. 107–121. DOI: https://doi.org/10.1002/(SICI)1099-0771(199806)11:2<107::AID-BDM292>3.0.CO;2-Y.

[53] Dietmar Jannach, Markus Zanker, Alexander Felfernig, and Gerhard Friedrich. *Recommender Systems: An Introduction.* Cambridge University Press, 2010. DOI: 10.1017/CBO9780511763113.

[54] Yitong Ji, Aixin Sun, J Zhang, and Chenliang Li. "Do Loyal Users Enjoy Better Recommendations?: Understanding Recommender Accuracy from a Time Perspective". In: *Proceedings of the 2022 ACM SIGIR International Conference on Theory of Information Retrieval* (2022).

[55] Owen Jones. *The racist hijacking of Microsoft's chatbot shows how the internet teems with hate.* Mar. 2016. URL: https://www.theguardian.com/world/2016/mar/29/microsoft-tay-tweets-antisemitic-racism.

[56] JZ from Alibaba Group. *From AUC to GAUC.* URL: https://medium.com/@j.zh/from-auc-to-gauc-928e1c4f1fc9 (visited on 11/21/2022).

[57] Danielle Leah Kehl and Samuel Ari Kessler. "Algorithms in the criminal justice system: Assessing the use of risk assessments in sentencing". In: (2017).

[58] Diederik P Kingma and Max Welling. "Auto-encoding variational bayes". In: *arXiv preprint arXiv:1312.6114* (2013).

[59] Diederik P. Kingma and Jimmy Ba. *Adam: A Method for Stochastic Optimization.* 2014. DOI: 10.48550/ARXIV.1412.6980. URL: https://arxiv.org/abs/1412.6980.

[60] Mark A Kramer. "Nonlinear principal component analysis using autoassociative neural networks". In: *AIChE journal* 37.2 (1991), pp. 233–243.

[61] Sanjay Krishnan, Jay Patel, Michael J. Franklin, and Ken Goldberg. "A Methodology for Learning, Analyzing, and Mitigating Social Influence Bias in Recommender Systems". In: *Proceedings of the 8th ACM Conference on Recommender Systems.* RecSys '14. Foster City, Silicon Valley, California, USA: Association for Computing Machinery, 2014, pp. 137–144. ISBN: 9781450326681. DOI: 10.1145/2645710.2645740. URL: https://doi.org/10.1145/2645710.2645740.

[62] Yi Li and Nuno Vasconcelos. "REPAIR: Removing Representation Bias by Dataset Resampling". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR).* June 2019.

[63] Benjamin Libet. “"Subjective Perception"”. In: *Science* 247.4943 (1990), pp. 727–727. DOI: 10.1126/science.2105530. eprint: https://www.science.org/doi/pdf/10.1126/science.2105530. URL: https://www.science.org/doi/abs/10.1126/science.2105530.

[64] Blerina Lika, Kostas Kolomvatsos, and Stathes Hadjiefthymiades. “Facing the cold start problem in recommender systems”. In: *Expert Systems with Applications* 41.4, Part 2 (2014), pp. 2065–2073. ISSN: 0957-4174. DOI: https://doi.org/10.1016/j.eswa.2013.09.005. URL: https://www.sciencedirect.com/science/article/pii/S0957417413007240.

[65] Chen Lin, Xinyi Liu, Guipeng Xv, and Hui Li. “Mitigating Sentiment Bias for Recommender Systems”. In: SIGIR ’21. Virtual Event, Canada: Association for Computing Machinery, 2021, pp. 31–40. ISBN: 9781450380379. DOI: 10.1145/3404835.3462943. URL: https://doi.org/10.1145/3404835.3462943.

[66] *List of cognitive biases.* Dec. 2022. URL: https://en.wikipedia.org/wiki/List_of_cognitive_biases#cite_note-1.

[67] David C. Liu, Stephanie Rogers, Raymond Shiau, Dmitry Kislyuk, Kevin C. Ma, Zhigang Zhong, Jenny Liu, and Yushi Jing. “Related Pins at Pinterest: The Evolution of a Real-World Recommender System”. In: *Proceedings of the 26th International Conference on World Wide Web Companion.* WWW ’17 Companion. Perth, Australia: International World Wide Web Conferences Steering Committee, 2017, pp. 583–592. ISBN: 9781450349147. DOI: 10.1145/3041021.3054202. URL: https://doi.org/10.1145/3041021.3054202.

[68] Dugang Liu, Pengxiang Cheng, Zhenhua Dong, Xiuqiang He, Weike Pan, and Zhong Ming. “A General Knowledge Distillation Framework for Counterfactual Recommendation via Uniform Data”. In: *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval.* SIGIR ’20. Virtual Event, China: Association for Computing Machinery, 2020, pp. 831–840. ISBN: 9781450380164. DOI: 10.1145/3397271.3401083. URL: https://doi.org/10.1145/3397271.3401083.

[69] Yiming Liu, Xuezhi Cao, and Yong Yu. “Are you influenced by others when rating? Improve rating prediction by conformity modeling”. In: *Proceedings of the 10th ACM conference on recommender systems.* 2016, pp. 269–272.

[70] Zhiyuan Liu, Yankai Lin, and Maosong Sun. *Representation Learning for Natural Language Processing, Springer eBook Collection.* Singapore: Springer Singapore; 2020. ISBN: 9789811555732. DOI: 10.1007/978-981-15-5573-2. URL: https://www.tib.eu/de/suchen/id/TIBKAT%3A1726034682.

[71] Zhuang Liu, Yunpu Ma, Yuanxin Ouyang, and Zhang Xiong. *Contrastive Learning for Recommender System.* 2021. DOI: 10.48550/ARXIV.2101.01317. URL: https://arxiv.org/abs/2101.01317.

[72] Minh-Thang Luong, Hieu Pham, and Christopher D. Manning. *Effective Approaches to Attention-based Neural Machine Translation.* 2015. DOI: 10.48550/ARXIV.1508.04025. URL: https://arxiv.org/abs/1508.04025.

[73] Zhe Ma and Qiang Dong. "Alleviating the unfairness of recommendation by eliminating the conformity bias". In: *Journal of Physics: Conference Series* 2004.1 (Aug. 2021), p. 012008. DOI: 10.1088/1742-6596/2004/1/012008. URL: https://doi.org/10.1088/1742-6596/2004/1/012008.

[74] Diane M Mackie and Scott T Allison. "Group attribution errors and the illusion of group attitude change". In: *Journal of Experimental Social Psychology* 23.6 (1987), pp. 460–480.

[75] Masoud Mansoury, Himan Abdollahpouri, Mykola Pechenizkiy, Bamshad Mobasher, and Robin Burke. "Feedback Loop and Bias Amplification in Recommender Systems". In: *Proceedings of the 29th ACM International Conference on Information &amp; Knowledge Management*. CIKM '20. Virtual Event, Ireland: Association for Computing Machinery, 2020, pp. 2145–2148. ISBN: 9781450368599. DOI: 10.1145/3340531.3412152. URL: https://doi.org/10.1145/3340531.3412152.

[76] Masoud Mansoury, Himan Abdollahpouri, Mykola Pechenizkiy, Bamshad Mobasher, and Robin Burke. "Feedback loop and bias amplification in recommender systems". In: *Proceedings of the 29th ACM international conference on information & knowledge management*. 2020, pp. 2145–2148.

[77] Masoud Mansoury, Himan Abdollahpouri, Mykola Pechenizkiy, Bamshad Mobasher, and Robin Burke. "A Graph-Based Approach for Mitigating Multi-Sided Exposure Bias in Recommender Systems". In: *ACM Trans. Inf. Syst.* 40.2 (Nov. 2021). ISSN: 1046-8188. DOI: 10.1145/3470948. URL: https://doi.org/10.1145/3470948.

[78] Benjamin Marlin, Richard S. Zemel, Sam Roweis, and Malcolm Slaney. *Collaborative Filtering and the Missing at Random Assumption*. 2012. DOI: 10.48550/ARXIV.1206.5267. URL: https://arxiv.org/abs/1206.5267.

[79] Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan. "A Survey on Bias and Fairness in Machine Learning". In: *ACM Comput. Surv.* 54.6 (July 2021). ISSN: 0360-0300. DOI: 10.1145/3457607. URL: https://doi.org/10.1145/3457607.

[80] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. *Efficient Estimation of Word Representations in Vector Space*. 2013. DOI: 10.48550/ARXIV.1301.3781. URL: https://arxiv.org/abs/1301.3781.

[81] Tomas Mikolov, Quoc V. Le, and Ilya Sutskever. *Exploiting Similarities among Languages for Machine Translation*. 2013. DOI: 10.48550/ARXIV.1309.4168. URL: https://arxiv.org/abs/1309.4168.

[82] Silvia Milano, Mariarosaria Taddeo, and Luciano Floridi. "Recommender systems and their ethical challenges". In: *Ai & Society* 35.4 (2020), pp. 957–967.

[83] Stanley Milgram and Christian Gudehus. *Obedience to authority*. 1978.

[84] Tom M Mitchell. *The need for biases in learning generalizations*. Department of Computer Science, Laboratory for Computer Science Research ..., 1980.

[85] Sirous Mobini, Shirley Reynolds, and Bundy Mackintosh. "Clinical implications of cognitive bias modification for interpretative biases in social anxiety: An integrative literature review". In: *Cognitive Therapy and Research* 37.1 (2013), pp. 173–182.

[86] Paromita Nitu, Joseph Coelho, and Praveen Madiraju. “Improvising personalized travel recommendation system with recency effects”. In: *Big Data Mining and Analytics* 4.3 (2021), pp. 139–154. DOI: 10.26599/BDMA.2020.9020026.

[87] Eoin D O’Sullivan and SJ Schofield. “Cognitive bias in clinical medicine”. In: *Journal of the Royal College of Physicians of Edinburgh* 48.3 (2018), pp. 225–232.

[88] Alexandra Olteanu, Carlos Castillo, Fernando Diaz, and Emre Kıcıman. “Social data: Biases, methodological pitfalls, and ethical boundaries”. In: *Frontiers in Big Data* 2 (2019), p. 13.

[89] Eben Otuteye and Mohammad Siddiquee. “Overcoming Cognitive Biases: A Heuristic for Making Value Investing Decisions”. In: *Journal of Behavioral Finance* 16.2 (2015), pp. 140–149. DOI: 10.1080/15427560.2015.1034859. eprint: https://doi.org/10.1080/15427560.2015.1034859. URL: https://doi.org/10.1080/15427560.2015.1034859.

[90] Zohreh Ovaisi, Ragib Ahsan, Yifan Zhang, Kathryn Vasilaky, and Elena Zheleva. “Correcting for Selection Bias in Learning-to-Rank Systems”. In: *Proceedings of The Web Conference 2020*. WWW ’20. Taipei, Taiwan: Association for Computing Machinery, 2020, pp. 1863–1873. ISBN: 9781450370233. DOI: 10.1145/3366423.3380255. URL: https://doi.org/10.1145/3366423.3380255.

[91] Anish Patankar, Joy Bose, and Harshit Khanna. “A Bias Aware News Recommendation System”. In: *2019 IEEE 13th International Conference on Semantic Computing (ICSC)*. 2019, pp. 232–238. DOI: 10.1109/ICOSC.2019.8665610.

[92] Jeffrey Pennington, Richard Socher, and Christopher Manning. “GloVe: Global Vectors for Word Representation”. In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Doha, Qatar: Association for Computational Linguistics, Oct. 2014, pp. 1532–1543. DOI: 10.3115/v1/D14-1162. URL: https://aclanthology.org/D14-1162.

[93] Abigail Player, Georgina Randsley de Moura, Ana C. Leite, Dominic Abrams, and Fatima Tresh. “Overlooked Leadership Potential: The Preference for Leadership Potential in Job Candidates Who Are Men vs. Women”. In: *Frontiers in Psychology* 10 (2019). ISSN: 1664-1078. DOI: 10.3389/fpsyg.2019.00755. URL: https://www.frontiersin.org/articles/10.3389/fpsyg.2019.00755.

[94] Tao Qi, Fangzhao Wu, Chuhan Wu, Peijie Sun, Le Wu, Xiting Wang, Yongfeng Huang, and Xing Xie. *ProFairRec: Provider Fairness-aware News Recommendation.* 2022. DOI: 10.48550/ARXIV.2204.04724. URL: https://arxiv.org/abs/2204.04724.

[95] Junyang Rao, Aixia Jia, Yansong Feng, and Dongyan Zhao. “Taxonomy Based Personalized News Recommendation: Novelty and Diversity”. In: *Web Information Systems Engineering – WISE 2013*. Ed. by Xuemin Lin, Yannis Manolopoulos, Divesh Srivastava, and Guangyan Huang. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 209–218. ISBN: 978-3-642-41230-1.

[96] Shaina Raza and Chen Ding. “A survey on news recommender system - Dealing with timeliness, dynamic user interest and content quality, and effects of recommendation on news readers”. In: *CoRR* abs/2009.04964 (2020). arXiv: 2009.04964. URL: https://arxiv.org/abs/2009.04964.

[97] P. Reichl, S. Egger, R. Schatz, and A. D'Alconzo. "The Logarithmic Nature of QoE and the Role of the Weber-Fechner Law in QoE Assessment". In: *2010 IEEE International Conference on Communications.* 2010, pp. 1–5. DOI: `10.1109/ICC.2010.5501894`.

[98] Lee Ross and Constance Stillinger. "Barriers to Conflict Resolution". In: *Negotiation Journal* 7.4 (1991), pp. 389–404. DOI: `https://doi.org/10.1111/j.1571-9979.1991.tb00634.x`. eprint: `https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1571-9979.1991.tb00634.x`. URL: `https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1571-9979.1991.tb00634.x`.

[99] Navin Sabharwal and Amit Agrawal. *Hands-on Question Answering Systems with BERT : Applications in Neural Networks and Natural Language Processing, Springer eBook Collection.* New York: Apress; 2021. ISBN: 9781484266649. DOI: `10.1007/978-1-4842-6664-9`. URL: `%5Clinkhttps%7Bhttps://www.tib.eu/de/suchen/id/TIBKAT%5C%3A1746356189%7D`.

[100] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. "DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter". In: *ArXiv* abs/1910.01108 (2019).

[101] Gustavo Saposnik, Donald Redelmeier, Christian C Ruff, and Philippe N Tobler. "Cognitive biases associated with medical decisions: a systematic review". In: *BMC medical informatics and decision making* 16.1 (2016), pp. 1–14.

[102] D.L. Schacter, D.T. Gilbert, and D.M. Wegner. *Psychology.* Worth Publishers, 2010. ISBN: 9781429237192. URL: `https://books.google.de/books?id=emAyzTNy1cUC`.

[103] Tobias Schnabel, Adith Swaminathan, Ashudeep Singh, Navin Chandak, and Thorsten Joachims. "Recommendations as Treatments: Debiasing Learning and Evaluation". In: *Proceedings of The 33rd International Conference on Machine Learning.* Ed. by Maria Florina Balcan and Kilian Q. Weinberger. Vol. 48. Proceedings of Machine Learning Research. New York, New York, USA: PMLR, June 2016, pp. 1670–1679. URL: `https://proceedings.mlr.press/v48/schnabel16.html`.

[104] E. H. Simpson. "The Interpretation of Interaction in Contingency Tables". In: *Journal of the Royal Statistical Society: Series B (Methodological)* 13.2 (1951), pp. 238–241. DOI: `https://doi.org/10.1111/j.2517-6161.1951.tb00088.x`. eprint: `https://rss.onlinelibrary.wiley.com/doi/pdf/10.1111/j.2517-6161.1951.tb00088.x`. URL: `https://rss.onlinelibrary.wiley.com/doi/abs/10.1111/j.2517-6161.1951.tb00088.x`.

[105] Stuart Soroka. "Why do we pay more attention to negative news than to positive news?" In: *USApp–American Politics and Policy Blog* (2015).

[106] Stuart Soroka, Patrick Fournier, and Lilach Nir. "Cross-national evidence of a negativity bias in psychophysiological reactions to news". In: *Proceedings of the National Academy of Sciences* 116.38 (2019), pp. 18888–18892.

[107] Elias K Spanakis and Sherita Hill Golden. "Race/ethnic difference in diabetes and diabetic complications". In: *Current diabetes reports* 13.6 (2013), pp. 814–823.

[108] Harini Suresh and John V Guttag. "A framework for understanding unintended consequences of machine learning". In: *arXiv preprint arXiv:1901.10002* 2 (2019), p. 8.

[109] Amos Tversky and Daniel Kahneman. "The Framing of Decisions and the Psychology of Choice". In: *Science* 211.4481 (1981), pp. 453–458. DOI: `10.1126/science.7455683`. eprint: `https://www.science.org/doi/pdf/10.1126/science.7455683`. URL: `https://www.science.org/doi/abs/10.1126/science.7455683`.

[110] UNU-WIDER. *The Global Distribution of Household Wealth.* 2015. URL: `https://www.wider.unu.edu/publication/global-distribution-household-wealth` (visited on 10/17/2022).

[111] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. "Attention is all you need". In: *Advances in neural information processing systems* 30 (2017).

[112] Maksims Volkovs, Guang Wei Yu, and Tomi Poutanen. "Content-based neighbor models for cold start in recommender systems". In: *Proceedings of the Recommender Systems Challenge 2017.* 2017, pp. 1–6.

[113] Hongwei Wang, Fuzheng Zhang, Xing Xie, and Minyi Guo. *DKN: Deep Knowledge-Aware Network for News Recommendation.* 2018. DOI: `10.48550/ARXIV.1801.08284`. URL: `https://arxiv.org/abs/1801.08284`.

[114] Silvana Weber and Elena Knorr. "Kognitive Verzerrungen und die Irrationalität des Denkens". In: *Die Psychologie des Postfaktischen: Über Fake News, „Lügenpresse", Clickbait & Co.* Ed. by Markus Appel. Berlin, Heidelberg: Springer Berlin Heidelberg, 2020, pp. 103–115. ISBN: 978-3-662-58695-2. DOI: `10.1007/978-3-662-58695-2_10`. URL: `https://doi.org/10.1007/978-3-662-58695-2_10`.

[115] *Weights and Biases.* URL: `https://wandb.ai/site`.

[116] Dale F Whelehan, Kevin C Conlon, and Paul F Ridgway. "Medicine and heuristics: cognitive biases and medical decision-making". In: *Irish Journal of Medical Science (1971-)* 189.4 (2020), pp. 1477–1484.

[117] Markus Antonius Wirtz. *Simpson-Paradoxon im Dorsch Lexikon der Psychologie.* 2021. URL: `https://dorsch.hogrefe.com/stichwort/simpson-paradoxon`.

[118] Chuhan Wu, Fangzhao Wu, Mingxiao An, Jianqiang Huang, Yongfeng Huang, and Xing Xie. *Neural News Recommendation with Attentive Multi-View Learning.* 2019. DOI: `10.48550/ARXIV.1907.05576`. URL: `https://arxiv.org/abs/1907.05576`.

[119] Chuhan Wu, Fangzhao Wu, Mingxiao An, Jianqiang Huang, Yongfeng Huang, and Xing Xie. "NPA: Neural News Recommendation with Personalized Attention". In: *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining.* KDD '19. Anchorage, AK, USA: Association for Computing Machinery, 2019, pp. 2576–2584. ISBN: 9781450362016. DOI: `10.1145/3292500.3330665`. URL: `https://doi.org/10.1145/3292500.3330665`.

[120] Chuhan Wu, Fangzhao Wu, Suyu Ge, Tao Qi, Yongfeng Huang, and Xing Xie. "Neural News Recommendation with Multi-Head Self-Attention". In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP).* Hong Kong, China: Association for Computational Linguistics, Nov. 2019, pp. 6389–6394. DOI: `10.18653/v1/D19-1671`. URL: `https://aclanthology.org/D19-1671`.

[121] Chuhan Wu, Fangzhao Wu, and Yongfeng Huang. *DebiasGAN: Eliminating Position Bias in News Recommendation with Adversarial Learning.* 2021. DOI: 10.48550/ARXIV.2106.06258. URL: https://arxiv.org/abs/2106.06258.

[122] Chuhan Wu, Fangzhao Wu, Tao Qi, and Yongfeng Huang. "SentiRec: Sentiment Diversity-aware Neural News Recommendation". In: *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing.* Suzhou, China: Association for Computational Linguistics, Dec. 2020, pp. 44–53. URL: https://aclanthology.org/2020.aacl-main.6.

[123] Chuhan Wu, Fangzhao Wu, Tao Qi, and Yongfeng Huang. "Empowering News Recommendation with Pre-Trained Language Models". In: *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval.* SIGIR '21. Virtual Event, Canada: Association for Computing Machinery, 2021, pp. 1652–1656. ISBN: 9781450380379. DOI: 10.1145/3404835.3463069. URL: https://doi.org/10.1145/3404835.3463069.

[124] Chuhan Wu, Fangzhao Wu, Tao Qi, Yongfeng Huang, and Xing Xie. *Fastformer: Additive Attention Can Be All You Need.* 2021. DOI: 10.48550/ARXIV.2108.09084. URL: https://arxiv.org/abs/2108.09084.

[125] Chuhan Wu, Fangzhao Wu, Xiting Wang, Yongfeng Huang, and Xing Xie. "Fairness-aware news recommendation with decomposed adversarial learning". In: *Proceedings of the AAAI Conference on Artificial Intelligence.* Vol. 35. 5. 2021, pp. 4462–4469.

[126] Fangzhao Wu, Ying Qiao, Jiun-Hung Chen, Chuhan Wu, Tao Qi, Jianxun Lian, Danyang Liu, Xing Xie, Jianfeng Gao, Winnie Wu, et al. "Mind: A large-scale dataset for news recommendation". In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics.* 2020, pp. 3597–3606.

[127] Shuyuan Xu, Juntao Tan, Shelby Heinecke, Jia Li, and Yongfeng Zhang. "Deconfounded Causal Collaborative Filtering". In: *arXiv preprint arXiv:2110.07122* (2021).

[128] Taha Yasseri and Jannie Reher. "Fooled by facts: quantifying anchoring bias through a large-scale experiment". In: *Journal of Computational Social Science* 5.1 (2022), pp. 1001–1021. DOI: 10.1007/s42001-021-00158-0. URL: https://doi.org/10.1007/s42001-021-00158-0.

[129] Zhi Zheng, Zhaopeng Qiu, Tong Xu, Xian Wu, Xiangyu Zhao, Enhong Chen, and Hui Xiong. "CBR: Context Bias Aware Recommendation for Debiasing User Modeling and Click Prediction". In: *Proceedings of the ACM Web Conference 2022.* WWW '22. Virtual Event, Lyon, France: Association for Computing Machinery, 2022, pp. 2268–2276. ISBN: 9781450390965. DOI: 10.1145/3485447.3512099. URL: https://doi.org/10.1145/3485447.3512099.

A Appendix: Foundations

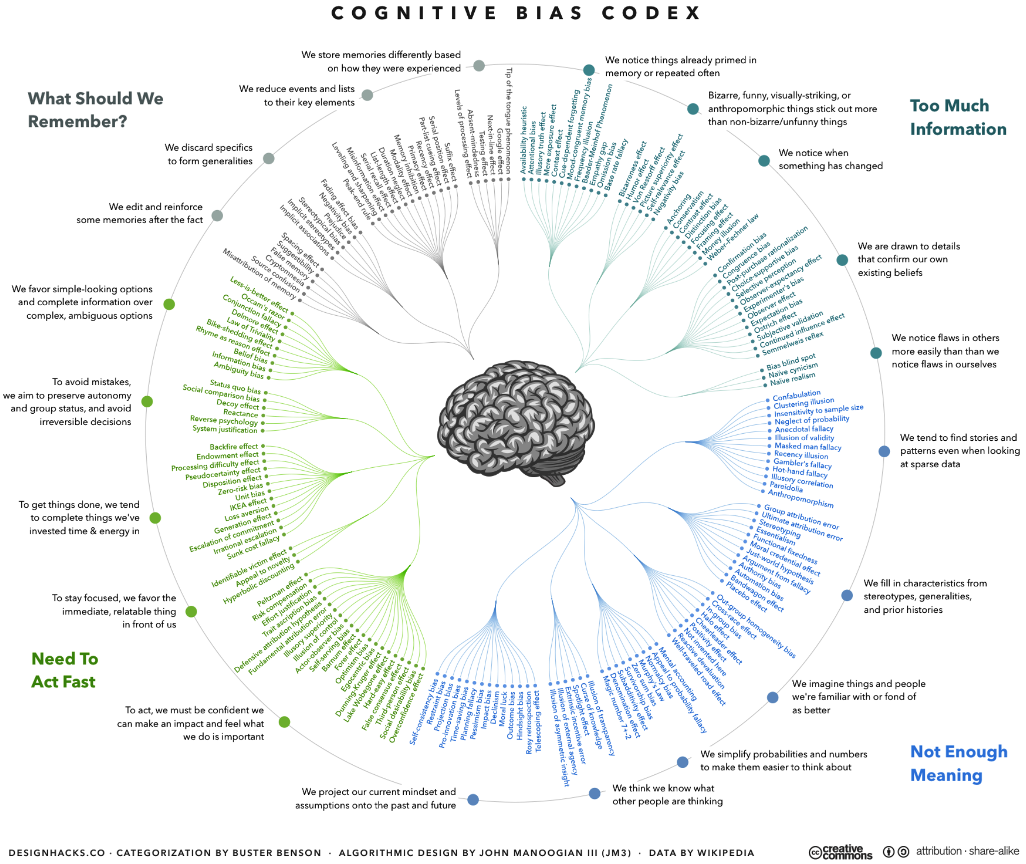


Figure A.1: Categorized cognitive biases (design: John Manoogian III[1])

B Appendix: Experimental Setup

Keys	Description
Label	The WikiData[1] knowledge graph entity name
Type	The WikiData type of entity
WikidataId	The WikiData ID
Confidence	The confidence of the entity linking
OccuranceOffsets	The entity offset in the text according to characters
SurfaceForms	The actual entity names

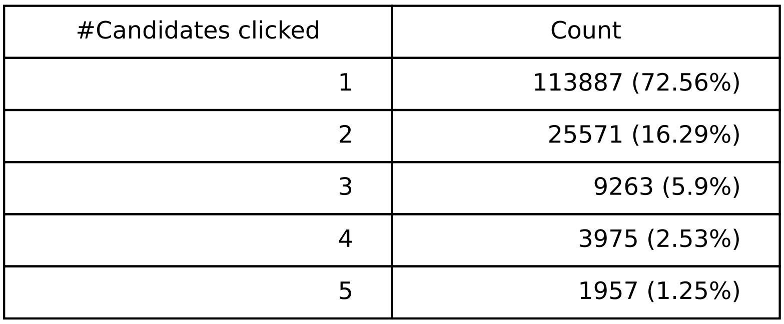
Table B.1: Keys and their description in the entities of the MIND dataset (Modified from MSNews)

|News|	161,013
|Different categories|	20
|Entities|	3,299,687
Average title len.	11.52
Average body len.	585.05
Average abstract len.	43
|Users|	1,000,000
|Impressions|	15,777,377
|Clicks in behaviors|	24,155,470

Table B.2: Statistics for the MIND-large (taken from the original paper by Wu et al. [120])

Category	Number of subcategories	Most frequent subcategory
News	33	NewsUS
Sports	31	Football_NFL
Finance	32	FinanceNews
Food and Drink	16	NewsTrends
Lifestyle	47	LifestyleBuzz
Travel	14	NewsTrends
Video	14	News
Weather	2	WeatherTopStories
Health	20	Medical
Autos	24	AutosNews
TV	10	TVNews
Music	11	MusicNews
Movies	7	MovieNews
Entertainment	15	News
Kids	5	Animals
Middle East	1	MiddleEast-Top-Stories
North America	1	NorthAmerica-Video

Table B.3: Categories with their Subcategory count



#Candidates clicked	Count
1	113887 (72.56%)
2	25571 (16.29%)
3	9263 (5.9%)
4	3975 (2.53%)
5	1957 (1.25%)

Figure B.1: Number and percentage of clicked news in impressions

C Appendix: Experiments

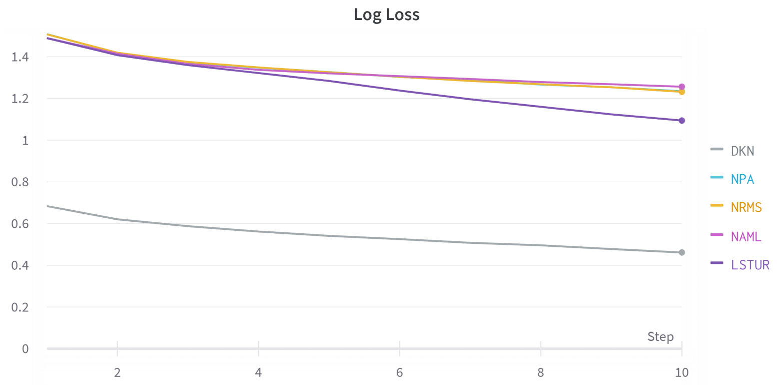


Figure C.1: Log loss for 10 epochs of training

Parameter/Model	*NRMS* [120]	*LSTUR* [9]	*NAML* [118]	*NPA* [119]	*DKN* [113]
History size	50	50	50	50	50
Word emb. dimension	300	300	300	300	100
Entity emb. size	-	-	-	-	100
Context emb. size	-	-	-	-	100
User emb. size	50	50	50	100	100
Vert emb. size	-	100	100	-	-
Subvert emb. size	-	100	100	-	-
Max. title length (in words)	30	30	30	10	-
Max. body length (in words)	-	-	50	-	-
Attention layer dimension	200	200	200	200	200

Table C.1: Model hyperparameters referring to the proposed hyperparameters in the corresponding papers.

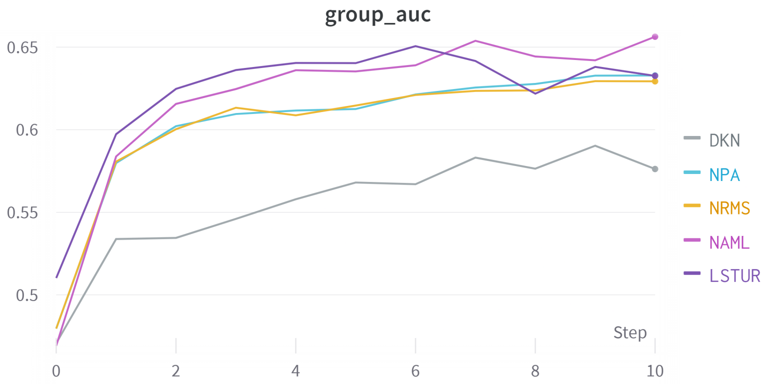


Figure C.2: Group AUC for 10 epochs of training

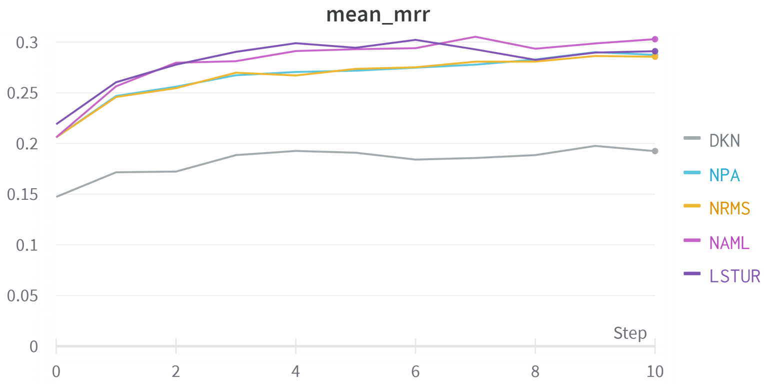


Figure C.3: Mean MRR for 10 epochs of training

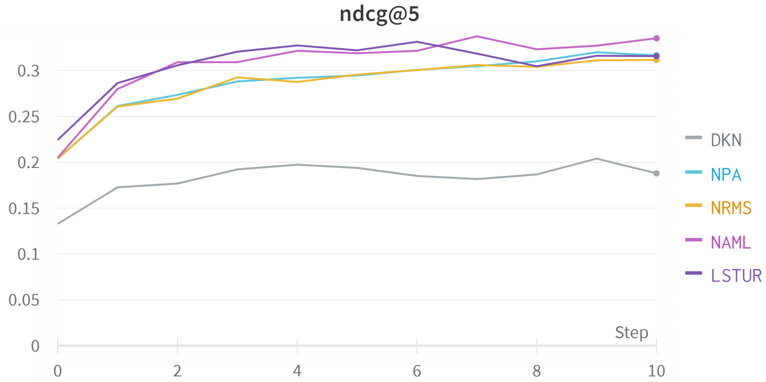


Figure C.4: nDCG@5 for 10 epochs of training

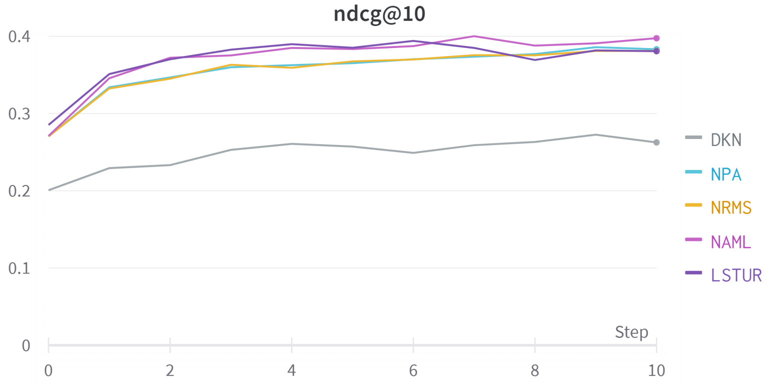


Figure C.5: nDCG@10 for 10 epochs of training

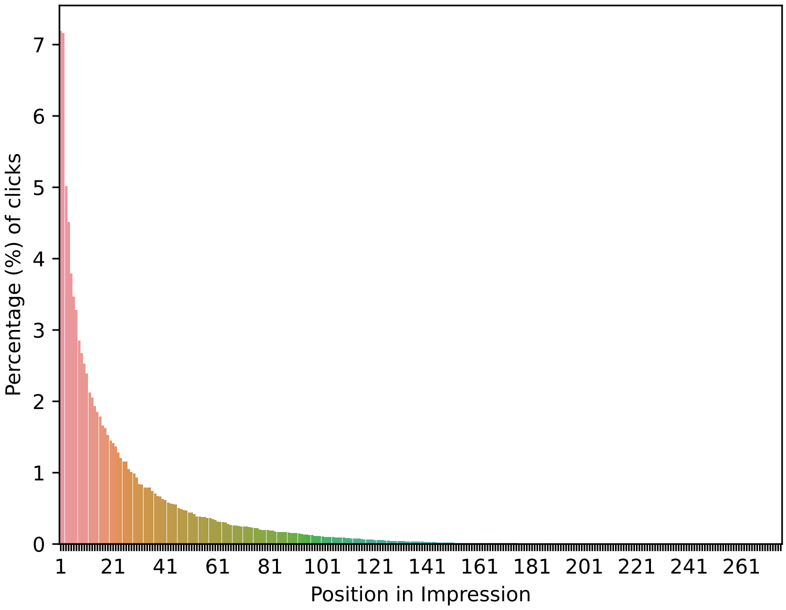


Figure C.6: Percentages of each clicked position in MIND small.