Explaining Content-based Recommendations with Topic Models

Gustavo Padilha Polleti
Computação e Sistemas Digitais
Escola Politécnica da Universidade de São Paulo
São Paulo, Brasil
gustavo.polleti@usp.br

Fabio Gagliardi Cozman
Escola Politécnica da Universidade de São Paulo
São Paulo, Brasil
fgcozman@usp.br

***Abstract*—Recommendation systems play a key role in current online commerce enterprises. Despite their success, they usually behave like black-boxes from the user perspective, typically failing to produce high quality human-computer interactions; interpretability is thus a major concern for the next generation of recommendation systems. In this paper we propose a model-agnostic method based on topic models that generates explanations for content-based recommendation systems.**

***Index Terms*—Recommendation Systems, Explanations, Topic Models, Latent Feature Models**

I. Introduction

Explanation generation has been a topic of great interest in recommendation systems [1] [2] [3]. Indeed, the performance of recommendation systems relies not only on their providing good suggestions, but also on how much their recommendations contribute to the user making decisions — something that depends on the user interaction with the system. Previous studies suggest that users are not just looking for blind recommendations from a system, but are also looking for a justification of the system's choice [4].

Popular techniques employed in recommendation systems, like matrix factorization (MF) [5] and phrase embeddings [6], can detect hidden patterns through latent feature models, and have led to significant advances in performance. However, as they convert rich semantic information into real-valued vector-spaces, they are quite hard to interpret. Models like Latent Dirichlet Allocation (LDA) [7] can instead lead to decisions that are relatively easy to interpret; as such, they have recently been used to explain latent features in collaborative filtering [8]. This paper extends such approaches into a novel explanation generation method, which is further analysed and evaluated in the context of content-based recommendation systems (instead of collaborative filtering ones). We demonstrate that LDA indeed offers valuable explanations for content-based recommendations.

The paper is organized as follows. Section II discusses related work and presents some notation and terminology. We then propose in Section III an explanation generation method. We describe empirical results in Section IV, and offer some concluding remarks in Section V.

II. Background

In this section we look first at recommendation systems, then at topic models, and finally at interpretability.

A. Recommendation Systems

Recommendation systems provide suggestions for items so as to support user decision-making [5]. User interests are usually expressed as a historic profile of actions or ratings. Despite their success [9], recommendation systems often face difficulties with cold starts and changes in user interests. Recent efforts have explored adaptive behavior, reinforcement learning and dialogue systems [10], [11].

Recommendation systems can be classified into six groups: Content-based, Collaborative filtering (CF), Demographic, Knowledge-based, Community-based and Hybrid [5]. All of them aim at identifying similarities, but their targets are different. While content-based systems are based on item description, collaborative filtering, on the other hand, considers similarities between users interests, so that recommends items regardless if they are similar or not to the user profile. Knowledge-based recommendations relies on expert beliefs about how items meet users needs and preferences. Community-based and demographic systems are based on features of the user herself.

As content-based systems do not depend exclusively on user profiles, they can avoid difficulties like cold-starts; besides, they can be associated with collaborative filtering techniques [12] [13]. One way to build content-based systems is to rely on latent feature models that map semantically rich features into numerical vectors. Such embeddings are expected to map similar items to nearby vectors; thus one can select items that are similar to any given item. For instance, in Figure 1 we have items e_h and e_t mapped to a two-dimensional space where the distance $d(e_h, e_t)$ indicates how "similar" they are.

B. Topic Models

A topic model is usually built so as to discover the main themes that pervade a large collection of unstructured documents [14]. The first and still most used technique is a three-level Bayesian network called Latent Dirichlet Allocation (LDA), in which each item of a collection is modeled as a distribution over topics and, in the context of text, each topic

2643-6264/19/$31.00 ©2019 IEEE
DOI 10.1109/BRACIS.2019.00143

appears as a distribution of words [7]. Figure 2 depicts the graph behind LDA, with D documents, each one of N words, and K topics. First, for each document, Θ_d are sampled. Then, for each word in this document, a topic $Z_{d,n}$ is chosen; finally, a word is chosen from a multinomial probability parameterized by $Z_{d,n}$. In reality, only the collection of words that constitutes the documents are observable, so the LDA aims at inferring this hidden structure from the textual data.

The distribution Θ_d for a document implies that the topic β_k represents the proportion $\theta_{d,k} \in \Theta_d$ of the total content. As a result of LDA, each document has its content structured as topics that can be interpreted as document features.

The most appropriate value for the number of topics is an hyper-parameter of LDA, and tuning it affects overall interpretability. The most commonly used metric for this purpose is known as *coherence* [15], which measures how much statements support each other. On the other hand, to evaluate the models convergence when learning an LDA, one important metric is *perplexity* [7].

C. Interpretability

Rarely one follows a recommendation that cannot be backed by some meaningful words. It is however not easy to establish when a technique displays high "interpretability" [16], [17].

Intuitively, one should expect that more complicated models can capture more subtle patterns and hence lead to higher accuracy; however as the model becomes more flexible, its interpretability suffers. A recent central concern in in machine learning has been to reduce the gap between interpretability and performance. [16] [18]. Note that interpretability is not the same as *transparency*; the latter is related to the kinds of relationships an algorithm can extract [19]. A recommendation system may fail to be interpretable and yet it may be transparent in that it tells the user the features it relies upon and the data it collects.

Many techniques now adopt a model-agnostic approach; that is, they are not limited to a specif class of models. Techniques vary in scope: some focus on explaining the whole model of interest (a *global* or *holistic* approach) while others focus on a particular prediction, or a particular set of predictions. Often the interpretation of a single prediction is based on the construction of a simpler model "around" the observed features so as to explain how the observations locally led to the prediction.

The evaluation of techniques with respect to interpretability is still a challenge; objective measures are lacking, and the most effective methods rely on human inspection, a time consuming and expensive process.

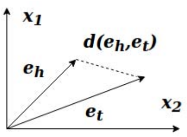

Fig. 1. Euclidean distance as similarity measure between items.

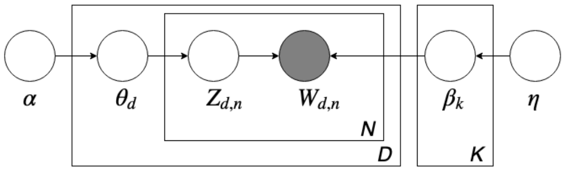

Fig. 2. LDA Graphical model representation.

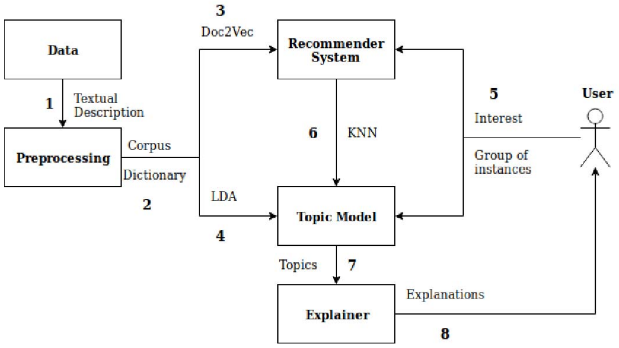

Fig. 3. Diagram of recommendation system with topic model explainer.

III. Topic Model Explanations

Topic models let one organize, search and filter documents by identifying previously hidden topics. Due to their intrinsic interpretability, topic models have recently been proposed as an alternative to explanation generation in collaborative filtering [8]. The main assumption behind that work is that characteristics of a textual description can explain at least partially the preferences of an user. That is, the user preference is related to the topics an item is about, and those can be structured by topic models.

The same intuition can be applied to content-based recommendation systems. That is, a topic model can capture the meaning behind the opaque latent features used to provide recommendations. In fact, *as the system is expected to recommend based on content similarity, the explanation should follow the same working principle.*

The goal of this paper is to propose a model-agnostic method that generates local explanations for a single or group of content-based recommendations. Therefore, it should not depend on the algorithm or machine learning model used to make recommendations: the recommendation system is viewed as a black-box from the perspective of the explainer. As the explanations generated by this method are geared towards an end user, comprehensibility and simplicity are essential.

Figure 3 summarizes the proposal we present in this paper. First, during the training phase, unstructured data, like textual documents, are collected (step 1) and preprocessed (step 2).

The preprocessed data are then fed both to the algorithm building the recommendation system and to the LDA training algorithm (steps 3 and 4). By learning from the same source, both models represent the same content; it is however to be expected that the black box recommendation system will capture subtle and complex patterns, while the LDA will be more interpretable but perhaps less accurate in suggesting good recommendations.

Example 1: To illustrate these differences, consider a toy example regarding the movies *Guardians of the Galaxy vol.2* (G.G.) and *The Arrival*. The LDA might represent their respective distributions over topics (Θ_d) as the following numerical representation:

$$\Theta_i = \begin{bmatrix} x_{alien} \\ x_{hero} \end{bmatrix} : \Theta_{G.G.} = \begin{bmatrix} 0.6 \\ 0.4 \end{bmatrix}, \Theta_{Arrival} = \begin{bmatrix} 0.9 \\ 0.1 \end{bmatrix} \qquad (1)$$

In this example, consider the topics *Alien*, *Super-Hero*. So one could take that these two movies are about aliens, as both of them have a high probability for this topic ($\theta_{d,k}$). That is, we can identify the topics that entities have in common. Suppose additionally that the recommendation system adopts a complex model that can capture more hidden patterns in the data than the LDA. Even though the latent features used by the recommendation system lead to valuable recommendations based on similarity, they may not be meaningful to a human subject. □

Returning to our framework, once we have a content-based recommendation and a topic model learned as indicated before, the user provides an input, a historic profile D_u of previous liked items (step 5). That is, D_u is a list of known items that as a whole describes the user's interests.

In step 6 the *K* items (D_r) whose content are most similar to those provided by the user's input (D_u) are recommended. Recommended items D_r are suggested considering similarities among them and the user profile (D_u), but no explicit characteristics are deduced. On the other hand, the LDA encodes the same entities in terms of a distribution over topics. By applying the topic model, it is possible to identify the distribution of topics (Θ) for items ($D = D_u \bigcup D_r$) from both sets (step 7). The next step is to discover which topics these items have in common and, thus, which are their content similarities.

Example 2: Back to our toy example: The topic model turned the movies *Guardians of the Galaxy vol.2* and *The Arrival* into the distribution of the topics in Expression (1). One can expect that a recommendation system would take them to be similar to each other because of the topic *Alien*. □

The LDA thus provides a representation with interpretable features ($\theta_{d,k} \in \Theta_d$) that can be used to connect D_u and D_r in an interpretable manner. We now explain how this is done.

Firstly, let T_u be the set of topics that have probability greater than 0 for a document u in the user profile. The same for T_r and a document r in the recommendation set. That is,

$$(\forall t_k \in T_u)\, \theta_{u,k} \neq 0, u \in D_u, \theta_{u,k} \in \Theta_u,$$

$$(\forall t_k \in T_r)\, \theta_{r,k} \neq 0, r \in D_r, \theta_{r,k} \in \Theta_r.$$

Secondly, as the intersection $T_u \cap T_r$ contains the topics shared by both $u \in D_u$ and $r \in D_r$, in order to discover the topics that are shared by at least one item in user profile and recommendations, it is calculated this intersection for each document as presented in Equation (2). Note that even if a topic is shared by members from the same set, if it doesn't have a match on the other one, then it is not present in T_p:

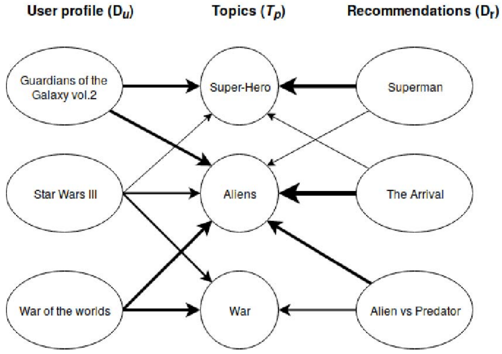

Fig. 4. Toy example for movie recommendations of the topic similarities model.

$$T_p = \bigcap_{u}^{D_u} \bigcap_{r}^{D_r} T_u \cap T_r. \qquad (2)$$

As the topics ($T_p^{\complement}$) do not contribute to establish links between these two sets, they are, therefore, discarded. The filtered topics are supposed to contain only the topics (T_p) that indeed represent similarities between the user profile and recommendation sets. For example, if an user profile item is about a topic that does not plays a part in any of the recommendations, it is removed from our analysis.

Consider the special case where the recommended set and user profile does not share any topic in common ($T_p = \varnothing$). In this case, as the recommendation cannot be explained, and is not expected to properly fit the user's interests, it should be removed from the recommendation set.

In order to identify their content similarities in a interpretable way, it is necessary to match each item in recommendation set to the user profile ones using the topics (T_p) to establish connections. These links are then modelled as a bipartite graph data model (G), in which the user profile and recommendations are disjoint sets and each item represents a node. The topics bridges nodes from each set that share a common subject. It should be noted that the connections are weighted by the topic probability distribution value. We thus have

$$G = (N, E), \qquad N = D_u \cup D_r,$$

$$E = \{\theta_{d,k} | d \in D \wedge t_k \in T_p\}.$$

Example 3: A toy example is depicted in Figure 4: arrows indicate that an item belongs to a certain topic (the stronger the relation the thicker the arrow), so the movie "*Superman*" is more related "*Super-hero*" than to "*Aliens*". □

To identify the main reasons behind a recommendation, it is necessary to find the most influential topics; thus a *relevance score* must be attributed to each topic. The scoring formula proposed in this paper is based on the intuition that a good explanation should emphasize those topics that describe the current recommendation using the items the user liked [20]. Given a topic $t_k \in T_p$, its score is calculated as:

$$score(k, D_u, D_r) = (\gamma_1 \frac{n_{k,D_u}}{|D_u|} + \gamma_2 \frac{n_{k,D_r}}{|D_r|}) * IDF(k), \quad (3)$$

$$n_{k,D_u} = \sum_{i}^{D_u} \theta_{i,k}, \quad n_{k,D_r} = \sum_{j}^{D_r} \theta_{j,k}. \quad (4)$$

Expression (3) was adapted from Ref. [20]. While n_{k,D_u} is the number of edges, weighted by its respective influence, that connects the topic t_k with the items in the user profile, n_{k,D_u} is similar but regarding items in the recommendation set. These terms can be interpreted as the summation of the topic probabilities for each document, as presented in Expression (4). The terms γ_1 and γ_2 are weighting factors, for example $\gamma_1 \gg \gamma_2$ favors topics that are more influential in user profile's items than in the recommendations. The $IDF(k)$ modifies Inverse Document Frequency (IDF) [21] to represent the proportion of documents about topic t_k. Due to IDF, the pure frequency of a topic is weighted with the inverse of its popularity, thus more interesting and less common explanations patterns are likely to emerge [20].

The resulting scores are sorted in descending order and the top-k are the topics that better represent the content similarity between the user profile (D_u) and the recommendations (D_r). It is thus possible to establish reasoning like "*The item D_r was recommended because it is about topic t_k.*", thus eliciting the content similarities that support the recommendations.

IV. Experiments

To validate our proposal, an explainable content-based recommendation system for scientific articles was implemented with our proposed method. We collected a dataset from publications databases Scopus[1] from Elsevier and Web of Science[2] (WoS) from Thomson Reuters. The title and abstract of 8014 articles from 1971 to 2019 about conversational agents were collected. We implemented the whole system in Python, with implementations of doc2vec, LDA and coherence models available at gensim [3]. The plots and tables presented in this paper were made matplotlib [4] and pandas [5].

The implemented recommendation system used the descriptive information in abstracts to compare and identify similarities between articles. To do so, we used doc2vec [6] to build a phrase embedding where each article is represented by a dense vector that captures semantic relationships. Similar articles should be mapped to nearby vectors. The recommendation system resorts to a k-nearest neighbors model that recommends the k closest documents in the embedding space. For this work, we investigated values of 10, 300 and 100 for k, for vector dimension and for number of epochs.

Figure 5 shows visualization of the 10 nearest (blue) and the 10 furthest documents (green) for the sample article titled "*Empirical evaluation of a reinforcement learning spoken dialogue system*" (red) in a two dimensional space. To build this figure, PCA reduced the embedding dimension from 200 to 50 and then the t-SNE algorithm produced coordinates for each document. One can see that clusters separate well the documents, suggesting that the embedding is appropriate.

The LDA used in the explainer module had parameters, such as number of passes and number of topics selected according to traditional metrics like perplexity, log likelihood [7] and coherence [15]. This process is detailed in the next section. To find out after how many passes the topic model converges, three traditional LDA metrics were employed: log likelihood, perplexity and topic difference [7]. Perplexity should decrease monotonically to some point where generalization performance is attained. Log likelihood instead increases. The topic difference represents how much the topics change after each iteration, when the changes are insignificant the model has converged. Figure 6 depicts these metrics during the training of a topic model with 20 topics and 100 passes. By pass 78, the values of topic diff remain the same up to the 3 decimal places.

We also had to determine the number of topics that provide best interpretability. The best objective metrics for this purpose are those related to model coherence [15]. Figure 7 shows the values of three coherence metrics (Cuci, Umass and Cv) for each LDA model varying its number of topics. Despite the noise, it is possible to observe that Cuci and Cv reach a maximum between 20 and 30 topics, while Umass exhibits a slight plateau in the same interval. After a manual inspection the models in this interval, the one with 29 topics appeared to be the most interpretable, thus it was selected to be used for posterior analysis.

Table I contains the 5 topics with the highest coherence values (cv). These topics seems interpretable and labels can be assigned to them.

Finally, the topic model and the content-based recommendation system were combined to evaluate the explanation

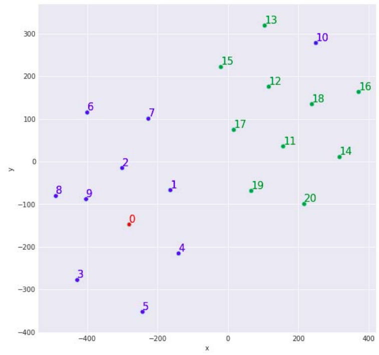

Fig. 5. t-SNE visualization of the nearest (blue) and furthest (green) for sample document (red).

[1] https://www.scopus.com/home.uri
[2] http://www.webofknowledge.com/
[3] https://radimrehurek.com/gensim/
[4] https://matplotlib.org/
[5] https://pandas.pydata.org/

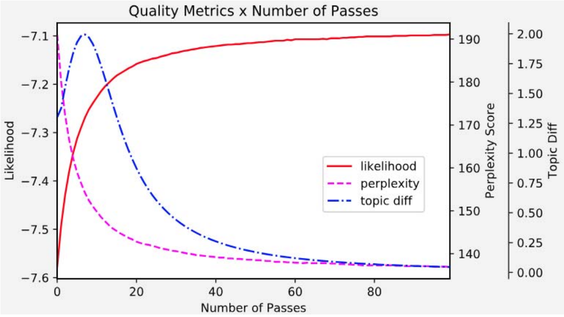

Fig. 6. Convergence curve.

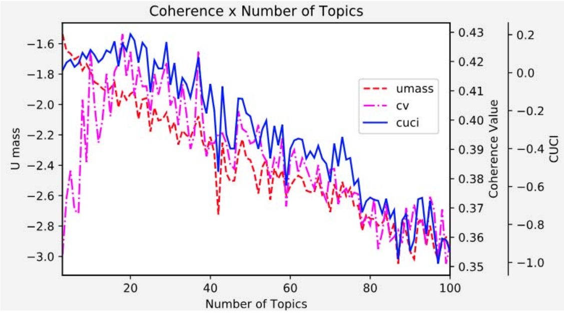

Fig. 7. Coherence curve.

method, so as to answer:

1) The topic model can successfully correlate the items in user profile with those in recommendations ?
2) The relevance score is able to rank the most significant topics that justifies the recommendations ?
3) Both a single and a group of recommendations can be explained using this method ?

As these questions rely on a subjective and qualitative analysis, the entire system was evaluated via manual inspection of recommendations and their respective explanations. To illustrate the analysis, Table IV presents the top 5 most relevant topics for a sample test case, using $\gamma_1 = \gamma_2 = 1$ to calculate the relevance score. Furthermore, the results also contain the most significant topics for each document. Those topics that are also taken to be the most relevant ones are highlighted.

Table IV shows that almost all (except one) recommendations share at least one topic in common with the user profile item, what conducts to the conclusion that the topic model indeed identified content similarities that connects both sets, answering the first question. Also, after reading all recommended articles, was noticed that the topic *Healthcare* is the one that better links the recommendation set to the user profile, what answers the second question.

As shown by Table IV, the most relevant characteristics to be considered for this set are "*health*", "*conversational agents*" and "*knowledge*", in this order. Therefore, even though the embedding latent features are not interpretable, the recommendations can be explained by pointing to which topics both

TABLE I
TOP 5 MOST COHERENT TOPICS

Topics				
education	**discussion**	**embodied**	**health**	**knowledge**
8	**26**	**9**	**10**	**4**
learn	argument	gestur	health	semant
student	base	model	patient	knowledg
learner	contain_paper	behavior	intervent	gener
tutor	proceed_contain	speech	support	question
educ	topic_discuss	gener	medic	inform
develop	discuss_includ	verbal	studi	structur
environ	design	anim	base	sentenc
Topic Coherence (CV)				
0.559466	0.550118	0.531331	0.528111	0.522872

TABLE II
RECOMMENDATIONS FOR SAMPLE DOCUMENT

Articles	
Point	**Title**
Most Similar	
1	optimizing dialogue management with reinforcement learning
2	demonstration of the parlance system
3	personalizing a dialogue system with transfer reinforcement learning
4	designing and evaluating an adaptive spoken dialogue system
5	evaluation of a hierarchical reinforcement learning spoken
Most Dissimilar	
11	a multi-modal dialog system for a mobile robot
12	"sorry, i cannot understand": ways of dealing with non-undefined
13	rmrsbot - using linguistic information to enrich a chatbot
14	an interactive concierge for independent living
15	7th mexican international conference on artificial intelligence

the recommendation and user profile items have in common. In addition, a single recommendation can also be explained by its own topics, for instance, the third recommended item share with the user profile item the topics "*health*" and "*conversational interface*", what answers the third question. Overall, the method can identify latent features (topics) that are easy to interpret, thus leading to successful local explanations that are model agnostic as they do not make assumptions about the recommendation system.

V. CONCLUSION

This paper has proposed a method for explanation generation in content-based recommendation systems, in addition describing an implementation and successful tests. Topic models emerge as powerful tools for explaining complex latent feature models, notably for document embeddings.

This work is a first step in the development of interpretable content-based recommendations. The next step is to capture user feedback with human subjects so as to examine subjetive properties of explanations in recommendations.

ACKNOWLEDGEMENTS

This work was carried out with the support of Itaú Unibanco S.A.; the first author has been supported by the Itaú Scholarship Program (PBI), linked to the Data Science Center (C2D) of the Escola Politécnica da Universidade de São Paulo. The second author has been partially supported by the Conselho Nacional de Desenvolvimento Cientifico e Tecnologico (CNPq), grant 312180/2018-7. The work was also

TABLE III
TOP 5 MOST RELEVANT TOPICS

Topics				
health	**conversational agent**	**knowledge**	**conversational interface**	**speech**
10	**24**	**4**	**5**	**14**
health	agent	semant	user	word
patient	human	knowledg	interfac	utter
intervent	interact	gener	interact	featur
support	research	question	devic	method
medic	convers_agent	inform	environ	recognit
studi	commun	structur	voic	detect
base	social	sentenc	servic	base
Relevance Score				
0.544651	0.288909	0.174326	0.084051	0.066822

Title	**Topics**
User Profile Items	
ontology-based dialogue systems for improved patient hpv vaccine ...	**10** 0 **24** **4** **5** **14**
Recommendation Items	
generating a chatbot with teenager personality for preventing ...	20 7 19 26 **4** 12
acceptability in interaction from robots to embodied conversational agents	**24** 6 9 23 25
assessing the posture prototype: a late-breaking report on patient views	**10** 12 13 **5** 19
acceptability in interaction: from robots to embodied conversational agents	**24** 6 13
how do you want your chatbot? an exploratory wizard-of-oz study ...	7 26 23
emotion assessment for affective computing based on physiological ...	25 23 **5** 28 **10**
using dialogues to access semantic knowledge in a web ir system	**4** 16 21
agent-user concordance and satisfaction with a virtual hospital ...	**10** **24** 13
a simple connectionist approach to language understanding ...	18 15 **4** **14** 21
verbal indicators of psychological distress in interactive ...	23 **10** **14** 20 13 24

supported by the Fundação de Amparo a Pesquisa do Estado de São Paulo (FAPESP), grant 2016/18841-0, and also by the Coordenação de Aperfeiçoamento de Pessoal de Nivel Superior (CAPES) - finance code 001. We are also grateful to the Center for Inovation at Universidade de São Paulo (InovaUSP) for hosting our lab.

REFERENCES

[1] N. Tintarev and J. Masthoff, "A survey of explanations in recommender systems," in *Proceedings of the 2007 IEEE 23rd International Conference on Data Engineering Workshop*, ser. ICDEW '07. Washington, DC, USA: IEEE Computer Society, 2007, pp. 801–810. [Online]. Available: http://dx.doi.org/10.1109/ICDEW.2007.4401070

[2] N. Tintarev, "Explanations of recommendations," in *Proceedings of the 2007 ACM Conference on Recommender Systems*, ser. RecSys '07. New York, NY, USA: ACM, 2007, pp. 203–206. [Online]. Available: http://doi.acm.org/10.1145/1297231.1297275

[3] G. Friedrich and M. Zanker, "A taxonomy for generating explanations in recommender systems," *AI Magazine*, vol. 32, pp. 90–98, 2011.

[4] R. Sinha and K. Swearingen, "The role of transparency in recommender systems," in *CHI '02 Extended Abstracts on Human Factors in Computing Systems*, ser. CHI EA '02. New York, NY, USA: ACM, 2002, pp. 830–831. [Online]. Available: http://doi.acm.org/10.1145/506443.506619

[5] Y. Koren and R. M. Bell, "Advances in collaborative filtering." in *Recommender Systems Handbook*, F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor, Eds. Springer, 2011, pp. 145–186. [Online]. Available: http://dblp.uni-trier.de/db/reference/rsh/rsh2011.htmlKorenB11

[6] Q. Le and T. Mikolov, "Distributed representations of sentences and documents," in *Proceedings of the 31st International Conference on International Conference on Machine Learning - Volume 32*, ser. ICML'14. JMLR.org, 2014, pp. II–1188–II–1196. [Online]. Available: http://dl.acm.org/citation.cfm?id=3044805.3045025

[7] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet allocation," *J. Mach. Learn. Res.*, vol. 3, pp. 993–1022, Mar. 2003. [Online]. Available: http://dl.acm.org/citation.cfm?id=944919.944937

[8] M. Rossetti, F. Stella, and M. Zanker, "Towards explaining latent factors with topic models in collaborative recommender systems," in *Proceedings of the 2013 24th International Workshop on Database and Expert Systems Applications*, ser. DEXA '13. Washington, DC, USA: IEEE Computer Society, 2013, pp. 162–167. [Online]. Available: http://dx.doi.org/10.1109/DEXA.2013.26

[9] B. Smith and G. Linden, "Two decades of recommender systems at amazon.com," *IEEE Internet Computing*, vol. 21, no. 3, pp. 12–18, May 2017. [Online]. Available: https://doi.org/10.1109/MIC.2017.72

[10] K. Christakopoulou, F. Radlinski, and K. Hofmann, "Towards conversational recommender systems," in *Proceedings of the 22Nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '16. New York, NY, USA: ACM, 2016, pp. 815–824. [Online]. Available: http://doi.acm.org/10.1145/2939672.2939746

[11] E. Liebman and P. Stone, "Dj-mc: A reinforcement-learning agent for music playlist recommendation," *Proceedings of the 13th International Conference on Autonomous Agents and Multiagent Systems (AA-MAS 2014)*, vol. abs/1401.1880, 2014. [Online]. Available: http://arxiv.org/abs/1401.1880

[12] M. Volkovs, G. W. Yu, and T. Poutanen, "Content-based neighbor models for cold start in recommender systems," in *Proceedings of the Recommender Systems Challenge 2017*, ser. RecSys Challenge '17. New York, NY, USA: ACM, 2017, pp. 7:1–7:6. [Online]. Available: http://doi.acm.org/10.1145/3124791.3124792

[13] Z. Lu, Z. Dou, J. Lian, X. Xie, and Q. Yang, "Content-based collaborative filtering for news topic recommendation," in *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, ser. AAAI'15. AAAI Press, 2015, pp. 217–223. [Online]. Available: http://dl.acm.org/citation.cfm?id=2887007.2887038

[14] D. M. Blei, "Probabilistic topic models," *Commun. ACM*, vol. 55, no. 4, pp. 77–84, Apr. 2012. [Online]. Available: http://doi.acm.org/10.1145/2133806.2133826

[15] M. Röder, A. Both, and A. Hinneburg, "Exploring the space of topic coherence measures," in *Proceedings of the Eighth ACM International Conference on Web Search and Data Mining*, ser. WSDM '15. New York, NY, USA: ACM, 2015, pp. 399–408. [Online]. Available: http://doi.acm.org/10.1145/2684822.2685324

[16] F. Doshi-Velez and B. Kim, "Towards a rigorous science of interpretable machine learning," *arXiv*, 2017. [Online]. Available: https://arxiv.org/abs/1702.08608

[17] T. Miller, "Explanation in artificial intelligence: Insights from the social sciences," *CoRR*, vol. abs/1706.07269, 2017. [Online]. Available: http://arxiv.org/abs/1706.07269

[18] D. Gunning, "Darpa's explainable artificial intelligence (xai) program," in *Proceedings of the 24th International Conference on Intelligent User Interfaces*, ser. IUI '19. New York, NY, USA: ACM, 2019, pp. ii–ii. [Online]. Available: http://doi.acm.org/10.1145/3301275.3308446

[19] C. Molnar, *Interpretable Machine Learning*, 2019, https://christophm.github.io/interpretable-ml-book/.

[20] C. Musto, F. Narducci, P. Lops, M. de Gemmis, and G. Semeraro, "Linked open data-based explanations for transparent recommender systems," *International Journal of Human-Computer Studies*, vol. 121, pp. 93 – 107, 2019, advances in Computer-Human Interaction for Recommender Systems. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1071581918300946

[21] C. Manning, P. Raghavan, and H. Schutze, "Scoring, term weighting, and the vector space model," *Introduction to Information Retrieval*, pp. 100–123, 01 2008.