The Tag Genome Dataset for Books

Denis Kotkov
Department of Computer Science
University of Helsinki
Helsinki, Finland
kotkov.denis.ig@gmail.com

Alan Medlar
Department of Computer Science
University of Helsinki
Helsinki, Finland
alan.j.medlar@helsinki.fi

Alexandr Maslov
Faculty of Science and Engineering
Åbo Akademi University
Turku, Finland
maslov314@gmail.com

Umesh Raj Satyal
Faculty of Science and Engineering
Åbo Akademi University
Turku, Finland
satyalumesh@gmail.com

Mats Neovius
Faculty of Science and Engineering
Åbo Akademi University
Turku, Finland
mats.neovius@abo.fi

Dorota Glowacka
Department of Computer Science
University of Helsinki
Helsinki, Finland
dorota.glowacka@helsinki.fi

ABSTRACT

Attaching tags to items, such as books or movies, is found in many online systems. While a majority of these systems use binary tags, continuous item-tag relevance scores, such as those in tag genome, offer richer descriptions of item content. For example, tag genome for movies assigns the tag "gangster" to the movie "The Godfather (1972)" with a score of 0.93 on a scale of 0 to 1. Tag genome has received considerable attention in recommender systems research and has been used in a wide variety of studies, from investigating the effects of recommender systems on users to generating ideas for movies that appeal to certain user groups.

In this paper, we present tag genome for books, a dataset containing book-tag relevance scores, where a significant number of tags overlap with those from tag genome for movies. To generate our dataset, we designed a survey based on popular books and tags from the Goodreads dataset. In our survey, we asked users to provide ratings for how well tags applied to books. We generated book-tag relevance scores based on user ratings along with features from the Goodreads dataset. In addition to being used to create book recommender systems, tag genome for books can be combined with the tag genome for movies to tackle cross-domain problems, such as recommending books based on movie preferences.

CCS CONCEPTS

• **Information systems** → **Social tagging**; **Recommender systems**; *Users and interactive retrieval.*

KEYWORDS

recommender systems; tagging; books; dataset; tag genome; item-tag rating; tag relevance

ACM Reference Format:
Denis Kotkov, Alan Medlar, Alexandr Maslov, Umesh Raj Satyal, Mats Neovius, and Dorota Glowacka. 2022. The Tag Genome Dataset for Books. In *Proceedings of the 2022 ACM SIGIR Conference on Human Information Interaction and Retrieval (CHIIR '22), March 14–18, 2022, Regensburg, Germany.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3498366.3505833

CHIIR '22, March 14–18, 2022, Regensburg, Germany

ACM ISBN 978-1-4503-9186-3/22/03.
https://doi.org/10.1145/3498366.3505833

1 BACKGROUND

Tags are keywords that concisely describe items, such as books, movies or audio recordings [17]. For example, the book "Harry Potter and the Philosopher's Stone" could be tagged with "magic", "wizard" and "school". Tagging is a popular feature in many online systems: MovieLens [5] allows users to attach tags to movies [24]. In Goodreads [2], users can put a book on a shelf, which can be regarded as tagging, and Instagram [3] supports hashtags for pictures and videos. While binary tags (tags that either apply to an item or not) are the most common kind of tag, tags can also include continuous relevance scores, indicating the degree to which a tag applies to a given item. For example, although "Harry Potter and the Philosopher's Stone" is mostly about wizards, it also contains elements of family drama. Therefore, the tag "family drama" could be applied with a relevance score of 0.3 on a scale from 0 (does not apply) to 1 (applies very strongly).

To the best of our knowledge, tag genome for movies (movie-tag genome from now on) is the only publicly available dataset that contains tag relevance scores [26]. Movie-tag genome encodes each movie as a vector, where each value corresponds to a relevance score between 0 and 1. To generate this dataset, Vig et al. [26] collected data associated with tags and movies, such as tag applications, movie ratings and reviews, and asked MovieLens users to indicate the degree to which each tag applied to a given movie. The authors then built a machine learning model that was trained on features extracted from the collected data and user responses to predict movie-tag relevance scores. This model was recently improved by Kotkov et al. [16].

Movie-tag genome has received considerable attention from the recommender systems research community. It has been used to calculate similarities between movies and describe them with tags [26], to investigate filter bubbles [20] and to understand how users evaluate lists of recommendations [13]. It has also been used as a baseline for describing movies [10, 21], differences between movies [12] and groups of movies to bootstrap new users [9]. It has also served as ground truth for movie similarity [6, 7], to create novel recommendation algorithms [14] and as the basis for critiquing

recommender systems [18, 26]. Movie-tag genome has also been used to measure diversity and serendipity [19], movie similarity to investigate serendipity [15], effects of exploration on users [23] and how users perceive movie similarity [31]. Vo & Soh even used movie-tag genome to generate descriptions of movies that would appeal to certain user groups [27].

In this paper, we present the tag genome for books (or simply book-tag genome), a dataset of book-tag relevance scores. While book-tag genome can be used to create book recommender systems, it was specifically designed to be combined with movie-tag genome via a substantial overlap in tags, allowing researchers to investigate cross-domain recommendation problems. Examples of such systems include making cross-domain recommendations at the item level [8], where users could be recommended books based on their interests in movies or bundles of movies and books [8]. Second, as tag genome supports critiquing, comparison and description tasks [26], a user could compare movies with books using tags or be presented with a description of a movie based on similar books they have already read. Third, researchers could conduct cross-domain studies to investigate the effects of recommender systems on users' movie and book consumption behavior. Lastly, the data could be used to improve item-tag relevance prediction via transfer learning.

In brief, to build book-tag genome, we selected popular books and tags from the Goodreads dataset [28, 29], which contains book ratings, book additions to shelves and reviews (Section 2.1.1). Based on this collection of books and tags, we designed a survey, where users were asked to indicate the degrees to which tags applied to books (Section 2.1.2). We used user answers from the survey, along with features extracted from the Goodreads dataset, to generate book-tag relevance scores (Section 3). We freely distribute book-tag genome, together with raw data (Table 1) and extracted features (Section 3) needed to generate relevance scores. The dataset is licensed under the Creative Commons Attribution-NonCommercial 3.0 License and is available at https://grouplens.org/datasets/book-genome.

1.1 Terminology

In this paper, we refer to the following datasets: (1) **Item-tag genome dataset**, includes items, tags and item-tag relevance scores (generated by a relevance score prediction method to indicate the degree to which a tag applies to an item). (2) **Item raw dataset**, includes items, tags, item-tag ratings and data related to items, such as tag applications, user reviews and item ratings. (3) **Item-tag rating dataset**, a subset of item raw dataset and only includes items, tags and item-tag ratings from user surveys. Each dataset has two versions: books and movies, i.e. movie-tag genome and book-tag genome. Item and tag data in the book raw dataset is a subset of the Goodreads dataset [28, 29], while the movie raw dataset contains a combination of data from MovieLens [5] and IMDB [4].

2 COLLECTING BOOK-TAG RELEVANCE RATINGS

In this section, we describe book-tag relevance ratings and their collection process.

2.1 Survey construction

2.1.1 Data selection. The tags we used in our survey were based on shelves from the Goodreads dataset [28, 29]. The Goodreads dataset contains information on books and user interactions from the Goodreads website [2]. The dataset contains 2,360,655 books and 15,700,468 user reviews. Each review contains a user rating on a scale of 1 to 5 stars with the granularity of 1 star (549,961 ratings are unknown). Books can also belong to "shelves", i.e. lists of books named and organized by a user.

When selecting tags, we had the following objectives: (i) tags should be applicable to the contents of the book, (ii) tags should correspond to shelves in the Goodreads dataset so that we can extract additional features for prediction of relevance scores, and (iii) tags should intersect with those from the movie-tag genome.

We performed the following procedure to select tags and books for inclusion in the survey. We first prepared sets of books and shelves for extraction: (1) We extracted the 10,000 most popular books translated into or written in English from the Goodreads dataset and 22,059 unique shelves, to which these books were added. (2) We removed punctuation and extracted lemmas (base forms of words) from shelf names.

Next, we detected an intersection between movie-tags and shelves: (1) We lemmatized tags from the movie-tag genome, because they contained different forms of the same word, e.g. "alien" and "aliens". (2) We matched movie-tags and shelves by their lemmas. (3) Following [26], we removed shelves associated with fewer than 10 books from the intersection to exclude obscure shelves. We also excluded movie-tags that were unrelated to book content, such as "best war films", "book was better" and "notable soundtrack", which resulted in 741 shelves corresponding to 616 movie-tags.

We combined the 1,000 most popular shelves with 741 shelves that corresponded to movie-tags to ensure that tags described books well and were compatible with the movie-tag genome. This resulted in 1,345 lemmatized shelf names, which corresponded to 1,667 shelves and had an overlap with 616 movie-tags.

Finally, we used lemmatized shelf names as book-tags and prepared them for the survey: (1) We manually excluded tags which (i) did not describe book content, such as "paperback", "1001 books to read" and "series", or (ii) represented only metadata, such as the author's name or the year of publication. Two raters marked the list of tags for exclusion with an almost perfect agreement (unweighted Cohen's kappa: 0.88, p-value < 10^{-15}). (2) We edited tags that were (i) erroneously shortened during lemmatization, e.g. "u history" was changed back to "us history", or (ii) described the same concept, e.g. "ya" and "young adult" or "contemporary fiction" and "fiction contemporary". This resulted in 826 tags corresponding to 1,207 shelves and 582 movie-tags. (3) We removed tags that covered multiple concepts, e.g. "paranormal young adult" includes the tags "paranormal" and "young adult". When removing these tags, we took into account the number of books at the intersection of multi-concept tags and their constituent sub-tags. For example, 99% of books tagged with "paranormal fantasy" were also tagged with both "paranormal" and "fantasy", leading to the removal of the "paranormal fantasy" tag. However, we kept the tag "dark humor" because while 94% of books tagged "dark humor" were also tagged

"humor", only 57% of them were tagged "dark", suggesting that the combination of the two tags has a different meaning.

After filtering, our subset consisted of 9,373 books due to the removal of duplicates and 727 book-tags, which corresponded to 1,046 shelves. Filtering tags is particularly important because it allows us to avoid collecting superfluous data in the survey and predicting redundant relevance scores later on. Following [26], we constructed a set of relatively popular tags (all above 0.25 quantile, most above 0.9 quantile). To simplify the combination of book and movie datasets in future research, we manually detected one-to-one relationships between book-tags and movie-tags, which resulted in 512 book-tags that correspond to movie-tags and 215 that do not.

2.1.2 Survey questions. The survey was conducted on Amazon Mechanical Turk [1]. Before starting the survey, users were presented with information about the goals of the survey and were asked to rate a few book-tag pairs as a tutorial. Users were then asked to select 10 books they have read using free-form text queries to search the list of available books. Users were also presented with three books as recommendations: the two books with the fewest ratings in the survey and one fake book to catch fraudulent survey responses.

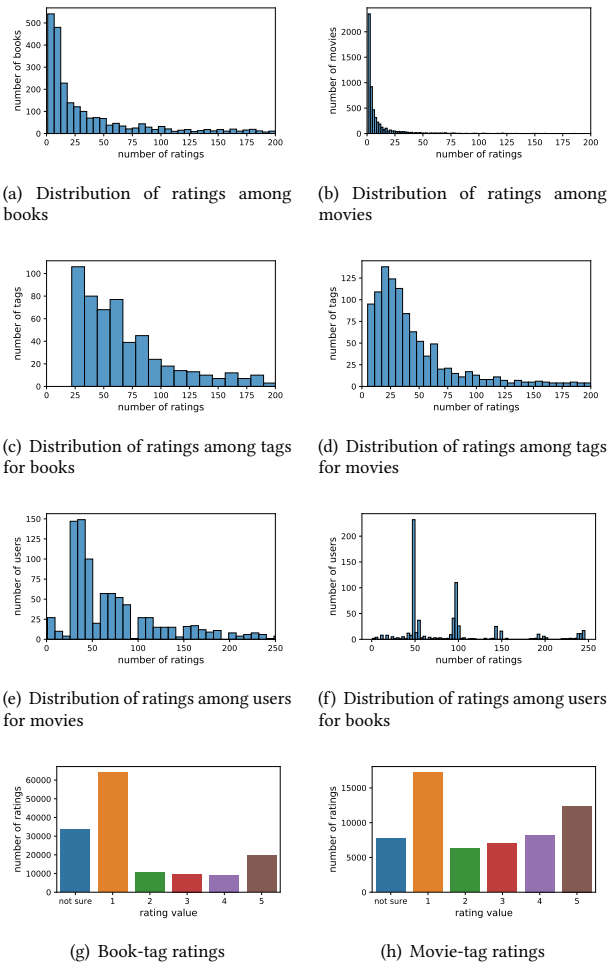
Next, we asked users to indicate the degree to which a tag applies to books that the user has selected on a scale of 1 ("strongly disagree") to 5 ("strongly agree"). If the user was unsure about the applicability of a tag, they could select "not sure" (which is outside of the 1-5 scale). Users were asked to provide at least 30 ratings. If the user selected fewer than ten books, they were asked to rate more tags related to their selected books so that the overall number of ratings was at least 30. Optionally, users could continue rating book-tag pairs after providing the required number of ratings.

It is easier for users to rate a book-tag pair when they see extreme examples of the scale (the so-called anchoring effect [21, 25]). Similar to [26], we included extreme examples in the book-tag questions. We picked tags that differed the most for the selected books. For example, if the user picks "Harry Potter and the Sorcerer's Stone" by J.K. Rowling, "Winnie the Pooh" by A.A. Milne and "Fight Club" by Chuck Palahniuk, they will likely be asked to rate tags, such as "animals" because "Winnie the Pooh" features animals and "Fight Club" does not, "dark" due to the inclusion of "Fight Club", and "wizards" due to the inclusion of "Harry Potter".

To avoid low quality ratings, we excluded users that: (i) selected obviously wrong answers in the tutorial, (ii) indicated that they cannot find any books they have read, and (iii) selected the fake (non-existent) book added to the list of recommendations.

2.2 Data collection

For each survey respondent, our goal was to select tags for them to rate that were predicted to have very high and very low relevance scores within the set of books they selected (the anchoring effect [21, 25]). To achieve this goal, we needed to be able to estimate book-tag relevance scores. For the first portion of ratings, we predicted book-tag relevance based on additions of books to shelves. After receiving 5,029 ratings from 43 users on 233 books and 192 tags (4,373 book-tag pairs), we extracted features (Section 3) and trained the multilevel regression model described in [26] to provide more precise relevance predictions.



(a) Distribution of ratings among books

(b) Distribution of ratings among movies

(c) Distribution of ratings among tags for books

(d) Distribution of ratings among tags for movies

(e) Distribution of ratings among users for movies

(f) Distribution of ratings among users for books

(g) Book-tag ratings

(h) Movie-tag ratings

Figure 1: Characteristics of our book-tag rating and the movie-tag rating datasets

Our goal was to cover as many books and tags as possible to provide a wide variety of examples for training models. We, therefore, avoided collecting further ratings for a particular book-tag pair after obtaining at least five ratings. We ran our survey in batches of 4-20% of all users included in the survey. After each batch, we analyzed the results and applied different strategies to flatten the distribution of ratings among books and book-tag pairs, such as hiding books which had already received a sufficient number of ratings and reranking book-tag pairs in the survey based on the number of ratings per book-tag pair.

2.3 Collected relevance ratings

Overall, our book-tag rating dataset consists of 145,825 ratings from 986 users on 2,535 books and 727 tags (116,694 book-tag pairs) including 33,589 "not sure" ratings. We compared our dataset to the movie-tag rating dataset, which we received from the authors of [26] and consists of 58,903 ratings from 679 users on 5,546 movies and 1,094 tags (45,914 movie-tag pairs) including 7,740 "not sure"

Table 1: Summary of raw datasets

Type of data	Movie raw dataset	Book raw dataset
Items	84,661 movies	9,374 books
Tag applications	Applications of tags to movies (832,896 applications)	Additions of books to shelves (46,750,123 additions)
User reviews	Reviews written by users for movies in IMDB (2,624,608 reviews)	Reviews written by users for books in Goodreads (5,307,626 reviews)
Item ratings	Movie ratings from 1 to 5 (0.5 step) (28,490,116 ratings)	Book ratings from 1 to 5 (1 step) (5,152,656 ratings)
Item-tag ratings	Movie-tag ratings from 1 to 5 (1 step) (58,903 ratings)	Book-tag ratings from 1 to 5 (1 step) (145,825 ratings)

ratings. Table 1 summarizes characteristics of tag genome for books and movies. Figure 1 shows the main characteristics of our book-tag rating dataset. The book-tag ratings dataset differs from the movie-tag rating dataset in the following ways:

(1) Our book-tag rating dataset contains more than twice as many ratings as the movie-tag rating dataset.
(2) The movie-tag rating dataset covers around twice as many items but only half the number of ratings compared to our dataset. Thus, our dataset provides richer data on each item (Figures 1(a) and 1(b)). The median number of ratings for books is 19, while for movies it is 3.
(3) Our dataset contains fewer tags than the movie-tag rating dataset (Figures 1(c) and 1(d)), but our manual tag filtering resulted in fewer duplicates, misspellings and tags with overlapping meanings. Our dataset contains richer information on tags: the minimum number of ratings per tag is 22 and the median is 76, while for the movie dataset these numbers are 5 and 30, respectively.
(4) Users involved in our survey provided more ratings on average (average: 148, median: 60) compared to users in the movie survey (average: 86, median: 54) (Figures 1(e) and 1(f)).
(5) The distributions of ratings are similar between the two datasets (Figures 1(g) and 1(h)), except that in our case the proportion of values with a rating of 1 is higher. This is related to our data collection method: we picked tags that were the most different for books chosen by the user. This strategy guarantees that tags apply to at least one of the selected books, but not to all the others. Whereas, during the collection of movie-tag ratings, the authors mostly showed movies that applied to the current tag at least partially [26].
(6) Our dataset has a higher number (1,251) of item-tag pairs with at least five ratings than the movie-tag rating dataset (717). A majority of item-tag pairs in both datasets have only one rating (66% for both datasets).

3 GENERATING RELEVANCE SCORES

To choose a method for generating book-tag scores, we compared several state-of-the-art tag relevance prediction methods. We extracted the following features from the book raw dataset (Table 1) using code published in [16]: (1) *tag-applied*(t, i) – a binary variable indicating whether tag t has been applied to item i; (2) *tag-lsi-sim*(t, i) - similarity between tag t and item i based on latent semantic indexing [11], where each document is a set of tags applied to item i; (3) *text-freq-nostem*(t, i) - number of times tag t appears in user reviews of item i; (4) *text-freq*(t, i) - the same as in (3), but calculated after applying word stemming with Porter stemmer [22] as implemented in [30]; (5) *text-lsi-sim*(t, i) - similarity between tag t and item i based on latent semantic indexing [11], where each document is the set of words in user reviews of item i; (6) *avg-rating*(t, i) - mean rating of item i; (7) *rating-sim*(t, i) - cosine similarity between ratings of item i and aggregated ratings of items tagged with tag t (added to shelves for books); (8) *regress-tag*(t, i) - predicted score based on a regression model with *tag-applied*(t, i) as the output variable and the other features as the input variables.

Table 2: Mean Absolute Error (MAE) for books with 95% confidence intervals

Method	MAE
Average	1.402 ± 0.003
Glmer	0.851 ± 0.005
TagDL	0.752 ± 0.008

In our experiment, we used 10-fold cross validation and evaluated the prediction methods with extracted features as predictor variables and collected book-tag ratings as the target variable. To measure the performance of our methods, we used Mean Absolute Error (MAE). We compared the following methods: (1) **Average** - average relevance score in the training dataset; (2) **Glmer** - the multilevel nonlinear regression model from [26] and (3) **TagDL** - the multi-layer perceptron-based model from [16].

Table 2 shows that TagDL outperforms the Glmer model (around 11% improvement, Mann–Whitney U test, p-value < 10^{-4}), while the average baseline has the lowest performance, which replicates the results from [16], except the improvement of TagDL compared to Glmer is higher for books than movies.

4 CONCLUSION

In this paper, we presented the book-tag genome, a novel dataset describing the degree to which various tags apply to books. In particular, we created this dataset so that many of the tags (512/727) correspond to tags in movie-tag genome to allow research in cross-domain recommendations and associated applications. We made the book-tag genome dataset including the data for its generation publicly available to encourage future research on this topic.

ACKNOWLEDGMENTS

We would like to thank Mengting Wan for allowing us to publish this dataset and organizations that supported this work: the Academy of Finland, grant 309495 (the LibDat project) and the Academy of Finland Flagship programme: Finnish Center for Artificial Intelligence FCAI.

REFERENCES

[1] [n.d.]. Amazon Mechanical Turk. mturk.com. [Online; accessed 09-June-2021].

[2] [n.d.]. Goodreads | Meet your next favorite book. https://www.goodreads.com/. [Online; accessed 09-June-2021].

[3] [n.d.]. Instagram. https://instagram.com/. [Online; accessed 09-June-2021].

[4] [n.d.]. Internet Movie Database. https://imdb.com/. [Online; accessed 09-June-2021].

[5] [n.d.]. MovieLens. Non-commercial, personalized movie recommendations. https://movielens.org/. [Online; accessed 09-June-2021].

[6] Konstantinos Bougiatiotis and Theodoros Giannakopoulos. 2016. Content representation and similarity of movies based on topic extraction from subtitles. In *Proceedings of the 9th Hellenic Conference on Artificial Intelligence*. 1–7.

[7] Konstantinos Bougiatiotis and Theodoros Giannakopoulos. 2018. Enhanced movie content similarity based on textual, auditory and visual information. *Expert Systems with Applications* 96 (2018), 86–102.

[8] Iván Cantador, Ignacio Fernández-Tobías, Shlomo Berkovsky, and Paolo Cremonesi. 2015. Cross-domain recommender systems. In *Recommender systems handbook*. Springer, 919–959.

[9] Shuo Chang, F Maxwell Harper, and Loren Terveen. 2015. Using groups of items for preference elicitation in recommender systems. In *Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing*. 1258–1269.

[10] Shuo Chang, F Maxwell Harper, and Loren Gilbert Terveen. 2016. Crowd-based personalized natural language explanations for recommendations. In *Proceedings of the 10th ACM Conference on Recommender Systems*. 175–182.

[11] Scott Deerwester, Susan T Dumais, George W Furnas, Thomas K Landauer, and Richard Harshman. 1990. Indexing by latent semantic analysis. *Journal of the American society for information science* 41, 6 (1990), 391–407.

[12] Joaquin Derrac and Steven Schockaert. 2015. Inducing semantic relations from conceptual spaces: a data-driven approach to plausible reasoning. *Artificial Intelligence* 228 (2015), 66–94.

[13] Michael D Ekstrand, F Maxwell Harper, Martijn C Willemsen, and Joseph A Konstan. 2014. User perception of differences in recommender algorithms. In *Proceedings of the 8th ACM Conference on Recommender systems*. 161–168.

[14] Bu Sung Kim, Heera Kim, Jaedong Lee, and Jee-Hyong Lee. 2014. Improving a recommender system by collective matrix factorization with tag information. In *2014 Joint 7th International Conference on Soft Computing and Intelligent Systems (SCIS) and 15th International Symposium on Advanced Intelligent Systems (ISIS)*. IEEE, 980–984.

[15] Denis Kotkov, Joseph A Konstan, Qian Zhao, and Jari Veijalainen. 2018. Investigating serendipity in recommender systems based on real user feedback. In *Proceedings of the 33rd Annual ACM Symposium on Applied Computing*. 1341–1350.

[16] Denis Kotkov, Alexandr Maslov, and Mats Neovius. 2021. Revisiting the Tag Relevance Prediction Problem. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Virtual Event, Canada) *(SIGIR '21)*. Association for Computing Machinery, New York, NY, USA, 1768–1772. https://doi.org/10.1145/3404835.3463019

[17] Paul Lamere. 2008. Social tagging and music information retrieval. *Journal of new music research* 37, 2 (2008), 101–114.

[18] Benedikt Loepp, Tim Donkers, Timm Kleemann, and Jürgen Ziegler. 2019. Interactive recommending with tag-enhanced matrix factorization (TagMF). *International Journal of Human-Computer Studies* 121 (2019), 21–41.

[19] Tien T Nguyen, F Maxwell Harper, Loren Terveen, and Joseph A Konstan. 2018. User personality and user satisfaction with recommender systems. *Information Systems Frontiers* 20, 6 (2018), 1173–1189.

[20] Tien T Nguyen, Pik-Mai Hui, F Maxwell Harper, Loren Terveen, and Joseph A Konstan. 2014. Exploring the filter bubble: the effect of using recommender systems on content diversity. In *Proceedings of the 23rd international conference on World wide web*. ACM, 677–686.

[21] Tien T Nguyen, Daniel Kluver, Ting-Yu Wang, Pik-Mai Hui, Michael D Ekstrand, Martijn C Willemsen, and John Riedl. 2013. Rating support interfaces to improve user experience and recommender accuracy. In *Proceedings of the 7th ACM Conference on Recommender Systems*. 149–156.

[22] Martin F Porter. 1980. An algorithm for suffix stripping. *Program* (1980).

[23] Tobias Schnabel, Paul N Bennett, Susan T Dumais, and Thorsten Joachims. 2018. Short-term satisfaction and long-term coverage: Understanding how users tolerate algorithmic exploration. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*. 513–521.

[24] Shilad Sen, F Maxwell Harper, Adam LaPitz, and John Riedl. 2007. The quest for quality tags. In *Proceedings of the 2007 international ACM conference on Supporting group work*. 361–370.

[25] Amos Tversky and Daniel Kahneman. 1974. Judgment under uncertainty: Heuristics and biases. *science* 185, 4157 (1974), 1124–1131.

[26] Jesse Vig, Shilad Sen, and John Riedl. 2012. The Tag Genome: Encoding Community Knowledge to Support Novel Interaction. *ACM Trans. Interact. Intell. Syst.* 2, 3, Article 13 (Sept. 2012), 44 pages. https://doi.org/10.1145/2362394.2362395

[27] Thanh Vinh Vo and Harold Soh. 2018. Generation meets recommendation: proposing novel items for groups of users. In *Proceedings of the 12th ACM Conference on Recommender Systems*. 145–153.

[28] Mengting Wan and Julian J. McAuley. 2018. Item recommendation on monotonic behavior chains. In *Proceedings of the 12th ACM Conference on Recommender Systems, RecSys 2018, Vancouver, BC, Canada, October 2-7, 2018*, Sole Pera, Michael D. Ekstrand, Xavier Amatriain, and John O'Donovan (Eds.). ACM, 86–94. https://doi.org/10.1145/3240323.3240369

[29] Mengting Wan, Rishabh Misra, Ndapa Nakashole, and Julian J. McAuley. 2019. Fine-Grained Spoiler Detection from Large-Scale Review Corpora. (2019), 2605–2610. https://doi.org/10.18653/v1/p19-1248

[30] Nianwen Xue, Edward Bird, et al. 2011. Natural language processing with python. *Natural Language Engineering* 17, 3 (2011), 419.

[31] Yuan Yao and F Maxwell Harper. 2018. Judging similarity: a user-centric study of related item recommendations. In *Proceedings of the 12th ACM Conference on Recommender Systems*. ACM, 288–296.