UMUAI manuscript No.
(will be inserted by the editor)

Learning to Recommend Similar Items from Human Judgements

Christoph Trattner · Dietmar Jannach

Received: date / Accepted: date

Abstract Similar item recommendations—a common feature of many websites—point users to other interesting objects given a currently inspected item. A common way of computing such recommendations is to use a *similarity function*, which expresses how much alike two given objects are. Such similarity functions are usually designed based on the specifics of the given application domain.

In this work, we explore how such functions can be learned from human judgements of similarities between objects, using two domains of "quality-and-taste"—cooking recipe and movie recommendation—as guiding scenarios. In our approach, we first collect a few thousand pairwise similarity assessments with the help of crowdworkers. Using this data, we then train different machine learning models that can be used as similarity functions to compare objects. Offline analyses reveal for both application domains that models that combine *different types* of item characteristics are the best predictors for human-perceived similarity.

To further validate the usefulness of the learned models, we conducted additional user studies. In these studies, we exposed participants to similar item recommendations using a set of models that were trained with different feature subsets. The results showed that the combined models that exhibited the best offline prediction performance led to the highest user-perceived similarity, but also to recommendations that were considered useful by the participants, thus confirming the feasibility of our approach.

Keywords Similar Item Recommendations; Similarity Measures; Content-based Recommender Systems; User Studies

C. Trattner
University of Bergen, Norway
E-mail: christoph.trattner@uib.no

D. Jannach
University of Klagenfurt, Austria
E-mail: dietmar.jannach@aau.at

1 Introduction

The recommendation of similar items in the context of a currently inspected object is a common feature of many modern online services. Such recommendations, which are often presented under a label like "More like this", can be found in a variety of domains, including e-commerce shops, music and media streaming services, or news sites. From the perspective of their utility, this type of recommendations can serve different purposes [26]. Similar item recommendations can, for example, represent *alternatives* for a given item in an e-commerce shop and thereby help users to understand the space of options. Alternatively, recommending similar items can also support the *discovery* of new items or entire item categories. A typical example is the recommendation of similar artists on music streaming sites.

Technically, similar item recommendations are usually based on a function, which expresses the similarity between two given objects and which is used to rank the recommendable items. Usually, the set of aspects to consider in such a similarity function is dependent on a particular application. In some domains, like when recommending electronic devices, the decision might be comparably easy, as objectively measurable characteristics like the item's size or price can be compared. In domains like books, movies, or food, however, it might not be immediately clear how to determine the similarity between two objects.

The design of a similarity function can be approached in different ways. One option is to construct a function based on expert knowledge or experience [63]. To validate and compare such manually engineered functions, one can run field tests (A/B tests) as in [6]. Such field tests can, however, be costly and one cannot be sure that all relevant parts of the design space were explored. An alternative is to collect human similarity assessments for a subset of the available items, which can then be used to analyze which item characteristics are important for users with respect to their similarity perception [3, 28, 66].

With this work, we contribute to the line of research based on human judgements. We, however, go beyond the mere analysis of potential factors that may determine the users' quality perception and propose a two-step approach. In the first step, human similarity judgements are collected, which can be used to train different machine learning models to predict the similarity of two items as perceived by humans. In the second step, we conduct a follow-up user study to validate that the model that performed best in the offline analysis also leads to the highest similarity perception by end users. This validation step, which is often missing in previous research, is particularly important as we cannot always assume that a correspondence between offline results and user perception exists. In fact, in a number of past studies—in particular works that compare offline accuracy with the users' quality perception—such a correspondence could not be established [4, 9, 16, 35, 46]. Furthermore, since most similar items are not necessarily the most useful recommendations, we investigated different aspects of usefulness of the resulting recommendations in the online study.

We selected two application domains to test our general approach of learning and validating a similarity function: movie recommendation and cooking recipe recommendation. We chose the movie domain to be able to compare our insights with the results presented in [66]. We furthermore consider the cooking domain for two main reasons. First, recipes are also a comparably *subjective* domain, where it

is not immediately obvious which factors determine the users' similarity perception. Second, the domain of food and recipe recommendation has been subject to increased interest in recent years [14, 55, 65], e.g., due to increasing information overload that users face on online sites that provide information for thousands of recipes.

The paper is organized as follows. In Section 2, we review existing works on (i) using human judgements in the context of similar item recommendation and (ii) common features that are used in the domains of movie and cooking recipe recommendation. The detailed research questions and an overview of the conducted studies are provided in Section 3. Section 4 and Section 5 describe the technical details of how we learned the similarity functions; Section 6 reports the findings from the validation studies. The paper concludes with a conclusions section (Section 8) and a discussion of the implications, limitations and potential future work of our research (Section 7). An appendix at the end of the paper includes further findings and details not presented in the main text of this article.

2 Previous Work

In this section, we first review existing computer science[1] research that relies on human judgements to determine item similarities in the context of recommendation and retrieval problems. Then, we discuss approaches to recipe recommendation, with a focus on content-based approaches and the corresponding similarity measures. Finally, we briefly review commonly used item features in the movie domain.

2.1 Determining Item Similarity based on Human Judgements

In the context of information retrieval and in particular for recommender systems, similarity functions are used for different purposes. In content-based recommenders, they serve as a basis to assess the match of a given item with the user's past preferences [34]. In the case of related item recommendations, similarities estimates are often part of a number of components that determine the item ranking process [6, 68]. Finally, similarity functions are also a central element in approaches that aim to obtain higher diversity of a list of recommendations, e.g., by looking at pairwise item similarities [61, 69]. The design of a similarity function, as mentioned, is, however, often based on domain expertise and intuition [63] or on what types of information are actually available. Only in a few studies, as discussed next, human judgements serve as a basis when designing such a function.

2.1.1 Information Retrieval Scenarios

Similarity functions play a prominent role in music information retrieval, where typical tasks include the identification of similar tracks or artists. The goal of

[1] Earlier work discussing the concept of similarities between objects from a psychological perspective can be, for example, found in [60]. In their work, the authors argue that human judgement of similarity is not only feature-based, as is assumed in our work. We agree with this view, and see the exploration of this topic as a promising area for future work.

the work by Ellis et al. [13], for example, was to investigate to what extent a set of given similarity metrics that are on based musical features correspond to the similarity perception by users. For that purpose, they collected a large number of artist similarity judgements through a web interface, which they then used for their subsequent analysis. Similar to their work, we use human judgements as a basis for our research. Our goal is, however, not only to compare similarity functions but to automatically learn such functions from the collected data, which we then validate in a recommendation context.

The authors of [3] ran a small (N=10) laboratory study to compare the similarity ranking of songs by users with the ranking obtained by their proposed similarity metric. The general idea of the validation was similar to our approach. In our work, we, however, compare a number of automatically learned measures and rely on a much larger set of study participants.

To assess the reliability of crowdsourced human judgements, as done in our approach, Lee [33] collected human similarity judgements for music through Amazon's Mechanical Turk platform. A comparison of the collected data with an existing ground truth dataset indicates that crowdsourcing can be considered a reliable source for evaluations.[2]

2.1.2 Recommendation Scenarios

In the realm of recommender systems, Wang et al. [63] recently explored the use of human similarity judgements when building a content-based approach. As a basis for their method, they used an item similarity dataset that was collected in the context of a previous study for the movie domain [8]. They used linear regression to compute importance weights for the different features of a movie, e.g., genre, writers etc. A user study was then conducted to compare the performance of a collaborative filtering (CF) technique and two versions of a content-based technique. The user study (N=79) showed that the content-based method—when the weights were determined using human judgements—was preferred over the alternative content-based approach. The best quality perception was, however, observed for the CF technique. The work shares some similarities with our work, e.g., that we learn importance weights from the collected data. Different from their work, however, we focus on the problem of *non-personalized* similar item recommendations in the context of a reference item and not on personalized recommendations based on long-term preferences.

Most recently, Yao and Harper investigated different facets of item similarity in the movie domain [66]. They collected over 23,000 human judgements for movie pairs through an online study. In this study, they asked the participants to what extent the movies are similar and to what extent they would recommend the second movie, given that someone likes the first. Based on that data they then measured if different strategies for similar item recommendation were able to match these user assessments. An additional user survey provided further insights on the role and value of similar item recommendations. One of the findings was that item similarity and user relevance can represent a trade-off.

[2] Stability and reliability aspects of human judgements in the music domain are also discussed in [28].

Our work is similar to the work of Yao and Harper in that we explore the capability of different algorithms to approximate the users' similarity perception. Differently from their work, we, however, do not evaluate different existing approaches, but automatically learn the importance weights of different item features from the human judgement data. Furthermore, going beyond the work in [66], we conduct additional user studies to validate that the method that works best in the offline evaluation also leads to a high similarity perception by the users. Differently from [66], the results of our validation studies indicate that high levels of similarity not necessarily lead to a lower perceived usefulness of the recommendations for the considered domain.

2.2 Features used in Recipe Recommendation

The problem of recipe recommendation, as mentioned, has attracted increased interest in recent years, see [55] and [56] for recent surveys. In the following, we will briefly review which types of recipe characteristics (features) have been used in the literature to train ranking and prediction models. Our subsequent approach to learn a similarity function (see Section 4), will be based on some of the most important features used in the literature.

In one of the earlier works on the topic [15], the authors propose a recommendation method that is based on the *ingredients* of the recipes. Their idea is to recommend recipes that the target user has rated positively and which contain similar ingredients. In their approach, cosine similarity was used to compare recipes. In a later work, [20] also considered recipes that were rated negatively by users and correspondingly reduced the recommendation scores of recipes with similar ingredients. In addition to the ingredients, the authors also considered *nutrition* information (e.g., calories or fat content) as features in the similarity computations. An even more sophisticated ingredient-based approach was proposed in [53], where the goal was to automatically determine relationships between ingredients using a large pool of recipes that were harvested from *allrecipes.com*. Their experiments show that using such "networks" of ingredients can lead to more accurate predictions of food choices than relying only on ingredient lists, cooking method or style. An ingredient-based entropy metric to derive food networks was lately also used in [29]. In our work, we also rely on ingredient information as one main factor in our models.

The use of recipe *images* in the recommendation process was for example explored in [65]. Since food decisions are often visually driven [37], the authors propose to automatically extrapolate important visual aspects of food images. Technically, they rely on Convolutional Neural Networks (CNN) for feature learning and—similar to our work—use embeddings and cosine similarity when computing the relatedness of the items. The experiments show that such a visual recommendation approach works remarkably well. Low-level image as well as *title* features were also used in [14] for predicting food preference of users. In their work, the authors propose a method to replace unhealthy recipes with more healthy variations. Similar to our work, they use brightness, colorfulness, sharpness, and title words as features in the recommendation process. For the computation of the similarity of two recipes, they rely on ingredients and use cosine similarity as a measure.

The cooking *directions* were used as features in [64], where the authors proposed a similarity measurement for recipes based on the preparation steps. Similarly, we use the cooking directions as a feature in our work. Finally, the recipe *type* was identified as an important feature that determines the similarity for users in [43]. In their study, the type was in fact the strongest predictor. In our work, we therefore focus our analysis to one category of recipes, "main dishes".

Overall, we will use a variety of features from the literature in our work to learn a similarity function. Additional features, like nutrition information, can easily be integrated into our general learning and validation approach in case such data is available. Note that while we use similar features, the goal of our work is different from most previous work. Most of the discussed works in this section aim at predicting the relevance of certain recipes for individuals or at predicting the popularity of a recipe [59], whereas our goal is to learn a function for similar item recommendations.

2.3 Features used in Movie Recommendation

The literature on movie recommendation algorithms is rich and a variety of movie features were considered over the years in the context of content-based and hybrid recommendation techniques.

The probably most frequently used features in the literature are the movie title, genre, release year, plot summaries, actor, and director information. Some of these features, in particular the genre, are also contained in the popular MovieLens datasets. Several other features are available on websites such as IMDb. In our study for the movie domain, we also include several of these features. For some of the features, in particular for the genre and the plot descriptions, it is possible to assess their similarity in different ways. In our study, we therefore, for example, compare the plot descriptions in two ways, (a) based on a TF-IDF encoding and (b) by applying Latent Dirichlet Allocation (LDA).

User-generated content represents another form of metadata that is commonly used both for recommendation and subsequent explanation. User-provided tags have for example been explored for improving recommendation accuracy in [17, 49]. Furthermore, tags turned out to be a very helpful means for explaining recommendations, e.g., in the form of "tagsplanations" [62] or as tag clouds [18]. Computing similarities through the "tag genome" also turned out to be a good predictor for user-perceived similarity in the study by Yao and Harper [66]. Therefore, we include user-provided tags also in our study in the movie domain as well. Note that on *allrecipes.com*, the site from which we retrieved our recipe dataset, no user-provided tags are available. We therefore made experiments including such tags only for the movie domain.

Consumer reviews represent another form of user-generated content that has been explored in content-based recommendation approaches [19]. In our study, we did not include user reviews as features as reading and assessing such reviews would lead to a higher level of cognitive effort by the participants. For the same reason, we do not include semantic information about the movies, which has been explored in the literature as well using, e.g., Linked Open Data or DBPedia [38, 41].

Using visual features of the movies themselves (or their trailers) has been explored recently, e.g., in [10]. In the context of our study, asking the participants

to watch trailers would have been too time-consuming. However, we assume that also the cover images of the movies could be a predictor of user-perceived similarity, since cover artwork in many cases is in some form representative of parts of a movie's content or shows an important scene. Therefore, we consider different ways of computing the similarity of two items based on the visual features of the movie cover in our study.

2.4 Summary of Previous Work and Key Differences

Our review of existing work in the computer science literature shows that the similar item recommendation problem has been studied from different angles and for different domains (most prominently in the music domain). Different features have been exploited as well, but limited work exists that reveals (i) how different types of content features and similarity metrics *compare* to each other and (ii) how they are actually *perceived* by human evaluators.

Datasets that contain human similarity judgements were previously collected through lab studies or crowd-sourced studies, for example, in [3], [33] or [66]. These datasets were, however, mainly used to understand what makes two items similar in a given domain and the authors only focused on evaluating already *existing* approaches for similar item recommendations.

Compared to these papers we, in contrast, aim to learn a similarity function in a structured and systematic way, based on content features derived from titles, images, etc. This allows us to understand which features and metrics correlate the most with human estimates. Differently from previous studies, we also ask participants which content cues they used to assess the similarity of two items, and we contrast this with the results of a correlation analysis based on content features.

Finally, what is missing in previous work is the validation of the learned function in a recommendation scenario. The results of our validation study ultimately shows that bootstrapping a similar item recommendation component based on automatically extracted content features is a viable approach.

3 Research Questions and Experiment Overview

The main contribution of our work is a proposal for a structured approach to learn similarity functions for similar item recommendation and to assess their effectiveness and usefulness in a recommender scenario. We performed different user studies as well as offline evaluations for this purpose in the recipe and movie application domain.

In the first set of studies (*Study 1a* and *Study 1b*), we collected sets of pairwise similarity judgements from crowdworkers for the recipe and movie domains, respectively. Based on an analysis of the data, we built a number of prediction models that use different sets of item features for each domain. We then evaluated these models both through offline experimentation and through additional user studies (*Study 2a* and *Study 2b*). In this second set of user studies, our goal was to assess the similarity perception of users when they were shown recommendations based on different prediction models. Our specific research questions are as follows:

- *RQ1.* Which *types* of features and which *specific* features determine the similarity between items as perceived by users? Knowledge about the importance of different factors is a prerequisite to design a suitable similarity function, and we use the data obtained from *Study 1a* and *Study 1b* to answer these questions.
- *RQ2.* Which specific combination of features is suited to predicting user-perceived similarity levels? We use an offline experiment based on the data of *Study 1a* and *Study 1b* in which we train different machine learning models and compare their prediction accuracy.
- *RQ3.* Do models with higher prediction accuracy lead to a higher perceived item similarity? We answer this question with *Study 2a* and *Study 2b*.
- *RQ4.* How do users assess the *usefulness* of recommendations that are based on different similarity functions? We also address this question with *Study 2a* and *Study 2b*.

For the studies in the recipe domain, we use recipe information that we harvested from the online platform *allrecipes.com*, which is one of the most popular online recipe websites [11]. Overall, we retrieved 60,983 recipes published by 25,037 users between the years 2000 and 2015. Besides the recipe titles, ingredient lists (including amounts in grams per 100g of a recipe)[3], and the cooking directions, we also downloaded the recipe images.

The studies in the movie domain are based on data from the MovieLens platform. The dataset[4] contains 58,000 movies of which 45,161 movies have complete metadata information such as movie title, cover URLs, plot information, etc. The movie covers were downloaded from the TMDB website[5], which provides an open access API to their services[6]. More details about the content features of the two datasets can be found in the appendix in Table 16.

In the following sections, we will provide the details of the studies that we conducted to learn and validate distance functions for similar item recommendations. The study designs for both domains were very similar. We will therefore focus our discussions on only one of the domains, recipe recommendations, and report specific aspects and differences for the movie domain afterwards.

4 Learning the Similarity Function – Recipe Domain

As the brief literature review in Section 2.2 shows, there are various aspects that can determine the similarity between two recipes. Correspondingly, our approach is to learn a similarity function that considers multiple aspects in parallel. To learn such a combined function, we designed a set of 17 functions which use a single feature of one of four types of information (title, image, ingredients, cooking directions) that were previously used to compute recommendations in the literature

[3] Note that on allrecipes.com the provided descriptions, e.g., ingredient lists, are peer-reviewed and standardized by community editors. This is in particular the case for recipes that are published under the main dish category, which we consider in this study. Applying our methods to other recipe datasets would make it necessary to apply a pre-processing step to standardize the ingredients in the corpus, see, for example [58].

[4] Released August 2018: `https://grouplens.org/datasets/movielens/latest/`

[5] `https://www.themoviedb.org/`

[6] `https://developers.themoviedb.org/3`

[14]. Our learning approach is then to find an optimal combination of these individual functions, where optimal means that the similarity functions minimize the discrepancy between the user-provided similarity judgements and the predictions of the model.

In this section, we will first review the 17 single-aspect similarity measures considered in our experiments (Section 5.1). Then, we describe how we collected similarity judgements from users (Section 5.2) and analyze the obtained dataset (Section 5.3). Using the same 17 functions, we then evaluate the predictive performance of different machine learning techniques, using the human judgements as gold standard (Section 5.3.3).

4.1 Catalog of Similarity Measures

Our similarity measures relate to four types of item features, the recipe title, image, ingredient list, and directions. Generally, we can compute the similarity between two sets of feature values in various ways. In the research literature on recommender systems, the Jaccard coefficient or cosine similarity[7] are often used, depending on the encoding of the feature values (binary or numeric). For text documents, TF-IDF encodings are very common in the information retrieval literature; in recent years, also *embeddings* are often used as item representations. In our work, we use several ways of computing the similarity between two recipes for all types of features. The details for all measures are shown in Table 1.

4.1.1 Title-based Similarity

We used four string-based similarity measures and one based on topical similarity. The string-based measures rely on the Levenshtein (LV) distance metric [67], the Longest Common Subsequence distance (LCS) metric [2], Jaro-Winkler's (JW) method [27], and the Bi-Gram distance (BI) method [31]. To determine the *similarity* values (*sim*) from the *distance* (*dist*) for two recipes r_i and r_j, we use $sim(r_i, r_j) = 1 - |dist(r_i, r_j)|$.

The fifth measure is based on LDA topic modeling [5] for the given recipe titles[8]. We set the number of topics to 100 after experimentation[9]. To compare two recipes, we finally compute the cosine similarity between two resulting weight vectors $LDA(r_i)$, and $LDA(r_j)$. We therefore compute $sim(r_i, r_j) = cos(LDA(Title(r_i)), LDA(Title(r_j)))$.

4.1.2 Image-based Similarity

We considered six image-based similarity measures in our study. Five of them are low-level image metrics based on image brightness, sharpness, contrast, and

[7] Details about the exact computation of the measures are provided in Table 10 in the appendix.

[8] LDA was also successfully used for recipe titles in [32] and [45].

[9] Perplexity was used as criterion to tune the model parameters. We run experiments from 10 to 1000 topics for all LDA models. At the end we decided to use the models with 100 topics which gave us close to optimal performance while keeping the number of features and computational costs low.

Table 1: Similarity metrics computed based on recipe titles, images, ingredients and cooking directions.

Name	Metric	Explanation
Title:LV	$sim(r_i, r_j) = 1 - \lvert dist_{LEV}(r_i, r_j)\rvert$	Title Levenshtein Distance-based similarity
Title:JW	$sim(r_i, r_j) = 1 - \lvert dist_{JW}(r_i, r_j)\rvert$	Title Jaro-Winkler Distance-based similarity
Title:LCS	$sim(r_i, r_j) = 1 - \lvert dist_{LCS}(r_i, r_j)\rvert$	Title Longest Common Subsequence Distance-based similarity
Title:BI	$sim(r_i, r_j) = 1 - \lvert dist_{BI}(r_i, r_j)\rvert$	Title Bi-Gram Distance-based similarity
Title:LDA	$sim(r_i, r_j) = \frac{LDA(Title(r_i)) \cdot LDA(Title(r_j))}{\lVert LDA(Title(r_i))\rVert \lVert LDA(Title(r_j))\rVert}$	Title LDA Cosine-based similarity
Image:BR	$sim(r_i, r_j) = 1 - \lvert BR(r_i) - BR(r_j)\rvert$	Image Brightness Distance-based similarity
Image:SH	$sim(r_i, r_j) = 1 - \lvert SH(r_i) - SH(r_j)\rvert$	Image Sharpness Distance-based similarity
Image:CO	$sim(r_i, r_j) = 1 - \lvert CO(r_i) - CO(r_j)\rvert$	Image Contrast Distance-based similarity
Image:COL	$sim(r_i, r_j) = 1 - \lvert COL(r_i) - COL(r_j)\rvert$	Image Colorfulness Distance-based similarity
Image:EN	$sim(r_i, r_j) = 1 - \lvert EN(r_i) - EN(r_j)\rvert$	Image Entropy Distance-based similarity
Image:EMB	$sim(r_i, r_j) = \frac{EMB(r_i) \cdot EMB(r_j)}{\lVert EMB(r_i)\rVert \lVert EMB(r_j)\rVert}$	Image Embedding Cosine-based similarity
Ing:COS	$sim(r_i, r_j) = \frac{Ing(r_i) \cdot Ing(r_j)}{\lVert Ing(r_i)\rVert \lVert Ing(r_j)\rVert}$	Ingredients Cosine-based similarity
Ing:JACC	$sim(r_i, r_j) = \frac{\{Ing(r_i)\} \cap \{Ing(r_j)\}}{\{Ing(r_i)\} \cup \{Ing(r_j)\}}$	Ingredients Jaccard-based similarity
Ing:TFIDF	$sim(r_i, r_j) = \frac{TFIDF(Ing(r_i)) \cdot TFIDF(Ing(r_j))}{\lVert TFIDF(Ing(r_i))\rVert \lVert TFIDF(Ing(r_i))\rVert}$	Ingredients Text Cosine-based similarity
Ing:LDA	$sim(r_i, r_j) = \frac{LDA(Ing(r_i)) \cdot LDA(Ing(r_j))}{\lVert LDA(Ing(r_i))\rVert \lVert LDA(Ing(r_j))\rVert}$	Ingredients LDA Cosine-based similarity
Dir:TFIDF	$sim(r_i, r_j) = \frac{TFIDF(Dir(r_i)) \cdot TFIDF(Dir(r_j))}{\lVert TFIDF(Dir(r_i))\rVert \lVert TFIDF(Dir(r_j))\rVert}$	Cooking Directions Cosine-based similarity
Dir:LDA	$sim(r_i, r_j) = \frac{LDA(Dir(r_i)) \cdot LDA(Dir(r_j))}{\lVert LDA(Dir(r_i))\rVert \lVert LDA(Dir(r_j))\rVert}$	Cooking Directions LDA Cosine-based similarity

entropy [48]. The sixth, more complex one, is a based on convolutional neural networks (CNNs) and image embeddings [51]. Both feature spaces—low-level image features and CNN features—have been shown to be useful in recommendation scenarios, e.g., to recommend artwork [36] or pins in Pinterest [12]. To measure the similarity between two recipes based on these low-level image features (LO_{Image}), we use the Manhattan distance., i.e., $sim(r_i, r_j) = 1 - \lvert LO_{Image}(r_i) - LO_{Image}(r_j)\rvert$. The following low-level image features using the OpenIMAJ library[10] as proposed by [48] were computed:

[10] `http://www.openimaj.org/`

- *Brightness* (BR) considers the subjective visual perception of the energy output of a light source and can be calculated as follows using the NTSC standard:

$$BR = \frac{1}{N}\sum_{x,y} Y_{xy}, \text{ with} \tag{1}$$
$$Y_{xy} = (0.299 * R_{xy} + 0.587 * G_{xy} + 0.114 * B_{xy}),$$

  where Y_{xy} denotes the luminance value and N is the number of pixels in the image. R_{xy}, G_{xy}, and B_{xy} are the three RGB color space channels of pixel(x,y).
- *Sharpness* (SH) can be computed using the Laplacian L of the image, divided by the locale average luminance (μ_{xy}) around pixel (x,y):

$$SH = \sum_{x,y} \frac{L(x,y)}{\mu_{xy}}, \text{ with } L(x,y) = \frac{\partial^2 I_{xy}}{\partial x^2} + \frac{\partial^2 I_{xy}}{\partial y^2}, \tag{2}$$

  where I_{xy} is the intensity of a pixel.
- *Contrast* (CO) is the relative difference luminance in an image using the intensity of each pixel. In this work, we employ the often used root mean square contrast (RMS-contrast) [48]:

$$CO = \frac{1}{N}\sum_{x,y} (I_{xy} - \overline{I}), \tag{3}$$

  where I_{xy} is the intensity of a pixel, $\overline{I}$ represents the arithmetic mean of the pixel intensity and N is the number of pixels.
- *Colorfulness* (COL) can be calculated via the individual color distance of the pixels. Therefore, we first transfer the images to the sRGB color space defined as $rg_{xy} = R_{xy} - G_{xy}$ and $yb_{xy} = 1/2(R_{xy} + G_{xy}) - B_{xy}$, where R_{xy}, G_{xy}, and B_{xy} are the color channels of the pixels, and subsequently measure colorfulness, as follows:

$$COL = \sigma_{rgyb} + 0.3 \cdot \mu_{rgyb}, \text{ with} \tag{4}$$
$$\sigma_{rgyb} = \sqrt{\sigma_{rg}^2 + \sigma_{yb}^2}, \ \mu_{rgyb} = \sqrt{\mu_{rg}^2 + \mu_{yb}^2},$$

  where σ and μ stand for the standard deviation and the arithmetic mean, and 0.3 is a pre-defined parameter in OpenIMAJ[11].
- *Entropy* (EN) can be seen as the amount of information content provided by a source. We use Shannon entropy to compare two images as follows. First, we convert the image to gray scale, where each pixel has exactly one intensity value. Second, we count the occurrence of each distinct value. We compute entropy as follows [50]:

$$EN = -\sum_{x \in [0..255]} p_x \cdot log_2(p_x), \tag{5}$$

  where p_x is the probability of finding the gray-scale value x among all the pixels in the image.

[11] The parameter was estimated in a user study by Hasler et al. [21] in 2003 and is considerd to be optimal. In their work, Hasler et al. obtained a correlation of more than 95% with human judgement using this formula and parameter.

- The sixth method is based on image embeddings. To compute the image embeddings (EMB) we rely on a pre-trained (ImageNet) VGG-16 network[12], as also employed in current state-of-the-art content-based recommender systems work, such as [12, 36]. Similarly, we use the first fully connected layer of this network as an output. Hence, each recipe image r is represented as a vector $EMB(r)$ with 4096 elements. Again, we use the cosine similarity between the embeddings to compare two recipes. Technically, we used the Keras[13] framework for the computations. All images were automatically downsampled to fit the input layer. Since all images were of the same height and width, downsampling was unproblematic.

4.1.3 Ingredients-based Similarity

We relied on four different metrics to determine the similarity between two recipes on the ingredient level.

- The first two are designed using an ingredient-based representation that is common in the recipe recommendation domain [14]. Each recipe r is encoded as a vector of its ingredients $Ing(r)$, where the values of the vector represent the normalized weight (in grams/100g of a recipe) of the respective ingredient. We use the cosine similarity and the Jaccard coefficient as alternative similarity functions.
- The third metric is based on TF-IDF encoding the entire block of text that contains the ingredient description of a recipe. The distance between two recipes is determined with the cosine similarity.
- The fourth metric is topic-based. We again use Latent Dirichlet Allocation (LDA) to derive a topic distribution from the text describing the ingredients and again rely on the cosine similarity function.

4.1.4 Directions-based Similarity

We computed two measures based on the block of text that contains the cooking directions for each recipe. One is based on a TF-IDF encoding of the text, and one based on topic modeling (LDA), which was done in a similar way as described above. In both cases, we used the cosine similarity to compare two recipes.

Generally, in our work we mostly relied on TF-IDF and LDA-based representations for text descriptions, e.g., for directions for recipes or plots for movies, for two reasons. First, these representations have been successfully used to encode text documents in a number of application domains in the past. Second, the same encoding techniques were also used in previous works, see, e.g., [14, 45, 66]. Clearly, a number of alternative encodings can be used here, and we see the exploration of alternative approaches, e.g., based on recent embedding approaches as an interesting area for future work.

[12] We plan to explore the use of alternative architectures in the future, such as, ResNet [23], Inception [52], etc.
[13] `https://keras.io/`

4.2 Collecting Human Judgments

In this section, we describe our procedure for collecting human similarity judgements as used in *Study 1a* and *Study 1b* in more detail. To recruit participants for the study, we used the crowdsourcing platform Amazon Mechanical Turk. The main task of the "workers" was to assess the similarity pairs of recipes on a five-point scale.[14] Furthermore, they answered additional questions about their background, preferences, and similarity judgement approach.

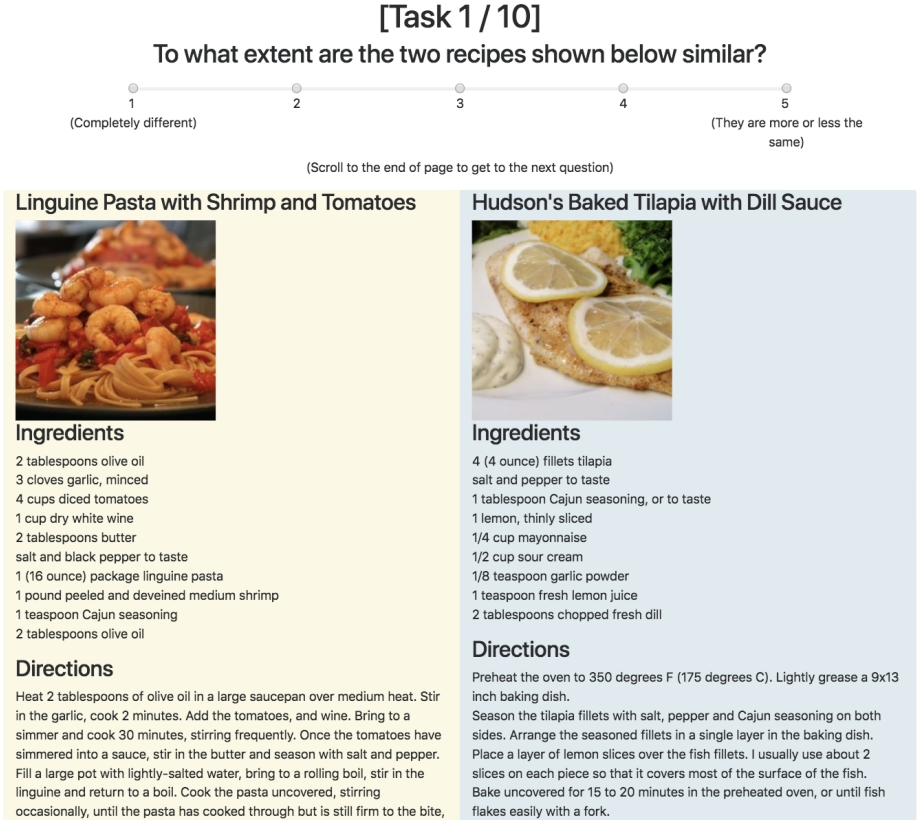

Fig. 1: Web interface to collect similarity judgements for recipes (*Study 1a*). The extreme values for the users' responses were "Completely Different (1)" and "They are more or less the same (5)".

[14] This procedure is similar to the one used in [66]. Alternative approaches for collecting similarity judgements are possible, e.g., by using a third item as a reference for the participants. Such designs might, however, lead to an increased complexity of the judgement task.

4.2.1 Determining Pairs for Human Judgment

To make sure that the recipe pairs to be judged are not entirely different (e.g., a cocktail recipe and a main dish), we restricted the selection of recipes in our experiments to the category of main dishes[15]. We, furthermore, only considered recipes with more than 20 ratings by the community, which ensures that the recipes are not too obscure and have a certain minimum quality. As shown in [57], the average number of ratings per recipe is 18. The threshold of 20 ensures that we do not base our work on niche recipes, which are not known by many people and where the similarity estimate might be not too reliable, e.g., due to the use of rarely used ingredients. This filtering process finally led to a set of 1,031 recipes.

In the next step, we determined a set of pairs to be presented to the human judges. To ensure that all aspects covered by our 17 similarity measures are considered, we proceeded as follows. First, we calculated all pairwise similarity values for all 17 measures. We then computed an overall similarity value for each pair by using a linear combination of these 17 measures using equal weights. Then, to ensure that there is enough variety in the pairs to be evaluated by humans, we employed a biased stratified sampling strategy [45, 66] to cover all parts of the similarity distribution. Using this strategy, we sampled $2,000$ recipes lying between quantile [Q0-Q1] of the distribution, $2,000$ lying between quantile [Q2-Q8], and $2,000$ recipes lying between quantile [Q9-Q10]. This process resulted in a sample of $6,000$ recipe pairs that can be used for human judgement.

4.2.2 Data Collection

We implemented a web application for the purpose of data collection. Each human judge was presented with 10 randomly chosen recipe pairs and asked to state to what extent the two recipes were similar. When selecting the pairs, we made sure that each pair of recipes is only rated by one human judge[16]. For the response, a five-point Likert scale was used, as shown in Figure 1. In addition, we asked for each pair—again using 5-point Likert items—to what extent they used the different factors (recipe title, image, ingredient list and directions) for their similarity assessment. After collecting the 10 judgements, the participants were asked questions about their gender, age range, how often they cook at home, and how often they use online recipe portals.

We took different actions to ensure that the responses by the crowdworkers were reliable [7, 22, 42]. First, we recruited only crowdworkers who had a "HIT accept rate"[17] of more than 98% on Mechanical Turk and who had a positive

[15] We have chosen main dishes as they are one of the most popular categories on the platform and we did not want that our study is confined to a smaller subset of recipe types on the platforms. Second, main dishes can be quite varied, which makes the similar item retrieval task more challenging than, for example, for deserts. Finally, one of our goals was to be consistent with previous works which also used main dishes as a basis for their experiments, e.g., [24, 59].

[16] The reason for using this procedure is to ensure that we obtain a larger number of judgements for a diverse set of items. This in turn allows to train more reliable models with a constrained budget. Having more judges per pair is possible, but needs significantly more study participants if we want to make sure that many dishes or movies are covered by the judgements.

[17] HIT stands for Human Intelligence Task on Amazon Mechanical Turk.

evaluation for more than 500 hits in the past. Since *allrecipes.com* is US-based, we limited participation to US residents. In addition, our web application included an attention check. In one of the recipe descriptions of the 10 presented pairs, we instructed the participants to answer all questions for this pair with the highest possible rating, independent of what they think. We estimated that our study participants will work approximately 5-10 minutes on the task on average. The reimbursement for the task was therefore set to USD 0.5 per HIT.

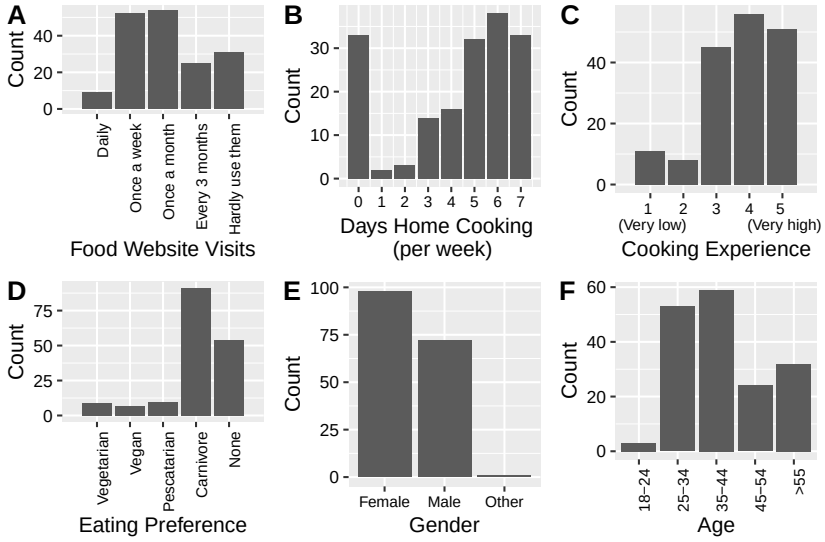

Fig. 2: Crowdworker characteristics of the similarity assessment study (*Study 1a*).

4.2.3 Participants

We recruited 400 crowdworkers for collecting the similarity judgements. 381 of them completed the study successfully and thus evaluated 3,810 recipe pairs. The median working time to complete the survey was 10 minutes, which was slightly higher than estimated.

Figure 2(A-F) shows the distribution of the characteristics of the participants. On average, the participants' self assessment regarding their cooking experience, their frequency of cooking at home, and their use of online recipe sites is relatively high, see Figure 2(A-C). Looking at the eating preferences, gender distribution, and age, see Figure 2(D-F), we can consider the sample to be diverse.

To some surprise, only 171 (44.88%) users passed the attention check, even though the specific instructions were printed at the beginning of the cooking directions in upper-case letters, and even though we restricted participation to experienced crowdworkers. Filtering out responses by users who were not working carefully enough, i.e, who did not pass the attention check, left us with 1,539 human similarity judgements.

4.3 Results

4.3.1 Information Cue Usage

We first looked at what the participants stated about their use of different types of information (information cues) when assessing the similarity of two recipes. Figure 3(A) shows that the title and the image were the most important factors according to the participants' self assessment. Ingredients were slightly less important, and the cooking directions were the least relevant factor. A statistical analysis using a one-way ANOVA[18] and a Tukey's HSD post-hoc test reveals that most differences are significant ($p < 0.01$ and $p < 0.001$), except for the difference between the image and the title cue, see Figure 3(B).

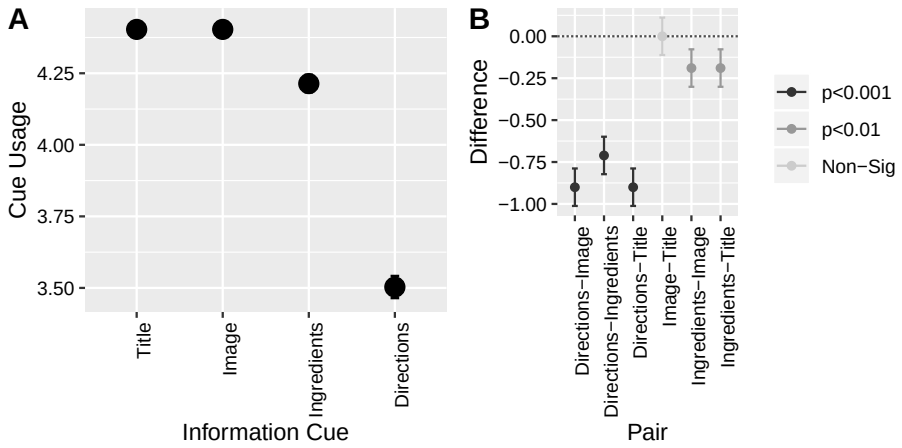

Fig. 3: *Study 1a*: (A) Information cue usage (means and std. errors) and (B) Pairwise comparison. Scale: 1 (not at all) — 5 (totally agree).

4.3.2 Correlation Analysis (RQ1)

Next, to address *RQ1* on the relative importance of the different features, we analyzed to what extent the provided similarity judgements correlated with our 17 computational similarity measures. Table 2 shows the results for Spearman's correlation coefficient[19] for those users who passed the attention check (ρ_{pass}) and all users (ρ_{all}).

Generally, the results show that the correlation with the pairwise judgements improves when the non-attentive study participants were removed. The highest correlation was observed with the *ING:TFIDF* metric ($\rho = 0.56$, $p < 0.001$), i.e., when treating the list of ingredients as a *block of text*. This is an interesting result,

[18] The homogeneity of variances for all ANOVA tests was checked with Levene's test.

[19] We have chosen Spearman as a correlation metric as the data (=user ratings) is (a) not normally distributed and (b) on an ordinal scale.

Table 2: *Study 1a*: Similarity metric correlation (Spearman) with user similarity estimates. ρ_{pass} indicate correlations with users who passed the attention check, while ρ_{all} denotes all users. Note: $^{*}p < 0.05, ^{**}p < 0.01, ^{***}p < 0.001$.

Metric	ρ_{pass}	ρ_{all}
Title:LV	0.48^{***}	0.38^{***}
Title:JW	0.46^{***}	0.35^{***}
Title:LCS	0.50^{***}	0.40^{***}
Title:BI	0.48^{***}	0.38^{***}
Title:LDA	0.22^{***}	0.19^{***}
Image:BR	0.18^{**}	0.14^{*}
Image:SH	0.16^{*}	0.11^{*}
Image:CO	0.29^{***}	0.20^{***}
Image:COL	0.09^{*}	0.07^{*}
Image:EN	0.34^{***}	0.28^{***}
Image:EMB	0.44^{***}	0.34^{***}
Ing:COS	0.54^{***}	0.44^{***}
Ing:JACC	0.51^{***}	0.41^{***}
Ing:TFIDF	$\mathbf{0.56}^{***}$	$\mathbf{0.44}^{***}$
Ing:LDA	0.45^{***}	0.36^{***}
Dir:TFIDF	0.50^{***}	0.40^{***}
Dir:LDA	0.54^{***}	0.43^{***}

because in the literature, the *ING:COS* similarity metric is typically preferred [14], which uses a more structured encoding of the ingredients. The metric with the lowest score is based on the image sharpness (*Image:SH*, $\rho = 0.16$, $p < 0.05$). However, image embeddings (*Image:EMB*, $\rho = 0.44$, $p < 0.001$), appear to be well correlated with human judgements in the recipe context[20].

In order to better understand the correlation of the *different types* of features, we analyzed the correlations when the similarity metrics of each type are considered together. Table 3 shows the correlations with human judgement (first row) and the other features types, when the metrics of each type are linearly combined with equal weights. We also show the correlation when all metrics are combined (*All*). The results show that ingredient-based and directions-based metrics, as well as the simultaneous consideration of all metrics, lead to the highest correlation values ($\rho = 0.61$, $p < 0.001$). Using only image-based metrics leads to a much lower, but still significant correlation ($\rho = 0.45$, $p < 0.001$). What can also be seen in the figure is that all features are correlated to a high degree, which indicates high multicollinearity. This, as a result, might lead to lower predictive performance in case regression models are employed which cannot deal well with such a situation.

Comparing these correlations with the participants' self assessments in Figure 3 shows an interesting contrast. While participants say that the recipe image is highly important for them, it turns out that this information cue—at least

[20] Image embeddings have been shown to be useful in many different application areas of multimedia. Recently, image embeddings have not only been used to classify images but also in the context of recommender systems to, for example, recommend images, etc. to people, see, e.g., [36]. Compared to explicit feature-based approaches, as also used in this paper, embeddings can capture several aspects of an image at the same time such as shapes, color, etc.

Table 3: *Study 1a*: Similarity metric correlation (Spearman) with user similarity estimates per cues when metrics are linearly combined with equal weights in the linear model. Note: All correlations are significant ($p < 0.001$).

	Humans	Title	Image	Ingredients	Directions	All
Humans	**1**	**0.53**	**0.42**	**0.61**	**0.60**	**0.61**
Title	–	1	0.50	0.57	0.59	0.72
Image	–	–	1	0.55	0.60	0.81
Ingredients	–	–	–	1	0.75	0.80
Directions	–	–	–	–	1	0.85
All	–	–	–	–	–	1

when using the specific image-based similarity measures from our experiment—is significantly less correlated with their similarity assessments than other features. The cooking directions, in contrast, which were considered much less important by the users, turned out to be well correlated with the provided human judgements. Overall, given that some of our image similarity measures actually correlate well with the participants' similarity judgements, see Table 2, we see an indication that directly asking users what they *think* is important is potentially not the most reliable basis when designing a similarity measure for a domain.[21]

4.3.3 Learning the Similarity Function (RQ2)

Since not all similarity metrics are equally correlated with human judgements, it is intuitive to apply machine learning to find the best metric combination. In other words, our goal—corresponding to *RQ2* on how to combine features for prediction—is to learn a model that leads to the lowest prediction error, i.e., the lowest average deviation from the predicted similarity for a given recipe pair and the human similarity judgements for the same pair.

Many machine learning algorithms can be in principle applied for the problem. In this work, we used different types of regression models such as Linear Regression (LR), Ridge Regression (Ridge), and Lasso Regression (Lasso), where the latter two are often considered to be better able to handle multicollinearities[22]. Furthermore, we included the Random Forest (RF) and Gradient Boosting (GB) techniques in our experiments, as they often lead to superior performance than regression models. As baselines, we used the overall mean of all similarity judgements as well as a random predictor. Formally, this can be expressed as follows:

$$sim_H(r_i, r_j) = REG(sim_{f_k}(r_i, r_j), ..., sim_{f_k}(r_i, r_j)), \tag{6}$$

where r_i and r_j are recipe pairs in the set of all recipes R, $sim_H(r_i, r_j)$ is the unique human similarity judgement for a recipe pair to be predicted on a scale $[1...5]$. Finally, REG is any arbitrary regression method employing feature-based

[21] Similar discrepancies were previously analyzed in the field of psychology, e.g., in [25].

[22] Compared to a standard Ordinary Least Squares models, Lasso and Ridge regressions introduce regularization terms (penalties) in their models [54]. The aim of Ridge regression is to "minimize the sum of squared residuals but also penalize the size of parameter estimates, in order to shrink them towards zero" [39]. The penalty is also called L2 penalty. Lasso, in contrast, is based on an L1 penalty; for further details see [39]. An alternative would be to use explicit feature selection such as done in [40].

similarity functions $sim_{f_k}(r_i, r_j)$ as, e.g., presented in Table 1. For example, in case of a linear model (LM), REG becomes:

$$REG = \sum_{f \in F} \beta_f * sim_f(r_i, r_j), \tag{7}$$

where $sim_f(r_i, r_j)$ are feature-based similarity functions, F is a set representing all available feature-based similarity functions, and β_f denotes the weights to be learned in the model.

To evaluate the models, we used the following performance measures: Root Mean Squared Error (RMSE), R squared (R^2), Mean Absolute Error (MAE), and Spearman Correlation (ρ). The performance measures were determined as the average obtained from a five-fold cross-validation procedure. Grid search was applied on a validation set from the training data to find the optimal hyper-parameters for each model[23]

The results of this predictive modeling experiment are shown in Table 4. For the given dataset, Ridge regression led to the best results, with the lowest RMSE and MAE values. The differences compared to the other models are sometimes very small. The improvement compared to the baselines is, however, substantial and statistically significant according to a Wilcoxon Rank-Sum test ($p < 0.01$).

Table 4: Performance of different learning approaches (recipe domain).

Method	RMSE	R^2	MAE	ρ
(Instances = 1,539)				
Model performance (All features)				
All (RF)	0.8958	0.4734	0.6787	0.6425
All (GB)	0.8805	0.4921	0.6672	0.6390
All (LM)	0.8700	0.5022	0.6668	0.6512
All (Lasso)	0.8667	0.5049	0.6680	0.6136
All (Ridge)	**0.8654**	**0.5063**	**0.6651**	**0.6625**
Baselines				
Mean	1.2292	0.4995	1.0433	0.0184
Random	1.2290	0.0010	1.0435	0.0489

Figure 4 shows the importance of the different predictor features, i.e., the normalized ranks of the model coefficients, for the best performing model as determined with the "varImp" method of R's *caret* package. The results are in line with the observations from above: Directions and ingredients are the most important metrics, whereas image-based features—while still relevant—are slightly less important.

Generally, in terms of the predictive performance, one can try to consider additional factors besides the similarity metrics. If, for example, the user characteristics are known, one can factor them into the prediction models. One assumption

[23] We used R's caret package for that purpose. Further details, on model training and parameter tuning can be found here: `https://topepo.github.io/caret/model-training-and-tuning.html#basic-parameter-tuning`

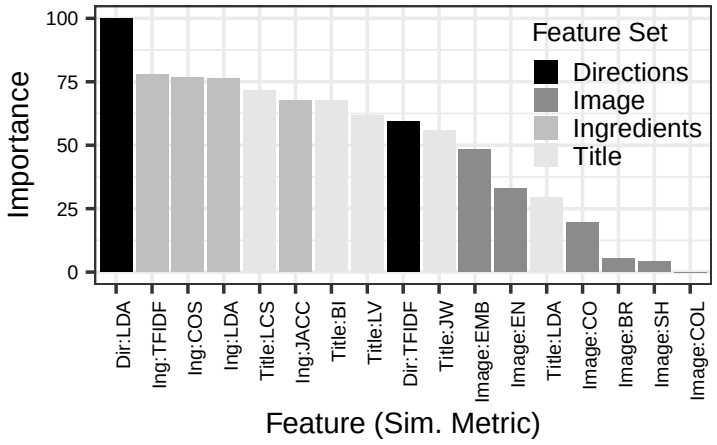

Fig. 4: Feature importance for the best performing Ridge regression model (recipe domain).

could be that participants with more cooking experience look more closely at the directions, while others might be more influenced by the recipe image.

We have therefore performed additional experiments where we constructed models that consider the characteristics of the user that were collected in the study (age, gender, cooking experience etc.) as predictor variables. For these experiments, we used Ridge regression as a learning technique. The results of these experiments are shown in Table 5.

The results show that considering certain additional features has only limited impact on the prediction performance when considered in isolation. However, when applied in combination, a significant improvement in terms of the performance measures (according to a Wilcoxon Rank-Sum test with $p < 0.05$) can be achieved. Note that we do not consider these additional user characteristics in our validation studies described later, since we generally cannot assume that the user characteristics are known.

Finally, Table 6 presents the results of four models when considering features from one information cue at the time. In general, the results reveal that the directions feature set performs best and the image cue worst. Interesting to note here is that while titles, ingredients, and directions perform similarly, as presented in Table 3, correlations can be significantly improved for the image feature set (LM = 0.42 vs. Ridge = 0.49, $p < 0.05$ according to a Pearson and Filon's z-test) if we consider Ridge regression as a learning method rather than a simple linear model with equal weights. Another interesting finding is that direction-based cues are among the best predictors, which is in line with [45]. In their work, Rokicki et al. show that cooking directions are better suited to predicting the healthiness of a meal than ingredients, as they, in general, contain the most information about a recipe.

Table 5: Results when considering additional features (recipe domain).

Method	RMSE	R^2	MAE	ρ
(Instances = 1,539)				
Model performance (All features)				
All (Ridge)	**0.8654**	**0.5063**	**0.6651**	**0.6625**
All (Ridge) + additional features				
Recipe Website Visits	0.8684	0.5031	0.6668	0.6558
Home Cooking	0.8648	0.5065	0.6646	0.6605
Cooking Experience	0.8631	0.5079	0.6615	0.6615
Age	0.8562	0.5170	0.6570	0.6699
Gender	0.8521	0.5203	0.6558	0.6755
All User Characteristics	**0.8393**	**0.5336**	**0.6448**	**0.6865**

Table 6: Results when considering only one information cue at the time (recipe domain).

Method	RMSE	R^2	MAE	ρ
(Instances = 1,539)				
Ridge Regression per Information Cue				
Title	1.0245	0.3079	0.8348	0.5278
Image	1.0680	0.2478	0.8706	0.4969
Ingredients	0.9449	0.4096	0.7493	0.6080
Directions	**0.9390**	**0.4190**	**0.7480**	**0.5998**

5 Learning the Similarity Function – Movie Domain

To test if our general approach generalizes to another domain, we repeated *Study 1a* in the movie domain (denoted as *Study 1b*) and we also trained various machine learning models to predict user-perceived similarity levels from the human judgement data.

5.1 Catalog of Similarity Measures

We created a catalog of 20 individual similarity measures. The measures fall into eight groups. The technical details regarding the exact calculation of all used measures are given in Table 10 in the appendix.

- Five measures were based on the *title* of the movies; again, we consider different forms of computing the similarities, e.g., based on the Levenshtein or the Jaro-Winkler distance.
- Six measures take the *cover image* as a basis to determine the similarity, based on their brightness, sharpness etc.
- *Plot summaries* were used in two different ways, using either a TF-IDF encoding or LDA.
- Two measures were based on *genre* similarity and one based on the *directors* of the movies.

- Inspired by the work in [66], we defined one similarity measure based on the *tag-genomes* of the movies, and one by their distance in the latent space that results from applying matrix factorization (*SVD*).
- The final two measures were based on the release date and the top-3 stars in the movies.

The main addition compared to *Study 1a* is that we include measures based on user-generated content (*tag-genome*) and that we consider collaborative information (*SVD*). While we cannot generally assume that such information is available, we included these measures in order to gauge the usefulness of including alternative types of information in the learning process.

Note that unlike in the recipe domain, where we focused on the category of main dishes, we did not concentrate on certain types of movies in this analysis, e.g., by only selecting movies of a certain genre. This decision was made to make our work comparable with previous work [66], which also did not focus on a certain genre or subset of the available movies. In our current approach, the genre is therefore used as a relevant item feature. However, future work could investigate if learning individual similarity functions for certain subsets of an item catalog is beneficial or not.

5.2 Collecting Human Judgments

Similar to *Study 1a*, we used a web interface to collect human judgements with the help of crowdworkers, see Figure 13 in the appendix. Furthermore, we restricted the selection of movies to those having obtained at least 2,000 ratings. The threshold of using movies recipes with more than 2,000 ratings was chosen to be consistent with previous works [66], where the goal was to avoid to base the research on too obscure (niche) items. Again, we used a stratified sampling strategy to determine a set of pairs to be presented to the participants. At the end of the process, we had 6,000 pairs involving 2,512 movies.

5.3 Results

We recruited another 400 crowdworkers for collecting the similarity judgements. Every participant was asked to rate 10 pairs, leading to 4,000 movie pairs. Figure 11(A-F) in the appendix shows the distribution of the characteristics of the participants. Filtering out responses by a large fraction of crowdworkers who were not working carefully enough, i.e, who did not pass the attention check[24], left us with 1,395 human similarity judgements. The following analyses are based on these 1,395 judgements.

5.3.1 Information Cue Usage

Figure 5 shows which information cues were used by the participants (based on their self-report). According to this analysis, plot and genre descriptions were the

[24] The attention check for the movie domain study was more or less identical as in the recipes study. Instead of displaying the attention check in the 'directions' text, we displayed it in the 'stars' section.

most important pieces of information by the participants, followed by the movie title and the cover image. Interestingly, the director, the release date, and the average star rating were on average not considered relevant at all.

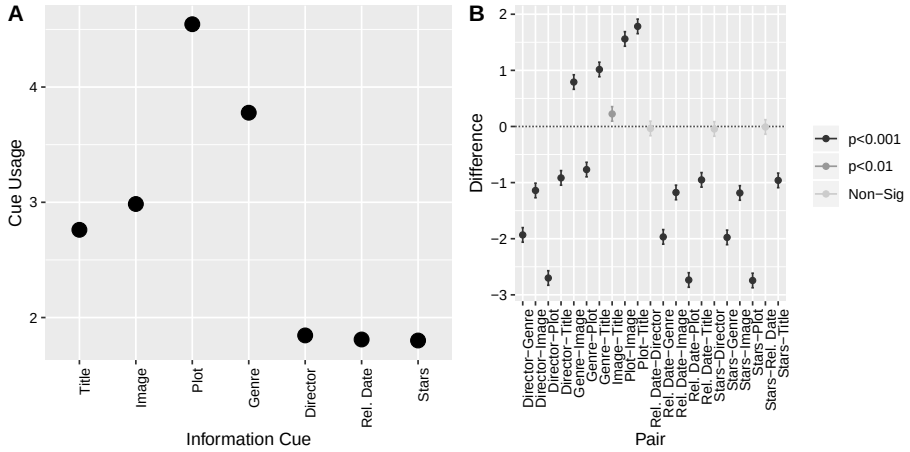

Fig. 5: *Study 1b* (A) Information cue usage (means and std. errors) and (B) Pairwise comparison. Scale: 1 (not at all) — 5 (totally agree).

5.3.2 Correlation Analysis (RQ1)

Table 7 shows the results for Spearman's correlation coefficient for those users who passed the attention check (ρ_{pass}) and all users (ρ_{all}). The strongest correlation between the user's judgement and the objective features was observed for the genre, but only when it was computed based on the Jaccard index (*Genre:JACC*, ρ = 0.56, p < 0.001). Plot features (*Plot:LDA*, ρ = 0.37, p < 0.001) also represent a comparably good predictor for human judgement, which is in line with the information cue usage statistic in Figure 5. The release date, somehow contradicting the participants self-report, to some surprise is also a good predictor (*Date:MD*, ρ = 0.37, p < 0.001). This confirms the findings from *Study 1a* that user-reported importance considerations are not necessarily good predictors. Of the images features, the image embeddings and image brightness show fairly strong correlations. Finally, the two special measures that were introduced in *Study 1b* (SVD-based similarity and similarities based on the tag-genome), were also very highly correlated with the assessments of the study participants (*Tag*, ρ = 0.49, p < 0.001 and *SVD*, ρ = 0.37, p < 0.001).

The correlation values when linearly combining the similarity measures of the same type are shown in Table 11 in the appendix.

5.3.3 Learning the Similarity Function (RQ2)

Like for *Study 1a*, we trained different types of machine learning models that combine the individual features to predict the user-perceived similarity of movies.

Table 7: *Study 1b (movie domain)* Similarity metric correlation (Spearman) with user similarity estimates. ρ_{pass} indicate correlations with users who passed the attention check, while ρ_{all} denotes all users. Note: $^{*}p < 0.05, ^{**}p < 0.01, ^{***}p < 0.001$.

Metric	ρ_{pass}	ρ_{all}
Title:LV	0.19^{***}	0.18^{***}
Title:JW	0.16^{***}	0.16^{***}
Title:LCS	0.20^{***}	0.19^{***}
Title:BI	0.17^{***}	0.17^{***}
Title:LDA	0.01	0.01
Image:EMB	0.18^{***}	0.16^{***}
Image:BR	0.22^{***}	0.20^{***}
Image:SH	0.10^{***}	0.08^{***}
Image:CO	0.03	0.03
Image:COL	0.15^{***}	0.14^{***}
Image:EN	0.15^{***}	0.09^{***}
Plot:TFIDF	0.25^{***}	0.20^{***}
Plot:LDA	0.37^{***}	0.34^{***}
Genre:JACC	$\mathbf{0.56}^{***}$	$\mathbf{0.53}^{***}$
Genre:LDA	0.13^{***}	0.13^{***}
Dir:Jacc	0.10^{***}	0.07^{***}
Date:MD	0.37^{***}	0.35^{***}
Stars:JACC	0.18^{***}	0.16^{***}
Tag	0.49^{***}	0.46^{***}
SVD	0.37^{***}	0.36^{***}

The results are shown in Table 8. Combining different features again proved to be favorable, and Ridge regression again led to the best results, thus confirming the findings of *Study 1a*. Like in the recipe domain, the differences between the tested regression models are often small. Compared to the baseline models, however, the improvements are again substantial and statistically significant according to a Wilcoxon Rank-Sum test ($p < 0.01$).

The relative importance of the features when used in the Ridge regression model are shown in Figure 12 in the appendix. Further information about the performance of individual features and the again helpful consideration of additional information about the users can be found in Table 13 and Table 12, also in the appendix.

5.3.4 Comparing Study 1a (recipes) and Study 1b (movies)

Overall, the findings of *Study 1b* in the movie domain are almost fully in line with those of *Study 1a* for the recipes. The correlation between some of our objective measures with user similarity estimates is in many cases good, which confirms the general suitability of the chosen measures. In both domains it turned out that combining all predictor variables leads to the best models. Ridge regression was in both cases the model that led to the highest accuracy. Finally, in both domains we

Table 8: *Study 1b*: Performance of different learning approaches.

Method	RMSE	R^2	MAE	ρ
(Instances = 1,395)				
Model performance (All features)				
All(Lasso)	0.8873	0.3574	0.7286	0.5952
All(RF)	0.8807	0.3543	0.7007	0.5943
All(GB)	0.8844	0.3489	0.7029	0.5897
All(LM)	0.8752	0.3616	0.6929	0.6007
All(Ridge)	**0.8745**	**0.3628**	**0.6926**	**0.6019**
Baselines				
Mean	1.0942	0.5001	0.9140	0.0001
Random	1.0948	0.0061	0.9140	0.0381

found some discrepancy between what users report is decisive for their estimates and what can be observed from the objective measures. In *Study 1b*, we included two additional features based on collaborative information (ratings and tags), and it turned out that these are correlated with the user's similarity perception as well.

6 Validating the Similarity Functions

The results so far show that using a combination of similarity measures leads to the best approximation of the human judgements. Our goal is now to validate that recommendations that are based on such a combined similarity function are also *perceived* to be more similar than recommendations that are based on individual information cues. We, furthermore, aim to assess the usefulness of similar item recommendations when based on different feature sets.

6.1 Design of the Validation Studies

We designed additional studies (*Study 2a* for recipes and *Study 2b* for movies) for that purpose, where the participants were presented with similar item recommendations that were generated using different recommendation strategies.

6.1.1 Implemented Recommendation Strategies

The general recommendation task can be described as follows. Given a recipe (or movie) r_i, find all top-k (in our case k=5)[25] most similar items r_j. Formally, this can be expressed as follows:

$$pred_k(r_i) = \underset{r_j \in R \setminus r_i}{\operatorname{argmax}^k} \{sim(r_i, r_j)\}, \tag{8}$$

[25] We chose a list length of 5 items not only to keep the cognitive load for participants low but also because on recipe sites often not more than 5 recommendations are displayed (without scrolling).

where $R \backslash r_i$ is the set of all items without r_i and $sim(r_i, r_j)$ is a similarity function.[26]

In the recipe domain (*Study 2a*), we compared six recommendation strategies. Five of them were regression-based and one was a random recommendation baseline. In the regression-based models, we varied the implementation of the *sim* function in Equation 8. One implementation used the combined and "optimal" function as discussed in the previous section, see Table 4: *All (Ridge)*. The other four were also learned using Ridge regression, but were limited to metrics of one type of features, i.e., recipe title, image, ingredients, or cooking directions, see Table 6. Using the same approach for the movie domain (*Study 2a*), we ended up with 12 strategies, which consisted of the optimal models (*All* and All_c), the individual models per feature type, plus the Tag-Genome approach, the SVD approach and a random baseline. While the *All* model combines all other models including SVD and Tag-Genome, but except random, ALL_c combines all content-based approaches, such as title, image, plot, genre, directors, release date and stars. For more details, see Table 9.

Table 9: Similar item recommendation strategies.

Name	Similarity metric(s) used
	Recipe Domain (see also Table 1)
All	All recipe similarity metrics combined using Ridge regression
Title	All recipe title similarity metrics combined using Ridge regression
Image	All recipe image similarity metrics combined using Ridge regression
Ingredients	All recipe ingredients similarity metrics combined using Ridge regression
Directions	All recipe directions similarity metrics combined using Ridge regression
Random	Random recipe recommendations
	Movie Domain (see also Table 10)
All	All movie similarity metrics combined using Ridge regression incl. SVD and tag genome
All_c	All movie similarity metrics combined using Ridge regression excl. SVD and tag genome
Title	All movie title similarity metrics combined using Ridge regression
Image	All movie image similarity metrics combined using Ridge regression
Plot	All movie plot similarity metrics combined using Ridge regression
Genre	All genre similarity metrics combined using Ridge regression
Directors	All directors similarity metrics combined using Ridge regression
Date	All date similarity metrics combined using Ridge regression
Stars	All stars similarity metrics combined using Ridge regression
SVD	Using the SVD similarity metric
Tag	Using the tag genome similarity movie metric
Random	Random movie recommendations

6.1.2 Data Collection

As with the first studies, we developed an online applications for *Study 2a* and *Study 2b*. The main task of the participants was to individually assess five similar item recommendations for a given reference item, see Figure 6 for an example in the recipe domain, and Figure 14 in the appendix for an example in the movie domain. The participants had to answer two questions in the form of five-point Likert scales for each recommendation. First, they had to state how similar they

[26] The set $R \backslash r_i$ does not contain recipes or movie pairs already used in *Study 1a* and *Study 1b*, respectively.

Fig. 6: Screen capture of *Study 2a* (recipe domain).

considered the recommended item with respect to the reference item. Second, they were asked to indicate how likely it is that they would try out each recommendation. Furthermore, we asked additional questions about the likeliness of trying out the reference recipe or watching the reference movie, and about their perception of the recommendations as a whole (see also [30, 44]), using five-point Likert scales. Specifically, we asked questions about the *helpfulness*, *diversity*, *surprisingness*, and *excitingness* of each recommendation list.

This procedure was repeated for five reference items and the corresponding recommendations. The reference items were selected from the pools of items that were used in *Study 1a* and *Study 1b*, respectively. The study participants were randomly assigned to one of the six (*Study 2a*) or twelve (*Study 2b*) conditions (recommendation strategies) in a between-subjects design. The reference items to be presented to the participants were also randomly chosen. After the participants

had completed the task for five reference items, we asked the participants if they plan to use the similar item recommendations in the future (*intention to reuse*). Finally, we again asked for additional user characteristics like age, gender, and so forth. The exact survey questions for all studies can be found in the appendix in Table 14 and 15.

6.1.3 Participants

We recruited 900 crowdworkers for *Study 2a* and another 1200 for *Study 2b*. The selection criteria (e.g., in terms of past success of the workers) were identical to *Study 1a* and *Study 1b*. Assuming a required workload of about 10 minutes, we paid USD 0.5 per HIT. We again implemented an attention check in the studies[27]. At the end, we had 349 participants who had successfully completed the task for the recipe domain and 837 successful participants for the movie domain.

The 349 users of *Study 1a* evaluated 1,745 recommendations lists in 6 conditions and provided 8,725 recipe similarity estimates. The median working time to complete the survey was 10 minutes, as estimated. The 837 participants of *Study 2b* provided 20,925 movie similarity assessments in 12 conditions. In the following, we report the results that were obtained based only on those users who completed the study and passed the attention check.

6.2 Results

In the subsequent discussions, we first analyze to what extent the different recommendation strategies lead to similar item suggestions that are actually also perceived to be similar be the participants. This is the main focus of *RQ3* described above. Subsequently, we investigate a number of facets related to the perceived usefulness the results from the different strategies.

6.2.1 Perceived Similarity (RQ3)

Figure 7(A) shows the average similarity judgements for the recommended items for each treatment condition in *Study 2a* in the recipe domain. The results confirm the trends that were observed in the analyses of the outcomes of *Study 1a*. Recommendations based on the directions, title, and ingredient lists, as well as the combined ("All") model, led to the highest similarity perception. The image-based approach, like in the offline experiments, performed significantly worse than the others ($p < 0.01$), according to a pairwise comparison using one-way ANOVA and Tukey's HSD post-hoc test.

The participants found the random recommendations, as expected, to be very dissimilar from the reference recipe. Employing a one-way ANOVA and Tukey's HSD post-hoc test shows that the random approach is significantly different from all other approaches. An interesting observation here is that the title-based approach performs very well, which was not expected from the offline experiments (see Table 6). One possible explanation for this slight discrepancy—remember that

[27] The attention check was in the 'description' section for the recipe recommender study and in the 'star(s)' section for the movie study.

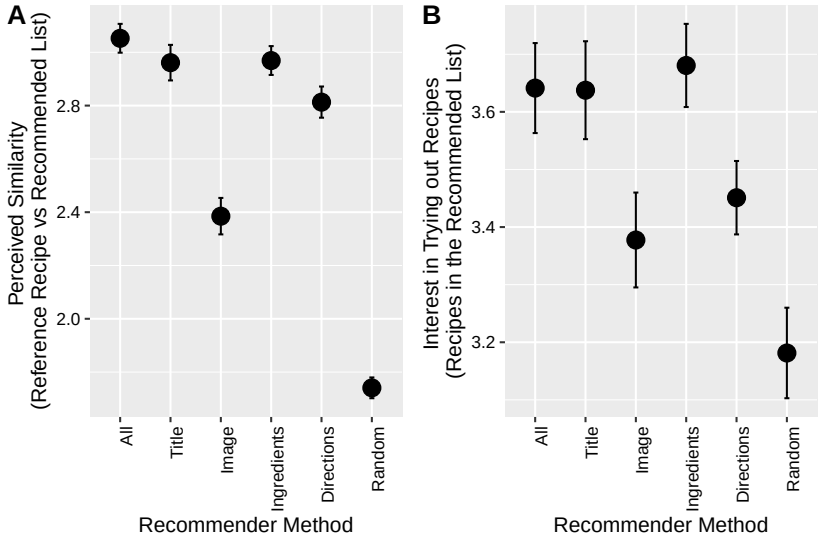

Fig. 7: *Study 2a*: (A) Perceived similarity (reference recipe vs recommended list) and (B) Interest in trying out a recommendation (means and std. errors). Scale: 1 (not at all) — 5 (very similar/will try).

the title-based metrics also worked quite well in the offline measurement—can lie in the different form of presentation in the validation study. In the validation study, multiple elements with similar titles were presented to the users, whereas the learned model was based on pairwise similarity judgements. More research is, however, required to better understand this phenomenon.

Figure 8(A) shows the corresponding results for the movie domain. As shown, the two combined models (*All* and All_c as well as the Tag-Genome and the SVD model) produced recommendations that led to the highest perceived similarity with the reference movie. According to a pairwise comparison using one-way ANOVA and Tukey's HSD post-hoc tests these four methods, however, were not found to be different to a statistically significant extent. Note that it is not very surprising that the tag model performs so well, because the tags that were provided by humans at the MovieLens site are actually based on the users' perception of key elements of a movie. As expected, the random model performed worst. All models, except the release date based model were significantly different from the random baseline ($p < 0.01$).

Overall, in both domains the combined method (All) was very effective in recommending items that were also perceived to be similar to the reference item. In the movie domain, it turned out that the tag-based and the SVD-based strategies were also quite effective. The applicability of these strategies however depends on the availability of community-provided data. Generally, the findings also indicate that the results obtained in the offline analysis are predictive for the online success. It, however, seems important to use human judgements as a gold standard instead of self assessments by users, expert knowledge, or intuition.

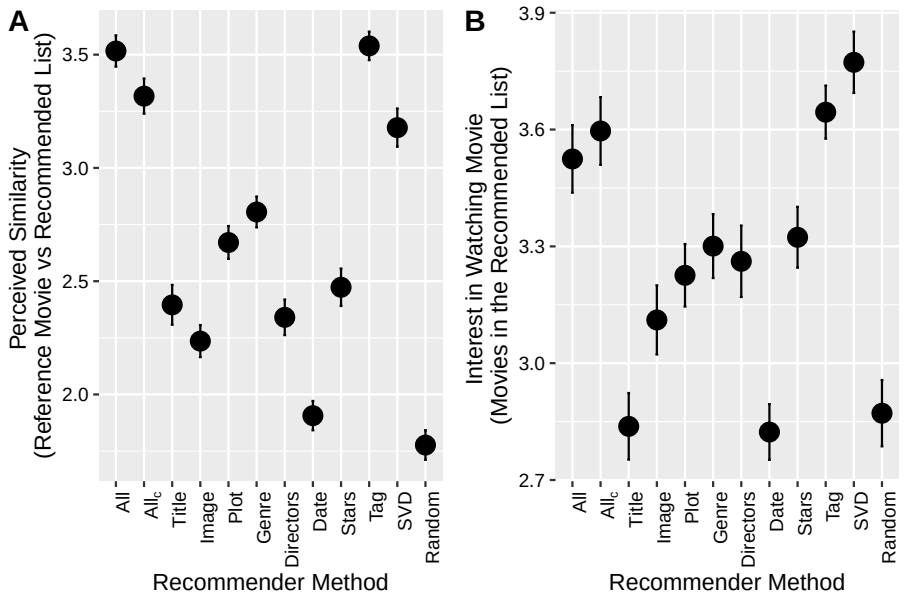

Fig. 8: *Study 2b*: (A) Perceived similarity (reference movie vs recommended list) and (B) Interest in trying out a recommendation (means and std. errors). Scale: 1 (not at all) — 5 (very similar/will try).

6.2.2 Usefulness (RQ4)

Recommending items that are very similar to each other can be of high value for users or not. Depending on the user's intent, they might be either interested in very similar recipes (e.g., because they already have the ingredients) or in recipes that are similar to what they know but also help them discover new things. Similar considerations apply for the movie domain. While our main focus is on perceived similarity, we also explored other potential forms of utility. The subsequent analyses therefore address *RQ4* on the *usefulness* of similar item recommendations.

Propensity to Try Out Recommendations. Figure 7(B) shows the average expressed intent of the participants to try out individual recommended recipes. In this measurement, we only considered those responses where the participants answered with a value higher than 3 that they were likely to try the reference recipe.[28] The results show that the combined model as well as the recommendations based on the title and the ingredients were those that the participants were most likely to try out. The method based on ingredients was slightly better than the runner-ups, although not to a statistically significant extent. The image-based recommendations appeared less attractive for users, and the random recommendations were deemed even less relevant. A one-way ANOVA and Tukey's HSD post-hoc test showed no

[28] Considering recommendations for reference recipes that the user does not like, e.g., because she is a vegetarian but the reference meal is meat, will lead to low response values also for the recommendations, as they are assumed to be similar.

significant differences between the combined model (All), the title-based model and the ingredients model. The differences to the random baseline were however significant ($p < 0.01$).

Figure 8 presents the results for the movie study showing that the both All models as well as the Tag Genome model and SVD performed best (no statistically significant difference), while title, release date and the random baseline performed worst, in terms of interest in watching a movie in the recommended list. These results are interesting as they show the power of collaborative filtering (SVD) and user-generated content (Tags). On the other hand, the combined models and All_c in particular provide evidence for the usefulness of content features.

Generally, regarding our assessment of the utility of the recommendations, note that applying typical measures like precision is in principle possible. However, it would require the introduction of an artificial threshold value to discern relevant from non-relevant items, with the additional problem that the ground truth for the items that were not presented to the users is not known.

Additional Quality Factors. In addition, we asked the participants of both validation studies to what extent they found the recommended list helpful, diverse, surprising, and exciting. Figure 9(A) shows the average *helpfulness* for each method for the recipe domain. The general trend is similar to the previous analysis of the users' propensity to try out the recommended recipes. Recommendations based on directions were considered to be the most helpful ones, the differences to the combined model, which is also considered very helpful, are, however, not statistically significant.

With respect to *diversity*, as shown in Figure 9(B), we can see that the diversity perception—as expected—is inversely related to the similarity perception.[29] Generally, whether or not high diversity and serendipity is desired, depends on the intended purpose and utility of the recommendation system. Optimizing solely for similarity will, as expected, not lead to high levels of diversity. Therefore, in case diversity is a desirable feature of the recommendations, various diversification strategies from the literature can be applied [1, 69].

The recommendations that were most similar to the reference recipe were also the least *surprising* ones, see Figure 9(C). A one-way ANOVA confirms that there are significant differences between the groups ($p < 0.001$) and Tukey's HSD post-hoc test confirms that these differences are significant ($p < 0.05$ and $p < 0.001$) for those pairs with no overlapping error bars. Finally, looking at Figure 9(D), we see that the random method was perceived as most *exciting* and the combined method (All) and the image-based method the least. However, a one-way ANOVA shows that there are no significant differences between the methods.

At the end of the study, we asked the users to what extent they will use similar item recommendations in the future (see Figure 10 for the results in the recipe domain). Again, on average the combined model leads to very good results. This time, the title-based recommendations performed worse and the image-based recommendations led to a significantly lower intention to rely on similar item recommendations in the future. Although these are interesting trends, a one-way ANOVA reveals no significant differences.

[29] We see this as another indicator of the reliability of the respondents.

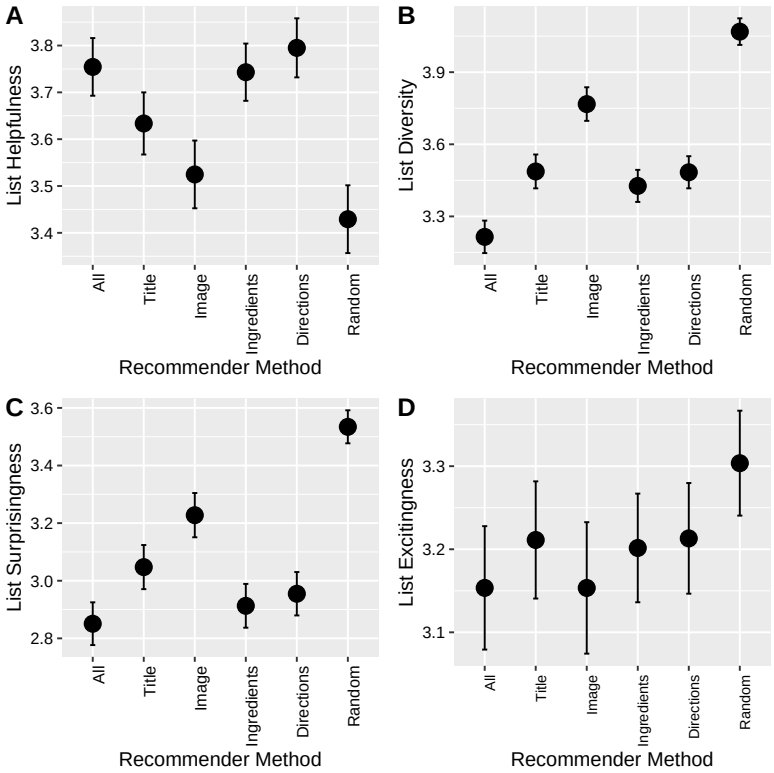

Fig. 9: *Study 2a* (A) Helpfulness, (B) Diversity, (C) Surprisingness and (D) Excitingness of the recommended lists (means and std. errors). Scale: 1 (not at all) — 5 (totally agree).

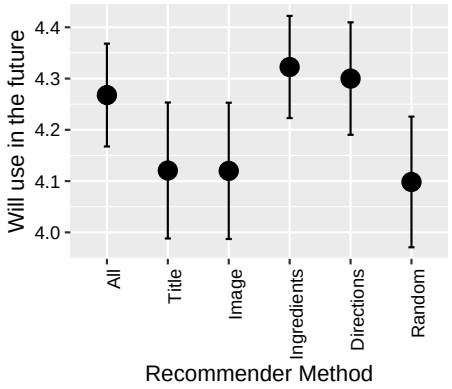

Fig. 10: *Study 2b* Intention to use the recommendation method in the future (means and std. errors). Scale: 1 (not at all) — 5 (totally agree).

Overall, the results obtained for the recipe domain show that the combined model not only resulted in recommendations that were perceived to be similar, but the users also expressed interest in trying them out and found them helpful in different ways. We see this as strong indicators of the usefulness of the learned similarity function in this application domain.

The results for the movie domain are mainly in line with these observations. The details can be found in Figure 15 and Figure 16 in the appendix. The tag-based, the SVD-based, and to some surprise also the genre-based method, were considered particularly helpful. The combined model also fared well in this dimension. The combined model, as expected, was leading to recommendations of low diversity and surprisingness, which was also the case for the tag-based approach. A one-way ANOVA confirms that there are significant differences between the groups ($p < 0.001$) and Tukey's HSD post-hoc test confirms that these differences are significant ($p < 0.05$ and $p < 0.001$) for those pairs with no overlapping error bars.

In terms of the participants' intention-to-reuse, the combined model was not better than most of the individual strategies, and not as good as the SVD-based and the tag-based model. However, a one-way ANOVA shows that there are no significant differences. Overall, our analysis shows that the tag-based model stands out in this comparison, even though—like the combined model—it produced lists of low diversity and surprisingness. Apparently, however, the community-provided tags are able to capture aspects of movies that are difficult to extract from meta-data, but which are particularly helpful for the users.

6.3 Comparison with the results from Yao and Harper [66]

Our study, as mentioned above in Section 2, shares some similarities with the work presented in [66], which also focuses on the user perception of similar item recommendations made in the context of a reference item. Differently from our work, one main goal in [66] was to understand to what extent different specifically-designed recommendation algorithms are able to generate movie recommendations that are perceived to be similar and useful. In our approach, in contrast, the goal was to learn the parameters of a regression-based recommendation algorithm and to validate it in a recommendation setting.

Furthermore, while our study was mostly based on item meta-data, Yao and Harper concentrate on community-provided information such as reviews, ratings, click events, or tags in their recommendation methods. The advantage of their approach is that user-provided and community-based information can be very helpful. Study 1b shows that the user-provided tags—the only community-provided information used also in our experiment—are indeed very well correlated with the collected similarity judgements. Our validation study (Study 2b) furthermore shows that tags are also leading to recommendations that are considered at leat as useful as a method that solely relies on meta-data features.

Interestingly, the ratings-based SVD method led to the highest usefulness scores in terms of the participants' tendency to try out a movie recommendation, whereas in [66] this method only led to mediocre results. These differences can probably be attributed to the different evaluation settings. In our work, we for example asked users to judge the usefulness of an item in the context of other

recommendations, whereas the assessment of the recommendation quality in [66] was made when the pairwise comparisons were presented.

Overall, we see our work and the work by Yao and Harper to be complementary. One specific advantage of our method is that it can be applied also for cold-start items for which know community feedback is yet available. However, given the strong potential of community-based and user-generated information, the combination of different types of data represents an interesting area of future research.

7 Limitations

So far, we have validated our approach in two application domains. Investigations for additional domains are part of our planned future works. Nonetheless, we are confident that the findings of our studies generalize at least to certain types of domains. Specifically, the problem of recipe and movie recommendation shares similarities with other domains of *quality and taste*, where the similarity perception can be both subjective and depending on the relative importance of different aspects to users.

A further potential limitation of our work is that our present studies are based on a specific set of computational similarity measures. These measures were successfully used before in the recommendation domain [36] and many of them correlate well with the users' judgements. However, there might be other metrics that are even better able to match the users' similarity perceptions. A particularly promising area in this context are image-based similarity measures that go beyond the comparably simple approaches that were used in our study. Interesting other alternatives are the SIFT, SURF or ORB methods [47] as well as the use of embedding-based approaches and other CNN architectures such as ResNet [23], Inception [52], etc. A further interesting direction of future work in that context lies in the exploration of structural similarity metrics as discussed in the psychology literature [60].

Additional investigations are also needed on the interplay between algorithmic performance and the presentation of these algorithms in a recommender interface. While in our two validation studies a common representation of similar item recommendations were used, we did not investigate different ways of presenting the similar item list. Finally, a closer look at memory biases is needed as the current work does not take into consideration to what extent a recipe or movie is known to the user. This might further influence how similarity estimates are made.

Finally, a general threat regarding generalization of our work lies in the fact that we relied on crowdworkers for the studies. Since we could identify and filter out a large fraction of non-attentive study participants through attention checks, we believe that this threat to validity is limited.

8 Conclusions & Outlook

The key findings of our research can be summarized as follows. Our work demonstrates the feasibility of learning similarity functions, as used, e.g., in similar item recommendations, from human judgements. It also turned out that considering

these human judgements is a necessity, because experts can err and because self assessments by users regarding the relative importance of certain factors might be misleading. Our experiments and studies also showed that it is important to consider several aspects in parallel when designing a similarity function, because similarity is a multi-faceted concept in many application domains. The validation through a user study furthermore showed that the chosen offline experimental design can be used as a predictor for the perception by users. Finally, we see our studies as a blueprint for further research in other domains. Specifically, going beyond previous works, it is not only important to rely on human assessments when designing and learning the similarity function, it is important to validate the function in the particular target application, in our case similar item recommendations.

Our immediate future work consists of the application of our learning and validation methodology to different domains. Furthermore, we will investigate trade-off problems in similar item recommendations. Such problems arise when the recommendations of too similar items and limited list diversity is not desirable in a particular application domain. For example, being recommended too many movies by the same director, compare also [69], might be undesirable for the user. In that context, we also plan to explore the use of multiple lists—as done on media streaming sites—where each list uses a different similarity criterion. Generally, some groups of users might rely more on certain information cues than others. For example, there might be differences between women and men as well as cultural differences. We will further explore such aspects as part of our future works.

References

1. Adomavicius, G., Kwon, Y.: Improving aggregate recommendation diversity using ranking-based techniques. IEEE Transactions on Knowledge and Data Engineering **24**(5) (2012)
2. Allison, L., Dix, T.I.: A bit-string longest-common-subsequence algorithm. Information Processing Letters **23**(5), 305–310 (1986)
3. Aucouturier, J.J., Pachet, F., et al.: Music similarity measures: What's the use? In: Proc. of ISMIR '02 (2002)
4. Beel, J., Langer, S.: A comparison of offline evaluations, online evaluations, and user studies in the context of research-paper recommender systems. In: Proc. of TPDL '15 (2015)
5. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. Journal of Machine Learning Research **3** (2003)
6. Brovman, Y.M., Jacob, M., Srinivasan, N., Neola, S., Galron, D., Snyder, R., Wang, P.: Optimizing similar item recommendations in a semi-structured marketplace to maximize conversion. In: Proc. of RecSys '16 (2016)
7. Buhrmester, M., Kwang, T., Gosling, S.D.: Amazon 's mechanical turk: A new source of inexpensive, yet high-quality, data? Perspectives on Psychological Science **6**(1) (2011)
8. Colucci, L., Doshi, P., Lee, K.L., Liang, J., Lin, Y., Vashishtha, I., Zhang, J., Jude, A.: Evaluating item-item similarity algorithms for movies. In: Proc. of CHI EA '16 (2016)

9. Cremonesi, P., Garzotto, F., Turrin, R.: Investigating the persuasion potential of recommender systems from a quality perspective: An empirical study. ACM Transactions on Intelligent Systems and Technology **2**(2) (2012)
10. Deldjoo, Y., Elahi, M., Cremonesi, P., Garzotto, F., Piazzolla, P., Quadrana, M.: Content-based video recommendation system based on stylistic visual features. Journal on Data Semantics **5**(2) (2016)
11. Ebizma: Ebizma rankings for recipe websites. available at `http://www.ebizmba.com/articles/recipe-websites`. last accessed on 19.04.2017 (2017)
12. Eksombatchai, C., Jindal, P., Liu, J.Z., Liu, Y., Sharma, R., Sugnet, C., Ulrich, M., Leskovec, J.: Pixie: A system for recommending 3+ billion items to 200+ million users in real-time. In: Proc. of The Web Conference '18 (2018)
13. Ellis, D.P.W., Whitman, B., Berenzweig, A., Lawrence, S.: The quest for ground truth in musical artist similarity. In: Proc. of ISMIR '02 (2002)
14. Elsweiler, D., Trattner, C., Harvey, M.: Exploiting food choice biases for healthier recipe recommendation. In: Proc. of SIGIR '17 (2017)
15. Freyne, J., Berkovsky, S.: Intelligent food planning: Personalized recipe recommendation. In: Proc. of IUI '10 (2010)
16. Garcin, F., Faltings, B., Donatsch, O., Alazzawi, A., Bruttin, C., Huber, A.: Offline and online evaluation of news recommender systems at swissinfo.ch. In: Proc. of RecSys '14 (2014)
17. Gedikli, F., Jannach, D.: Improving recommendation accuracy based on item-specific tag preferences. ACM Transactions on Intelligent Systems and Technology **4**(1) (2013)
18. Gedikli, F., Jannach, D., Ge, M.: How should i explain? a comparison of different explanation types for recommender systems. International Journal of Human-Computer Studies **72**(4), 367–382 (2014)
19. Golbeck, J., Hendler, J., et al.: Filmtrust: Movie recommendations using trust in web-based social networks. In: Proc. of CCNC '06 (2006)
20. Harvey, M., Ludwig, B., Elsweiler, D.: You are what you eat: Learning user tastes for rating prediction. In: Proc. of SPIRE '13 (2013)
21. Hasler, D., Suesstrunk, S.E.: Measuring colorfulness in natural images. In: Human vision and electronic imaging VIII, vol. 5007, pp. 87–96. International Society for Optics and Photonics (2003)
22. Hauser, D.J., Schwarz, N.: Attentive turkers: Mturk participants perform better on online attention checks than do subject pool participants. Behavior Research Methods **48**(1) (2016)
23. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proc. of CVPR '16, pp. 770–778 (2016)
24. Howard, S., Adams, J., White, M., et al.: Nutritional content of supermarket ready meals and recipes by television chefs in the united kingdom: cross sectional study. BMJ **345** (2012)
25. J. Einhorn, H., N. Kleinmuntz, D., Kleinmuntz, B.: Linear regression and process-tracing models of judgment. Psychological Review **86**, 465–485 (1979)
26. Jannach, D., Adomavicius, G.: Recommendations with a purpose. In: Proc. of RecSys '16 (2016)
27. Jaro, M.A.: Advances in record-linkage methodology as applied to matching the 1985 census of tampa, florida. Journal of the American Statistical Association **84**(406), 414–420 (1989)

28. Jones, M.C., Downie, J.S., Ehmann, A.F.: Human similarity judgments: Implications for the design of formal evaluations. In: Proc. of ISMIR '07 (2007)
29. Kim, S.D., Lee, Y.J., Cho, H.G., Yoon, S.M.: Complexity and similarity of recipes based on entropy measurement. Indian Journal of Science and Technology **9**(26) (2016)
30. Knijnenburg, B.P., Willemsen, M.C., Gantner, Z., Soncu, H., Newell, C.: Explaining the user experience of recommender systems. User Modeling and User-Adapted Interaction **22**(4) (2012)
31. Kondrak, G.: N-gram similarity and distance. In: Proc. of SPIRE '05, pp. 115–126. Springer (2005)
32. Kusmierczyk, T., Nørvåg, K.: Online food recipe title semantics: Combining nutrient facts and topics. In: Proc. of CIKM '16 (2016)
33. Lee, J.H.: Crowdsourcing music similarity judgments using mechanical turk. In: Proc. of ISMIR '10 (2010)
34. Lops, P., De Gemmis, M., Semeraro, G.: Content-based recommender systems: State of the art and trends. In: Recommender systems handbook. Springer (2011)
35. Maksai, A., Garcin, F., Faltings, B.: Predicting online performance of news recommender systems through richer evaluation metrics. In: Proc. of RecSys '15 (2015)
36. Messina, P., Dominguez, V., Parra, D., Trattner, C., Soto, A.: Content-based artwork recommendation: integrating painting metadata with neural and manually-engineered visual features. User Modeling and User-Adapted Interaction (2018)
37. Milosavljevic, M., Navalpakkam, V., Koch, C., Rangel, A.: Relative visual saliency differences induce sizable bias in consumer choice. Journal of Consumer Psychology **22**(1) (2012)
38. Mirizzi, R., Di Noia, T., Ragone, A., Ostuni, V.C., Di Sciascio, E.: "movie recommendation with dbpedia". In: Proc. of IIR '12 (2012)
39. Oleszak, M.: Regularization: Ridge, lasso and elastic net (2018). URL `https://www.datacamp.com/community/tutorials/tutorial-ridge-lasso-elastic-net`
40. O'Mahony, M.P., Smyth, B.: Learning to recommend helpful hotel reviews. In: Proceedings of the Third ACM Conference on Recommender Systems, RecSys '09, pp. 305–308 (2009)
41. Ostuni, V.C., Di Noia, T., Di Sciascio, E., Mirizzi, R.: Top-n recommendations from implicit feedback leveraging Linked Open Data. In: Proc. of RecSys '13 (2013)
42. Peer, E., Vosgerau, J., Acquisti, A.: Reputation as a sufficient condition for data quality on amazon mechanical turk. Behavior Research Methods **46**(4) (2014)
43. van Pinxteren, Y., Geleijnse, G., Kamsteeg, P.: Deriving a recipe similarity measure for recommending healthful meals. In: Proc. of IUI '11 (2011)
44. Pu, P., Chen, L., Hu, R.: A user-centric evaluation framework for recommender systems. In: Proc. of RecSys '11 (2011)
45. Rokicki, M., Trattner, C., Herder, E.: The impact of recipe features, social cues and demographics on estimating the healthiness of online recipes. In: Proc. of ICWSM '18 (2018)

46. Rossetti, M., Stella, F., Zanker, M.: Contrasting offline and online results when evaluating recommendation algorithms. In: Proc. of RecSys '16 (2016)
47. Rublee, E., Rabaud, V., Konolige, K., Bradski, G.R.: ORB: An efficient alternative to SIFT or SURF. In: Proc. of ICCV '14, vol. 11, p. 2 (2011)
48. San Pedro, J., Siersdorfer, S.: Ranking and classifying attractiveness of photos in folksonomies. In: Proc. of WWW '09 (2009)
49. Sen, S., Vig, J., Riedl, J.: Tagommenders: connecting users to items through tags. In: Proc. of WWW '09 (2009)
50. Shannon, C.E.: A mathematical theory of communication. Bell system technical journal **27**(3), 379–423 (1948)
51. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
52. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z.: Rethinking the inception architecture for computer vision. In: Proc. of CVPR '16, pp. 2818–2826 (2016)
53. Teng, C.Y., Lin, Y.R., Adamic, L.A.: Recipe recommendation using ingredient networks. In: Proc. of WebSci '12
54. Tibshirani, R.: Regression shrinkage and selection via the lasso. Journal of the Royal Statistical Society: Series B (Methodological) **58**(1), 267–288 (1996)
55. Tran, T.N.T., Atas, M., Felfernig, A., Stettinger, M.: An overview of recommender systems in the healthy food domain. Journal of Intelligent Information Systems (2017)
56. Trattner, C., Elsweiler, D.: Food recommender systems: Important contributions, challenges and future research directions. arXiv preprint arXiv:1711.02760 (2017)
57. Trattner, C., Elsweiler, D.: Investigating the healthiness of internet-sourced recipes: implications for meal planning and recommender systems. In: Proc. of WWW '17, pp. 489–498 (2017)
58. Trattner, C., Kusmierczyk, T., Nørvåg, K.: Investigating and predicting online food recipe upload behavior. Information Processing & Management **56**(3), 654–673 (2019)
59. Trattner, C., Moesslang, D., Elsweiler, D.: On the predictability of the popularity of online recipes. EPJ Data Science **7**(1) (2018)
60. Tversky, A., Gati, I.: Studies of similarity. Cognition and categorization **1**(1978), 79–98 (1978)
61. Vargas, S., Castells, P.: Rank and relevance in novelty and diversity metrics for recommender systems. In: Proc. of RecSys '11 (2011)
62. Vig, J., Sen, S., Riedl, J.: Tagsplanations: Explaining recommendations using tags. In: Proceedings of IUI '09, pp. 47–56 (2009)
63. Wang, C., Agrawal, A., Li, X., Makkad, T., Veljee, E., Mengshoel, O., Jude, A.: Content-based top-n recommendations with perceived similarity. In: Proc. of SMC '17 (2017)
64. Wang, L., Li, Q., Li, N., Dong, G., Yang, Y.: Substructure similarity measurement in chinese recipes. In: Proc. of WWW '08 (2008)
65. Yang, L., Hsieh, C.K., Yang, H., Pollak, J.P., Dell, N., Belongie, S., Cole, C., Estrin, D.: Yum-me: a personalized nutrient-based meal recommender system. ACM Transactions on Information Systems **36**(1) (2017)
66. Yao, Y., Harper, F.M.: Judging similarity: a user-centric study of related item recommendations. In: Proc. of RecSys '18 (2018)

67. Yujian, L., Bo, L.: A normalized levenshtein distance metric. IEEE Transactions on Pattern Analysis and Machine Intelligence **29**(6) (2007)
68. Zhong, Y., Menezes, T.L.S., Kumar, V., Zhao, Q., Harper, F.M.: A field study of related video recommendations: Newest, most similar, or most relevant? In: Proc. of RecSys '18 (2018)
69. Ziegler, C.N., McNee, S.M., Konstan, J.A., Lausen, G.: Improving recommendation lists through topic diversification. In: Proc. of WWW '05 (2005)

Christoph Trattner is an Associate Professor at the University of Bergen in the Information Science Department. Previously, he was an Assistant Professor at MODUL University Vienna and an area manager at the Know-Center, Austria's research competence for data-driven business and Big Data analytics. His research interests include Data Science and Recommender Systems. He published two books and over 90 scientific articles.

Dietmar Jannach is a Professor of Information Systems at the University of Klagenfurt, Austria. He has worked on different areas of artificial intelligence, including recommender systems, model-based diagnosis, and knowledge-based systems. He is the leading author of a textbook on recommender systems and has authored more than hundred research papers, focusing on the application of artificial intelligence technology to practical problems.

Appendix

Table 10: Similarity metrics computed based on movie titles, images, plots, genres, director(s), release dates and stars.

Name	Metric	Explanation
Title:LV	$sim(r_i, r_j) = 1 - \lvert dist_{LEV}(r_i, r_j)\rvert$	Title Levenshtein Distance-based similarity
Title:JW	$sim(r_i, r_j) = 1 - \lvert dist_{JW}(r_i, r_j)\rvert$	Title Jaro-Winkler Distance-based similarity
Title:LCS	$sim(r_i, r_j) = 1 - \lvert dist_{LCS}(r_i, r_j)\rvert$	Title Longest Common Subsequence Distance-based similarity
Title:BI	$sim(r_i, r_j) = 1 - \lvert dist_{BI}(r_i, r_j)\rvert$	Title Bi-Gram Distance-based similarity
Title:LDA	$sim(r_i, r_j) = \frac{LDA(Title(r_i)) \cdot LDA(Title(r_j))}{\lVert LDA(Title(r_i))\rVert \lVert LDA(Title(r_j))\rVert}$	Title LDA Cosine-based similarity
Image:BR	$sim(r_i, r_j) = 1 - \lvert BR(r_i) - BR(r_j)\rvert$	Image Brightness Distance-based similarity
Image:SH	$sim(r_i, r_j) = 1 - \lvert SH(r_i) - SH(r_j)\rvert$	Image Sharpness Distance-based similarity
Image:CO	$sim(r_i, r_j) = 1 - \lvert CO(r_i) - CO(r_j)\rvert$	Image Contrast Distance-based similarity
Image:COL	$sim(r_i, r_j) = 1 - \lvert COL(r_i) - COL(r_j)\rvert$	Image Colorfulness Distance-based similarity
Image:EN	$sim(r_i, r_j) = 1 - \lvert EN(r_i) - EN(r_j)\rvert$	Image Entropy Distance-based similarity
Image:EMB	$sim(r_i, r_j) = \frac{EMB(r_i) \cdot EMB(r_j)}{\lVert EMB(r_i)\rVert \lVert EMB(r_j)\rVert}$	Image Embedding Cosine-based similarity
Plot:TFIDF	$sim(r_i, r_j) = \frac{TFIDF(Plot(r_i)) \cdot TFIDF(Plot(r_j))}{\lVert TFIDF(Plot(r_i))\rVert \lVert TFIDF(Plot(r_i))\rVert}$	Plot Text Cosine-based similarity (TFIDF = TF-IDF weighted vector)
Plot:LDA	$sim(r_i, r_j) = \frac{LDA(Plot(r_i)) \cdot LDA(Plot(r_j))}{\lVert LDA(Plot(r_i))\rVert \lVert LDA(Plot(r_j))\rVert}$	Plot LDA Cosine-based similarity (LDA = LDA vector)
Genre:JACC	$sim(r_i, r_j) = \frac{\{Genre(r_i)\} \cap \{Genre(r_j)\}}{\{Genre(r_i)\} \cup \{Genre(r_j)\}}$	Genre Jaccard-based similarity
Genre:LDA	$sim(r_i, r_j) = \frac{LDA(Genre(r_i)) \cdot LDA(Genre(r_j))}{\lVert LDA(Genre(r_i))\rVert \lVert LDA(Genre(r_j))\rVert}$	Genre LDA Cosine-based similarity (LDA = LDA vector)
Dir:JACC	$sim(r_i, r_j) = \frac{\{Dir(r_i)\} \cap \{Dir(r_j)\}}{\{Dir(r_i)\} \cup \{Dir(r_j)\}}$	Director(s) Jaccard-based similarity
Date:MD	$sim(r_i, r_j) = 1 - \lvert dist_{days}(r_i, r_j)\rvert$	Release Date distance-based similarity (unit = days)
Stars:JACC	$sim(r_i, r_j) = \frac{\{Stars(r_i)\} \cap \{Stars(r_j)\}}{\{Stars(r_i)\} \cup \{Stars(r_j)\}}$	Stars Jaccard-based similarity
SVD	$sim(r_i, r_j) = svd(r_i, r_j)$	SVD-based similarity based on ratings
Tags	$sim(r_i, r_j) = \frac{Tag(r_i) \cdot Tag(r_j)}{\lVert Tag(r_i)\rVert \lVert Tag(r_i)\rVert}$	Tag Genome Cosine-based similarity

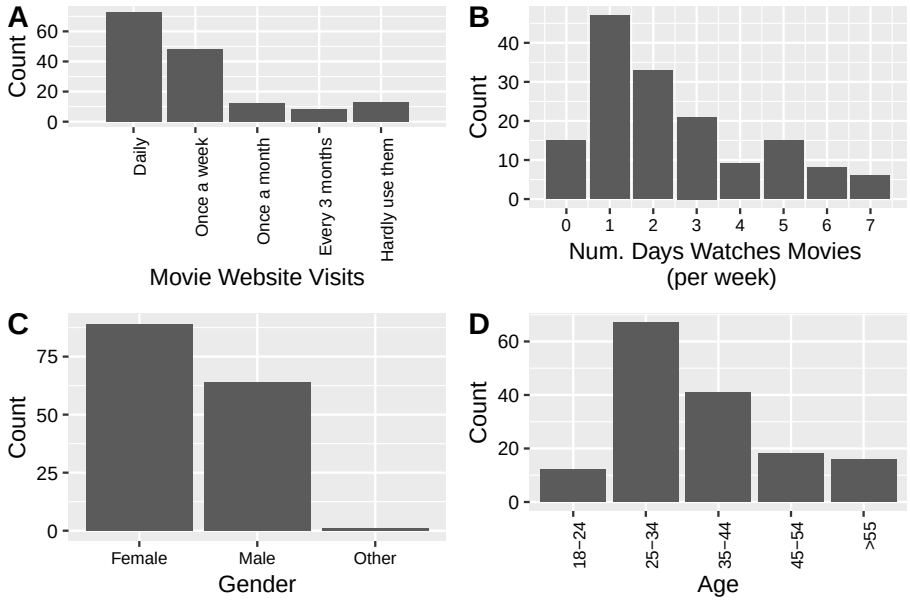

Fig. 11: Crowdworker characteristics (who passed the attention check) of the similarity assessment study (movie domain).

Table 11: Similarity metric correlation (Spearman) with user similarity estimates per cues when metrics are linearly combined (movie domain) using equal weights in the linear model. Note: All correlations are significant ($p < 0.001$), except for those marked with a † symbol.

	Humans	Title	Image	Plot	Genre	Dir.	Date	Stars	All
Humans	1	0.19	0.27	0.40	0.52	0.10	0.37	0.18	0.52
Title	–	1	0.17	0.23	0.27	$0.01^{\dagger}$	0.26	$0.01^{\dagger}$	0.41
Image	–	–	1	0.22	0.35	$0.03^{\dagger}$	0.26	0.07	0.41
Plot	–	–	–	1	0.43	$0.06^{\dagger}$	0.45	0.16	0.63
Genre	–	–	–	–	1	$0.04^{\dagger}$	0.49	0.14	0.72
Dir.	–	–	–	–	–	1	0.12	0.13	0.36
Date	–	–	–	–	–	–	1	0.15	0.79
Stars	–	–	–	–	–	–	–	1	0.23
All	–	–	–	–	–	–	–	–	1

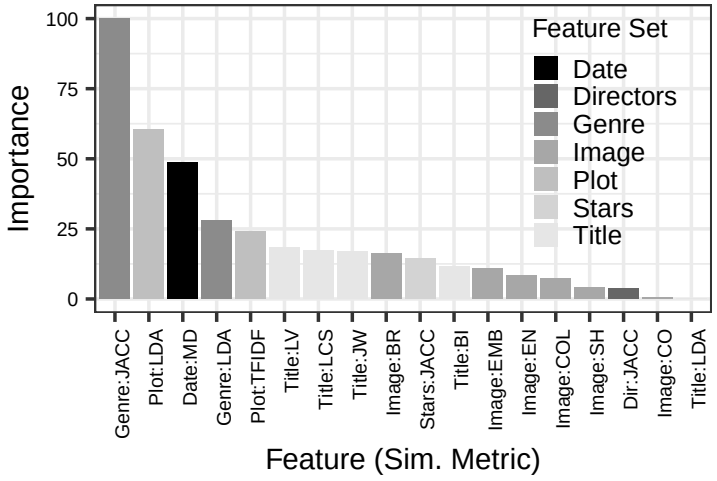

Fig. 12: Feature importance for the best performing Ridge regression model (movie domain).

Table 12: Results when considering additional features (movie domain).

Method	RMSE	R^2	MAE	ρ
(Instances = 1,395)				
Model performance (All features)				
All(Ridge)	0.8745	0.3628	0.6926	0.6019
All (Ridge) + additional features				
Movie Website Visits	0.8757	0.3615	0.6927	0.5999
Num. Days Watches Movie	0.8754	0.3667	0.6933	0.6049
Age	0.8764	0.3613	0.6931	0.6007
Gender	0.8770	0.3604	0.6946	0.5998
All User Characteristics	**0.8732**	**0.3682**	**0.6906**	**0.6064**

Table 13: Results when considering only one information cue at the time (movie domain).

Method	RMSE	R^2	MAE	ρ
(Instances = 1,395)				
Ridge Regression per Information Cue				
Title	1.0613	0.0615	0.8939	0.2437
Image	1.0460	0.0875	0.8681	0.2939
Plot	0.9786	0.2029	0.8105	0.4476
Genre	**0.9075**	**0.3140**	**0.7299**	**0.5593**
Stars	1.0729	0.0515	0.9041	0.2201
Directors	1.0885	0.0132	0.9149	0.1040
Date	1.0158	0.1385	0.8422	0.3717

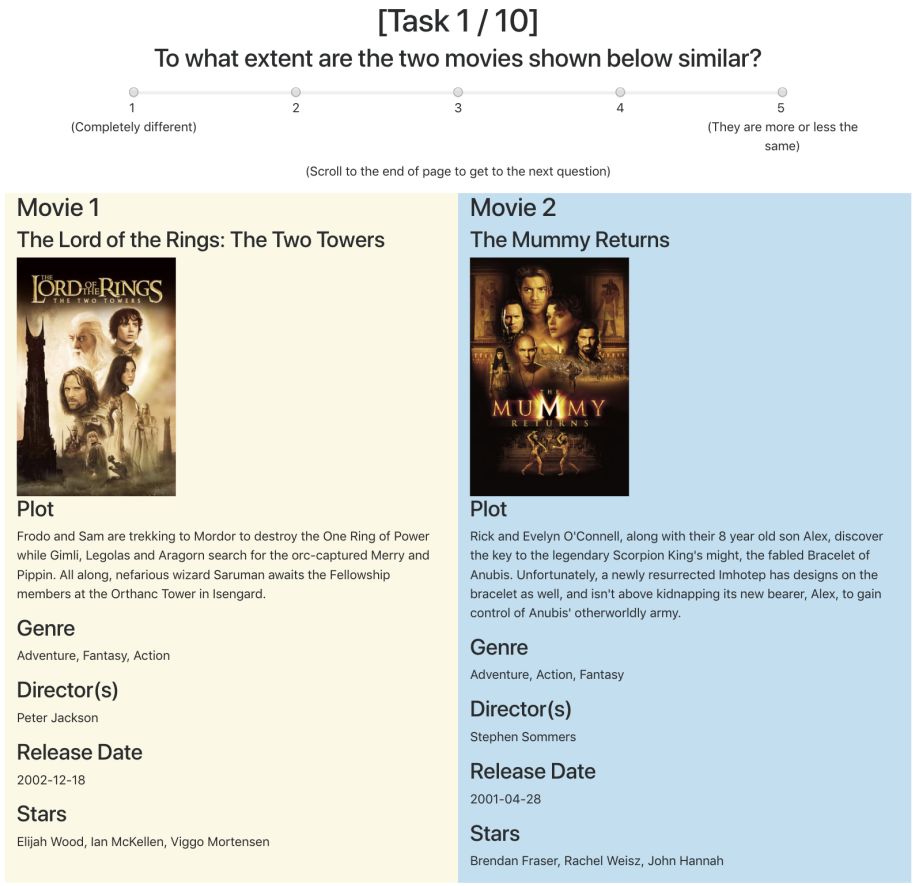

Fig. 13: *Study 1b*: Web interface to collect similarity judgements for movies. Regarding the choice of the features to be shown, note that it is not uncommon in practice to show more than just the title, image and short descriptions. ITunes, for example, shows the genre; IMDB shows also plot, directors and star ratings.

Fig. 14: Screen capture of *Study 2b* (movie domain).

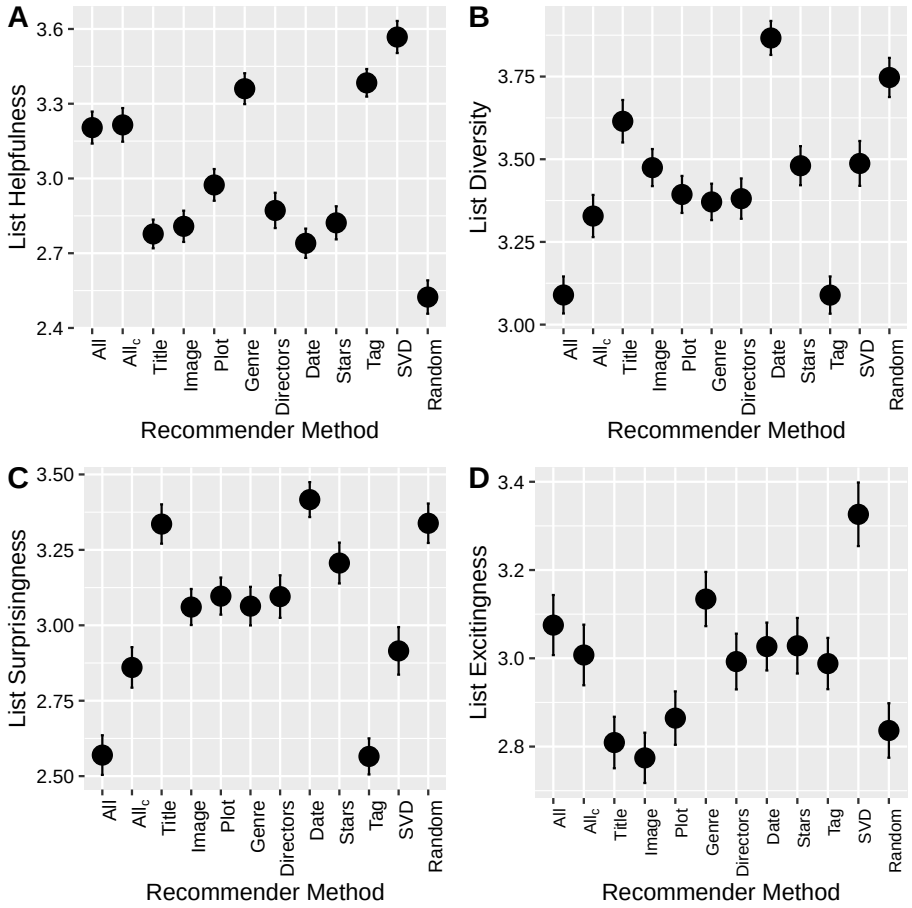

Fig. 15: *Study 2b*: (A) Helpfulness, (B) Diversity, (C) Surprisingness and (D) Excitingness of the recommended lists (means and std. errors). Scale: 1 (not at all) — 5 (totally agree).

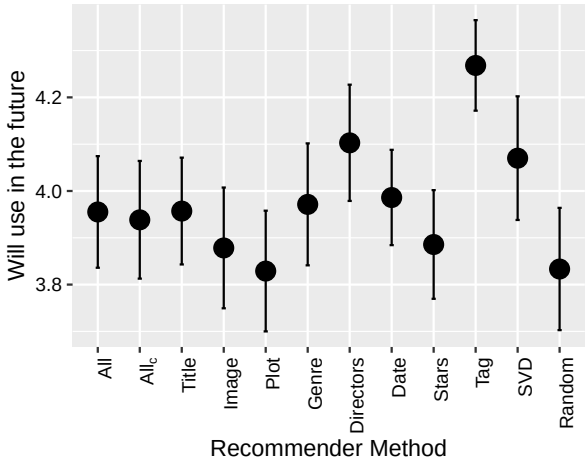

Fig. 16: *Study 2b* Intention to use the recommendation method in the future (means and std. errors). Scale: 1 (not at all) — 5 (totally agree).

Table 14: Survey questions for the recipe domain.

Question	Scale
Sim. Study (Study 1a)	
To what extent are the two recipes shown below similar?	Likert scale 1-5 (see Figure 1)
I looked at the *recipe title* to estimate the similarity between the recipes	Likert scale 1 (Totally disagree) - 5 (Totally agree)
I looked at the *recipe image* to estimate the similarity between the recipes	Likert scale 1 (Totally disagree) - 5 (Totally agree)
I looked at the *ingredient list* to estimate the similarity between the recipes	Likert scale 1 (Totally disagree) - 5 (Totally agree)
I looked at the *directions* to estimate the similarity between the recipes	Likert scale 1 (Totally disagree) - 5 (Totally agree)
Recommender Study (Study 2a)	
To what extent is this recipe similar to the reference recipe?	Likert scale 1 - 5 (see Figure 6)
How likely is it that you will try this recipe?	Likert scale 1 - 5 (see Figure 6)
On a scale from 1-5, to what extent is the recommended similar recipe list diverse?	Likert scale 1 (Not at all) - 5 (Completely diverse)
On a scale from 1-5, to what extent does the list contain recipes that were surprising?	Likert scale 1 (Not at all) - 5 (Completely surprising)
On a scale from 1-5, how exciting would you rate the recommended similar recipe list?	Likert scale 1 (Not at all) - 5 (Completely exciting)
On a scale from 1-5, how helpful would you rate the recommended similar recipe list?	Likert scale 1 (Not at all) - 5 (Completely helpful)
Sim./Recommender Study (Study 1a & 2a) - Demographic Questions	
What is your age?	Scale shown in Figure 2 (F)
What is your gender?	Scale shown in Figure 2 (E)
Which of the following best describe your dietary preferences?	Scale shown in Figure 2 (D)
How often do you visit recipe websites?	Scale shown in Figure 2 (A)
On average how many days per week do you eat a home cooked meal?	Scale shown in Figure 2 (B)
Please rate your experience of cooking	Scale shown in Figure 2 (C)
Recommender Study (Study 2a) - Additional Final Question	
Will you use similar recipe recommendations in the future (if provided)?	Likert scale 1 (Not at all) - 5 (Totally agree)

Table 15: Survey questions for the movie domain.

Question	Scale
Sim. Study (Study 1b)	
To what extent are the two movies shown below similar?	Likert scale 1-5 (see Figure 13)
I looked at the *movie title* to estimate the similarity between the movies	Likert scale 1 (Totally disagree) - 5 (Totally agree)
I looked at the *movie image* to estimate the similarity between the movies	Likert scale 1 (Totally disagree) - 5 (Totally agree)
I looked at the *plot* to estimate the similarity between the movies	Likert scale 1 (Totally disagree) - 5 (Totally agree)
I looked at the *genre* to estimate the similarity between the movies	Likert scale 1 (Totally disagree) - 5 (Totally agree)
I looked at the *directors(s)* to estimate the similarity between the movies	Likert scale 1 (Totally disagree) - 5 (Totally agree)
I looked at the *release date* to estimate the similarity between the movies	Likert scale 1 (Totally disagree) - 5 (Totally agree)
I looked at the *stars* to estimate the similarity between the movies	Likert scale 1 (Totally disagree) - 5 (Totally agree)
Recommender Study (Study 2b)	
To what extent is this movie similar to the reference movie?	Likert scale 1 - 5 (see Figure 14)
How likely is it that you will watch this movie?	Likert scale 1 - 5 (see Figure 14)
On a scale from 1-5, to what extent is the recommended similar movie list diverse?	Likert scale 1 (Not at all) - 5 (Completely diverse)
On a scale from 1-5, to what extent does the list contain movies that were surprising?	Likert scale 1 (Not at all) - 5 (Completely surprising)
On a scale from 1-5, how exciting would you rate the recommended similar movie list?	Likert scale 1 (Not at all) - 5 (Completely exciting)
On a scale from 1-5, how helpful would you rate the recommended similar movie list?	Likert scale 1 (Not at all) - 5 (Completely helpful)
Sim./Recommender Study (Study 1b & 2b) - Demographic Questions	
What is your age?	Scale shown in Figure 11 (D)
What is your gender?	Scale shown in Figure 11 (C)
Which of the following statements best describes your use of online movie services (e.g, Netflix, IMDB, etc.)?	Scale shown in Figure 11 (A)
On average how many days per week do you watch a movie?	Scale shown in Figure 11 (B)
Recommender Study (Study 2b) - Additional Final Question	
Will you use similar movie recommendations in the future (if provided)?	Likert scale 1 (Not at all) - 5 (Totally agree)

Table 16: Recipe and movie dataset content feature statistics.

Feature	Mean	Median	Min	Max
Recipe Dataset				
Number of words in the title	3.84	4	1	13
Number of characters in the title	25.32	24	5	82
Recipe image brightness	0.50	0.49	0.15	0.89
Recipe image sharpness	0.18	0.16	0.02	0.78
Recipe image contrast	0.17	0.17	0.01	0.48
Recipe image colorfulness	0.24	0.24	0.06	0.52
Recipe image entropy	0.95	0.95	0.49	1
Number of ingredients in the recipes	9.34	9	2	30
Number of words used in the ingredients description	26.22	25	3	97
Number of characters used in the ingredients description	175.79	166	22	673
Number of words used in the directions text	110.75	100	14	471
Number of characters used in the directions text	624.08	561	86	2643
Movie Dataset				
Number of words in the title	2.79	2	1	14
Number of characters in the title	15.58	14	1	83
Cover image brightness	0.41	0.39	0.03	0.95
Cover image sharpness	0.23	0.21	0.03	1
Cover image contrast	0.08	0.07	0.01	0.22
Cover image colorfulness	0.24	0.23	0	0.71
Cover image entropy	0.83	0.87	0.17	0.99
Number of words used in the plot description	51.04	50	7	167
Number of characters used in the plot description	298.60	294	53	973
Number of genres	2.71	3	1	7
Number of directors	1.08	1	1	12
Release date (year)	1993.78	1997	1922	2017
Number of stars (top-3)	2.99	3	0	3