Revisiting the Tag Relevance Prediction Problem

Kotkov, Denis

Association for Computing Machinery
2021-07-11

Kotkov , D , Maslov , A & Neovius , M 2021 , Revisiting the Tag Relevance Prediction Problem . in SIGIR '21: Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval . Association for Computing Machinery , þÿpp. 1768 1772 , International ACM SIGIR conference on research and d
Information Retrieval , Montréal , Quebec , Canada , 11/07/2021 . https://doi.org/10.1145/3404835.3463019

http://hdl.handle.net/10138/337804
https://doi.org/10.1145/3404835.3463019

Revisiting the Tag Relevance Prediction Problem

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

Alexandr Maslov
Faculty of Science and Engineering
Åbo Akademi University
Turku, Finland
alexandr.maslov@abo.fi

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

ABSTRACT

Traditionally, recommender systems provide a list of suggestions to a user based on past interactions with items of this user. These recommendations are usually based on user preferences for items and generated with a delay. Critiquing recommender systems allow users to provide immediate feedback to recommendations with tags and receive a new set of recommendations in response. However, these systems often require rich item descriptions that contain relevance scores indicating the strength, with which a tag applies to an item. For example, this relevance score could indicate how violent the movie "The Godfather" is on a scale from 0 to 1. Retrieving these data is a very demanding process, as it requires users to explicitly indicate the degree to which a tag applies to an item. This process can be improved with machine learning methods that predict tag relevance. In this paper, we explore the dataset from a different study, where the authors collected relevance scores on movie-tag pairs. In particular, we define the tag relevance prediction problem, explore the inconsistency of relevance scores provided by users as a challenge of this problem and present a method, which outperforms the state-of-the-art method for predicting tag relevance. We found a moderate inconsistency of user relevance scores. We also found that users tend to disagree more on subjective tags, such as "good acting", "bad plot" or "quotable" than on objective tags, such as "animation", "cars" or "wedding", but the disagreement of users regarding objective tags is also moderate.

CCS CONCEPTS

• **Information systems** → **Social tagging**; **Recommender systems**; *Users and interactive retrieval*; • **Computing methodologies** → *Neural networks.*

KEYWORDS

recommender systems; critiquing recommender systems; tag relevance prediction; tagging

ACM Reference Format:
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 (SIGIR '21), July 11–15, 2021, Virtual Event, Canada.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3404835.3463019

SIGIR '21, July 11–15, 2021, Virtual Event, Canada.
© 2021 Copyright held by the owner/author(s).
ACM ISBN 978-1-4503-8037-9/21/07.
https://doi.org/10.1145/3404835.3463019

1 INTRODUCTION

Recommender systems assist users in choosing items when the number of these items is overwhelming [14]. Traditionally, recommender systems provide a list of suggestions to a user based on past interactions with items of this user. These recommendations are usually based on users indicating their preference for items and generated with a delay. Conversational recommender systems allow users to provide immediate feedback to recommendations and receive a new set of recommendations in response. In this way, the process of recommendation corresponds to a multi-turn dialog of a user with the system [6].

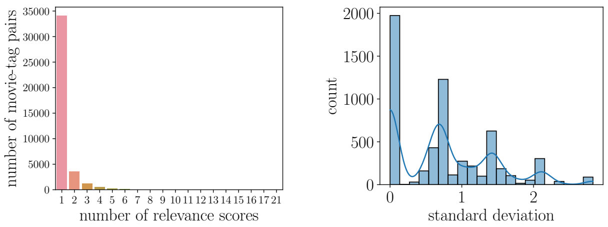
Critiquing recommender systems are a subcategory of conversational recommender systems. They allow users to navigate in the item space with critiques [7, 15, 17]. For example, users can specify if they are interested in movies that have more drama or less comedy than a particular movie [17]. These systems require rich item descriptions that contain relevance scores indicating the strength, with which a concept applies to an item. For example, a relevance score could indicate how violent the movie "The Godfather" is on the continuous scale from 0 to 1.

Tagging systems collect data, which describe items to a certain degree. However, these systems usually only allow users to attach tags, but not specify the strength, with which a tag applies to an item. Collecting relevance scores on item-tag pairs is a demanding process, as it requires users to explicitly indicate the degree to which a tag applies to an item. To the best of our knowledge, there has been only one work, where the authors collected such user feedback and proposed a tag relevance prediction method [17]. We received the method implementation and dataset used in that work to further investigate the tag relevance prediction problem.

In this paper, we define the tag relevance prediction problem, explore the inconsistency of relevance scores provided by users as a challenge of this problem and present a method, which outperforms the state-of-the-art method for predicting tag relevance scores in the movie domain.

We found a moderate inconsistency of user relevance scores, when they assigned scores to the same item-tag pairs. This inconsistency can be caused by different reasons. For example, users might use different scales for relevance scores or might forget a certain scene from a movie. However, in this paper, we focus on the subjectivity of tags. In particular, we found that users tend to disagree more on subjective tags, such as "good acting", "bad plot" or "quotable" than on objective ones, such as "animation", "cars" or "wedding". However, the disagreement of users regarding objective tags still remains moderate indicating presence of other reasons for the disagreement. This paper has the following contributions:

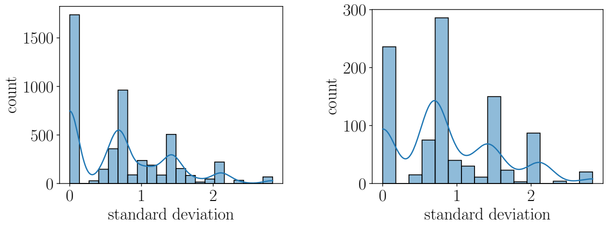
- We define the tag relevance prediction problem and describe its difference from tag recommendation
- We explore one of the challenges of this problem, which is inconsistency of tag relevance scores assigned by users
- We introduce the novel deep learning method, TagDL, which outperforms the state-of-the-art method
- We publish[1] data and code used in our experiments to allow reproducibility and inspire future efforts on this topic



(a) Number of movie-tag pairs per number of relevance scores

(b) Standard deviations of user relevance scores per movie-tag pair

Figure 1: Characteristics of survey data



(a) Objective movie-tag pairs

(b) Subjective movie-tag pairs

Figure 2: Standard deviations of subjective and objective movie-tag pairs

2 RELATED WORKS

To the best of our knowledge, tag relevance prediction problem has only been addressed in [17] and [4]. In [17], the authors described the problem and presented the method based on multilevel nonlinear regression model, which predicts relevance scores for movie-tag pairs based on relevance scores, movie ratings, reviews and tag applications. The resulting data structure was called Tag Genome.

In [4], the authors presented an unsupervised method based on semantic relationships between tags and movie reviews. The authors also conducted a user study, where participants assessed the quality of their method compared to baselines. In the study, the methods generated tags that described differences between two movies. According to the results of this study, Tag Genome outperformed the proposed method.

In this paper, we present a tag relevance prediction method based on a multi-layer perceptron. This idea is not new and has been employed to improve accuracy in the tag recommendation problem [8, 9, 11]. However, we applied this idea to the tag relevance prediction problem, where it had not be applied before, and demonstrated that this method outperforms the state-of-the-art method. One of the reasons for the lack of methods designed to improve the prediction of tag relevance might be the difficulty to reproduce the experiments, such as the absence of publicly available datasets and methods. We therefore will publish method implementations and data used in our experiments.

3 PROBLEM FORMULATION

In [1], tag recommendation task is defined as follows: “Object-centered tag recommendation. Given a set of input tags I_o associated with the target object o, generate a list of candidates C_o ($C_o \cap I_o = \emptyset$), sorted according to their relevance to object o, and recommend the k candidates in the top positions of C_o.” In this definition, candidates correspond to tags and objects to items.

We define object-centered tag relevance prediction and personalized tag relevance prediction similarly:

- Object-centered tag relevance prediction. Given data associated with tag t and item i, predict relevance of tag t for item i.
- Personalized tag relevance prediction. Given data associated with tag t, user u and item i, predict relevance of tag t for item i and user u.

In this paper, we only focus on object-centered tag relevance prediction or tag relevance prediction. We do not indicate what data exactly should be associated with the tag or the item. These data could include features, such as user ratings, fragments of texts or tag appliances.

Tag relevance prediction problem is different from tag recommendation, as tag relevance prediction is the score prediction problem, while tag recommendation is a ranking problem [1]. Although, solving the ranking problem requires predicting scores, these problems require optimizing for different measures. The ranking problem requires optimizing for the order of tags according to their relevance for a particular item, while the prediction problem focuses on the prediction of scores given by users.

Another difference, which is not necessary, but common in these two problems is the input data. Tag recommendation methods usually work with tag scores that (1) indicate the number of times a particular tag has been applied to an item and (2) miss information on tags that do not apply to an item [1]. Meanwhile, tag relevance prediction methods usually operate on relevance scores, which indicate the strength, with which a particular tag applies to an item [17].

4 INCONSISTENCY OF RELEVANCE SCORES

The authors of [17] conducted a survey, where they asked users to indicate the degree, to which a tag applies to a movie on a 5-point Likert scale from “not at all” to “very strongly”. The users could also indicate that they are not sure regarding this strength. The authors asked each user to indicate relevance of tags to a set of movies picked from those the user watched in the past. The authors collected 58,903 responses, 7,740 of which were marked as “not sure” leaving 51,163 responses for analysis. We received the dataset from this study and further explored it in this paper.

[1] https://github.com/Bionic1251/Revisiting-the-Tag-Relevance-Prediction-Problem

Figure 1a demonstrates the power law distribution of relevance scores assigned to movie-tag pairs. Most movie-tag pairs have only one relevance score. Figure 1b demonstrates the distribution of standard deviations for movie-tag pairs with more than one relevance score. Among 40,013 movie-tag pairs, only 5,942 pairs have more than one relevance score. The mean of standard deviations for these movie-tag pairs is 0.74 ± 0.023 (99% CI), which can be considered as moderate deviation (less than one point of the 5-point Likert scale). We use standard deviation as a measure of user disagreement.

We hypothesize that users have higher disagreement on subjective tags than on objective ones. In this paper, the term *subjective tag* refers to a tag, which is solely based on personal beliefs or feelings, while the term *objective tag* refers to a tag, which is not based on personal beliefs or feelings and its presence or absence in a movie can be considered as a fact. Subjective tags include the following examples: "bad acting", "so bad it's good", "beautiful scenery", "awesome soundtrack" and "dumb but funny". For objective tags, the examples are as follows: "chicago", "courtroom", "dogs", "harry potter" and "oscar'. We also included movie genres to the category of objective tags, as genres have relatively strict definitions. We suppose that users disagree more on subjective tags than on objective ones, because subjective tags involve user opinions, which can vary among users, while objective tags represent facts, which should be the same for different users in most cases.

To test our hypothesis, we asked two judges to label whether a particular tag is subjective. According to [10], the judges had a fair agreement (unweighted Cohen's kappa: 0.4, p-value $< 10^{-10}$, 99% CI [0.34, 0.46]). The judges resolved their disagreements, which resulted in 5,089 objective and 853 subjective movie-tag pairs (821 objective and 263 subjective tags). According to Figures 2a and 2b, standard deviation of subjective movie-tag pairs is more skewed left. The average standard deviation for objective movie-tag pairs is lower (0.71 ± 0.025, 99% CI) than that of subjective movie-tag pairs (0.87 ± 0.058, 99% CI) (t-test, p-value $< 10^{-9}$). Due to the moderate inconsistency of relevance ratings, the minimum possible mean absolute error (MAE) for object-centered tag relevance prediction is 0.19 for the whole dataset.

We found a statistically significant difference between standard deviations of objective and subjective movie-tag pairs. However, this difference can be caused by different factors, such as users forgetting certain parts of movies, using different scales or misunderstanding tags. Further research is needed to investigate contributions of different factors. We also considered only extremely subjective tags as subjective and others as objective, but tag subjectivity can be a continuous value. A different categorization might result in different findings.

5 METHOD

Our method (TagDL) is a multilayer perceptron (MLP) implemented in PyTorch framework [12]. Figure 3 depicts its architecture. As input data MLP receives a vector of concatenated features of an item-tag pair. The concatenated features include eight features used in the state-of-the-art method (Section 7) and a one-hot vector, which describes the tag (categorical variable). Unlike [17], we group relevance scores given to the same item-tag pairs and calculate their average in our input data.

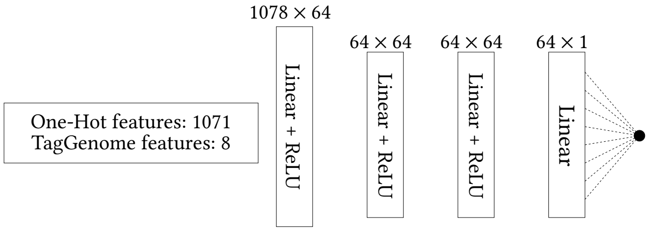


Figure 3: Multilayer perceptron architecture (TagDL)

While performing training step by tuning MLP parameters we discovered that the most important factors were the learning rate and the number of training epochs.

6 DATASET

Our dataset contains information on 5,192 movies. Each movie is associated with data from four sources:

(1) Tag applications correspond to tags that users attach to movies in MovieLens along with the number of these applications. Overall, our dataset contains 1,071 tags (actors' and directors' names have been removed from the set), which were applied 829,236 times.
(2) User reviews have been collected from the IMDB website[2], which resulted in 2,624,608 reviews.
(3) Ratings correspond to the numbers of stars users assign to movies in MovieLens. We used 95,379,210 ratings from the range [0.5, 5] stars with the granularity of 0.5.
(4) User survey was conducted by the authors of tag genome [17], where they asked users to indicate the score, with which a particular tag applies to a movie on a 5-point Likert scale. Overall, the dataset contains 51,163 relevance scores excluding scores where users indicated that they were not sure about the score.

7 BASELINE

We compare the performance of our method with the state-of-the-art method proposed in [17]. The method is based on a multilevel nonlinear regression model, which takes features extracted from the dataset as input and predicts relevance scores assigned to movie-tag pairs. The method uses the following features:

(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 of tag t and item i based on latent semantic indexing [3], where each document d_i is a set of tags applied to item i
(3) *text-freq*(t, i) - number of times tag t appears in text reviews of item i. The feature is calculated in stem-/no-stem- variants using raw reviews and after applying word stemming, i.e. reducing words to their root or base form using Porter Stemmer [13], as implemented in [18]
(4) *text-lsi-sim*(t, i) - similarity of tag t and item i using latent semantic indexing [3], where each document d_i is the set of words in user reviews of item i

[2] http://imdb.com/

(5) *avg-rating*(t, i) - average rating of item i
(6) *rating-sim*(t, i) - cosine similarity of ratings of item i and aggregated ratings of items with tag t
(7) *regress-tag*(t, i) - predicted relevance score based on a regression model using *tag-applied*(t, i) as the output variable and the other features (along with the number of tag applications) as the input variables

8 EXPERIMENTAL SETUP

To compare the performance of our method with that of existing ones, we performed a 10-fold cross-validation procedure and used mean absolute error (MAE) as a performance measure. To reproduce experimental conditions presented in [17], we received the code and the dataset from authors of the article. We compared the following methods:

- **Average** - average relevance score in the training dataset
- **Vig et al.** - the multilevel nonlinear regression model presented in [17] (Section 7)
- **TagDL** - our multi-layer perceptron (Section 5), which receives the same eight features (Section 7) as the Vig et al. method, concatenates these features with the one hot vector based on a tag (1071 features) and treats this vector as input data. We tuned our method based on the validation dataset, which resulted in the following hyperparameters: activation function: rectified linear unit, learning rate: 10^{-6}, epoch number: 30, input layer: 1079 neurons, two hidden layers: 64 neurons each and output layer: 1 neuron (see Figure 3)

9 RESULTS

Table 1 demonstrates performace of evaluated methods. To test if differences between mean values of obtained during 10-fold-cross-validation MAE values are statistically significant, we performed t-tests with the null hypothesis that the mean of the differences is equal to 0. We conducted three pairwise comparisons, which confirmed statistical significance of our results (three t-tests with max p-value $< 10^{-6}$). In this paper, we conducted five statistical tests and therefore corrected our p-values according to the Bonferroni correction.

Method	MAE
Average	1.451 ± 0.010
Vig et al. [17]	0.833 ± 0.011
TagDL	0.811 ± 0.012

Table 1: Average MAE values for 10-fold cross validation results for the evaluated methods with 99% confidence intervals.

According to Table 1, TagDL outperforms the Vig et al. method by 2.6% (MAE difference with 99% confidence interval: 0.022±0.005), which indicates that our method provides more precise predictions than the state-of-the-art method. The Vig et al. method outperforms the average baseline, as it was reported in [17]. The results of Vig et al. in our experiment match the results presented in the article [17], except we do not scale the prediction to the range [0, 1] (our MAE is multiplied by 4).

Our prediction is in 99% CI [0.799, 0.823], which is around one point of the 5-point Likert scale. However, the minimum theoretically possible MAE for object-centered prediction is 0.041, which indicates that there is a room for improvement. The minimum MAE is lower for the test set than for the whole dataset (Section 4), as the test set contains fewer relevance scores for the same movie-tag pair.

10 CONCLUSION AND FUTURE WORK

In this paper, we defined the tag relevance prediction problem and presented an approach based on deep learning, which outperforms the state-of-the-art algorithms at predicting tag relevance for movies. Our results suggest that deep learning has a great potential at solving this task. We found that users tend to disagree on relevance scores regarding objective item-tag pairs more than subjective ones, but assign relevance scores with moderate inconsistency to both categories.

We plan to explore the tag prediction problem further. In particular, by using language models like transformers [16], embeddings [2] to model textual data and other algorithms, such as neural collaborative filtering [5] which might improve the result. We are also interested in investigating ways to receive more consistent relevance scores from users and factors that affect inconsistency of relevance scores.

This paper outlines an initial attempt to improve tag relevance prediction. With this work, we wish to inspire others to work on this problem and therefore share data and the code of methods presented, which allow to reproduce described experiments.

ACKNOWLEDGMENTS

We would like to thank GroupLens for providing us with the dataset and programming code of the baseline algorithm. We would also like to thank organizations that supported this research: the Academy of Finland, grant #309495 (the LibDat project) and the Academy of Finland Flagship programme: Finnish Center for Artificial Intelligence FCAI.

REFERENCES

[1] Fabiano M Belém, Jussara M Almeida, and Marcos A Gonçalves. 2017. A survey on tag recommendation methods. *Journal of the Association for Information Science and Technology* 68, 4 (2017), 830–844.
[2] Kenneth Ward Church. 2017. Word2Vec. *Natural Language Engineering* 23, 1 (2017), 155–162.
[3] 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.
[4] 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.
[5] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. 2017. Neural collaborative filtering. In *Proceedings of the 26th international conference on world wide web*. 173–182.
[6] Dietmar Jannach, Ahtsham Manzoor, Wanling Cai, and Li Chen. 2020. A Survey on Conversational Recommender Systems. *arXiv preprint arXiv:2004.00646* (2020).
[7] Denis Kotkov, Qian Zhao, Kati Launis, and Mats Neovius. 2020. ClusterExplorer: Enable User Control over Related Recommendations via Collaborative Filtering and Clustering. In *Proceedings of the 2020 ACM conference on Recommender systems*.
[8] Kai Lei, Qiuai Fu, Min Yang, and Yuzhi Liang. 2020. Tag recommendation by text classification with attention-based capsule network. *Neurocomputing* 391 (2020), 65–73.
[9] Suman Kalyan Maity, Abhishek Panigrahi, Sayan Ghosh, Arundhati Banerjee, Pawan Goyal, and Animesh Mukherjee. 2019. DeepTagRec: A content-cum-user

based tag recommendation framework for stack overflow. In *European Conference on Information Retrieval*. Springer, 125–131.

[10] Mary L McHugh. 2012. Interrater reliability: the kappa statistic. *Biochemia medica* 22, 3 (2012), 276–282.

[11] Hanh TH Nguyen, Martin Wistuba, Josif Grabocka, Lucas Rego Drumond, and Lars Schmidt-Thieme. 2017. Personalized deep learning for tag recommendation. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*. Springer, 186–197.

[12] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *Advances in Neural Information Processing Systems 32*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (Eds.). Curran Associates, Inc., 8024–8035. http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf

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

[14] Paul Resnick and Hal R. Varian. 1997. Recommender Systems. *Commun. ACM* 40, 3 (March 1997), 56–58. https://doi.org/10.1145/245108.245121

[15] Taavi T Taijala, Martijn C Willemsen, and Joseph A Konstan. 2018. Movieexplorer: building an interactive exploration tool from ratings and latent taste spaces. In *Proceedings of the 33rd Annual ACM Symposium on Applied Computing*. ACM, 1383–1392.

[16] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *arXiv preprint arXiv:1706.03762* (2017).

[17] 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

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