On Using Arabic Language Dialects in Recommendation Systems

Abdulla Alshabanah **Murali Annavaram**
University of Southern California
{aalshaba, annavara}@usc.edu

Abstract

While natural language processing (NLP) techniques have been applied to user reviews in recommendation systems, the potential of leveraging Arabic dialects in this context remains unexplored. Arabic is spoken by over 420 million people, with significant dialectal variation across regions. These dialects, often classified as low-resource languages, present both challenges and opportunities for machine learning applications. This paper represents the first attempt to incorporate Arabic dialects as a signal in recommendation systems. We explore both explicit and implicit approaches for integrating Arabic dialect information from user reviews, demonstrating its impact on improving recommendation performance. Our findings highlight the potential for leveraging dialectal diversity in Arabic to enhance recommendation systems and encourage further research at the intersection of NLP and recommendation systems within the Arab multicultural world.

1 Introduction

Recommendation systems have become integral to various online platforms, enhancing user experience by providing personalized recommendations (Naumov et al., 2019; Covington et al., 2016; Ying et al., 2018) and consequently driving the economic activity of influential technology companies (Linden et al., 2003; Greenstein-Messica and Rokach, 2018; Gomez-Uribe and Hunt, 2016). These systems typically rely on user behaviour to understand preferences and improve recommendations. One source for understanding user preferences is the content generated by users for the product reviews they make public. As user-generated reviews contain valuable information, they have been used by many recommendation systems to improve the recommendation performance (Almahairi et al., 2015; Chen et al., 2015; Wang et al., 2021b; Sachdeva and McAuley, 2020). While previous works applied natural language processing (NLP) techniques to analyze text reviews in recommendation systems (Qiu et al., 2022; Catherine and Cohen, 2017), the potential of leveraging Arabic language dialects from such reviews remains unexplored.

Arabic, spoken by over 420 million people worldwide, comprises numerous dialects that vary in grammar and vocabulary. Although they share the same root in Modern Standard Arabic (MSA), these dialects diverge across the cultures of the Arab world, leading to differences in spoken practice and social media content. These dialects present unique challenges and opportunities for natural language processing tasks due to their extensive linguistic diversity and lack of standardization. The divergence of dialects often lead to mutual unintelligibility, different from the difference between English dialects, such as British and American English, which exhibit much higher mutual comprehensibility. Specific challenges include (1) semantic divergence, where words may look or sound similar across dialects but carry entirely different meanings. (2) diglossia, where MSA serves as the formal, high-resource language while numerous other Arabic dialects are underrepresented (Ferguson, 1959). (3) Lack of standard orthography, as dialectal Arabic is written informally with inconsistent spelling and frequent code-switching (Bergman and Diab, 2022). Current research in Arabic NLP has focused on Arabic dialect identification (Keleg and Magdy, 2023; Keleg et al., 2024, 2023), but their application in recommendation systems has yet to be explored.

In this paper, we explore the use of Arabic dialects as a signal for recommendation models and investigate its effect on recommendation performance. We first examine how Arabic dialects in user reviews can serve as an explicit signal to the recommendation model. In particular, we first propose to identify the dialect and incorporate that information as a user feature fed to a recommen-

Findings of the Association for Computational Linguistics: NAACL 2025, pages 2178–2186

dation model. We then demonstrate a successful approach to including Arabic dialect information as an implicit signal in the recommendation model. By improving the user behavior modeling with the aid of Arabic dialects in text reviews, we aim to draw the attention of the NLP and recommendation system communities to the opportunities within the Arab multicultural world.

2 Preliminaries

2.1 Arabic Dialects in NLP

Arabic Dialect Identification (ADI) is the task of automatically detecting specific Arabic dialects in written texts (Althobaiti, 2020), and recent shared tasks have focused on developing advanced computational methods to improve the ADI accuracy (Abdul-Mageed et al., 2021, 2022, 2023, 2024). ADI can be framed as a single label classification problem, where an Arabic text is assigned a label belonging to one of the Arabic dialect (Bouamor et al., 2019). However, A key limitation of framing ADI task as a single-label classification problem, is that assigning each sentence to only one dialect overlooks the fact that a sentence may be valid in multiple dialects (Abdul-Mageed et al., 2023). Hence, (Keleg and Magdy, 2023) analyzed the problems of the ADI single-label framing and framed ADI as a multi-label classification problem.

While ADI aims to classify Arabic text into dialect labels, the Arabic Level of Dialectness (ALDi) (Keleg et al., 2023) takes a more nuanced approach by quantifying the degree of dialectal content in a sentence on a continuous scale. ALDi recognizes that Arabic speakers often mix MSA and dialectal Arabic, resulting in a spectrum of dialectness rather than a discrete MSA/dialect distinction.

In our experiments, we employ both ADI and ALDI techniques to enhance user behavior modeling in recommendation systems, using a signal derived from the dialect of Arabic text in user reviews.

2.2 Recommendation Models

Recommendation models play a crucial role in personalizing user experiences by predicting which items a user is most likely to engage with. These models can be viewed through the lens of user-item interactions, which are often represented as a bipartite graph. In this graph, users and items are nodes, and interactions between them, such as clicks or purchases, form the edges. In an explicit feedback setting, the graph may also include additional information, like ratings, to weight these edges. In contrast, in an implicit feedback setting, interactions are treated with uniform edge weights. We consider an implicit feedback setting where user-item interactions are treated as uniformly weighted positive edges and associated with user reviews, while missing interactions (items a user did not interact with) are considered negative edges.

A variety of recommendation models have been developed to leverage these interaction graphs. One such model is the Two-Tower Neural Network (TTNN), which is widely used in large-scale recommendation systems (Balasubramanian et al., 2024; Yi et al., 2019; Naumov et al., 2019). TTNNs consist of two Deep Neural Networks (DNNs), one for users and one for items. The user network processes both dense and sparse features to generate a dense representation of the user, $\mathbf{x_u}$, while the item network does the same for items, producing a dense representation $\mathbf{y_i}$. The relevance of item i to user u is then calculated using a scoring function $F_s : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$, which could be as simple as a dot product or as complex as another DNN.

More formally, the user network processes user feature representations, which are concatenated and passed through a multi-layer perceptron (MLP) to produce a user representation:

$$\mathbf{x_u} = \text{MLP}_u(\text{feature representations}_u) \quad (1)$$

Similarly, the item network processes item feature representations and generates item representations:

$$\mathbf{y_i} = \text{MLP}_i(\text{feature representations}_i) \quad (2)$$

The relevance between a user u and an item i is calculated using a scoring function:

$$F_s(\mathbf{x_u}, \mathbf{y_i}) = \text{MLP}_s(\mathbf{x_u} \oplus \mathbf{y_i}) \quad (3)$$

Where $\oplus$ denotes the concatenation operator.

3 Arabic Dialects as Model Signals

In this section, we first present the design choices for incorporating Arabic dialect information as an explicit signal, followed by its use as an implicit signal in the recommendation model. Here, a signal refers to input data that impacts the model's learning. An explicit signal means that the Arabic dialect information is treated as a user feature where it is is explicitly passed to the model. In contrast, an implicit signal embeds the influence of Arabic dialects within the user representation without passing it explicitly to the model.

3.1 Dialect as an Explicit Signal

To introduce Arabic dialect as an explicit signal, we employed ADI and ALDi estimation models to extract dialect information from user reviews and create a user dialect feature. Specifically, we used BERT-based models fine-tuned for single-label and multi-label ADI (Keleg and Magdy, 2023), as well as a BERT-based model fine-tuned to estimate the ALDi scores (Keleg et al., 2023). The created user dialect feature $\mathbf{df_u}$ is then fed as a user feature into the user network in the TTNN to generate the user representation $\mathbf{x_u}$. Given the dialect label, either a single or multi label, and the ALDi score for each user review, we explored the following options for creating the user dialect feature $\mathbf{df_u}$:

• *Single-Label*: A single dialect label is used as the user dialect feature $\mathbf{df_u}$ for the positive user-item interaction associated with the user review. For negative interactions, which do not have an associated review, $\mathbf{df_u}$ will be a single dialect label sampled from the user's previously used dialects.

• *Multi-Label*: Multi dialect labels are used as the user dialect feature $\mathbf{df_u}$ for the positive user-item interaction associated with the user review. For negative interactions, the two most common dialects previously used by a user are chosen for $\mathbf{df_u}$.

• *Multi-Label++*: Considering the multi dialect labels for the reviews of all positive user-item interactions, the two most common dialects used by a user are chosen as the user dialect feature $\mathbf{df_u}$ for both positive and negative interactions.

For all three options, if a user review has an ALDi score less than 0.1, the review text will be regarded as Modern Standard Arabic and labeled accordingly.

3.2 Dialect as an Implicit Signal

To introduce the Arabic dialect as an implicit signal, we employed an Arabic Level of Dialectness estimation model (Keleg et al., 2023) to obtain ALDi scores from user reviews and use them in augmenting the user representation $\mathbf{x_u}$ in the TTNN's user network. This is achieved by sampling items the user interacted with and using the representations of the sampled items, generated from the item network, to enhance $\mathbf{x_u}$. Specifically, we let the sampling probability of item i for user u follow the ALDi scores $ALDi_{u,i}$ from the reviews written by user u for item i.

Sampling with a probability following the Arabic Level of Dialectness scores serves as an implicit signal to the model. The intuition is to increase the visibility of items reviewed by a user in dialectal Arabic. This approach is informed by sociolinguistic studies suggesting that linguistic choices, such as code-switching to dialects, express one's identity and serve as identity markers (Johnstone and Bean, 1997; Bassiouney, 2006). Therefore, items reviewed in dialectal Arabic are expected to be more informative about a user, and it has been shown in (Balasubramanian et al., 2024) that increasing the visibility of items that are more informative about a user is likely to aid in modeling user behavior. We explore two options for integrating the representations of the sampled items into the model:

• *Implicit$_{uniform}$*: We take an elemise-wise uniform mean of the sampled items representations and concatenate the resulting vector with the user representation $\mathbf{x_u}$ in the user network to get an augmented user representation $\mathbf{x_u^{AUG}}$.

• *Implicit$_{weighted}$*: As sampled items could have varying amounts of information, we let the model attend to the sampled item representations differently by using scaled-dot product attention (Vaswani et al., 2017). Then, we use the attention scores to take a weighted mean of the sampled item representations and concatenate it with the user representation $\mathbf{x_u}$ in the user network to obtain an augmented user representation $\mathbf{x_u^{AUG}}$.

The scoring function F_s in the TTNN will then use $\mathbf{x_u^{AUG}}$, instead of $\mathbf{x_u}$, along with the item represention $\mathbf{y_i}$ and to compute the relevance of item i to user u.

More formally, to incorporate implicit signals from the Arabic Level of Dialectness (ALDi) scores, we enhance the user representation $\mathbf{x_u}$ by sampling items from the user's interaction set $\mathcal{I}_u$. The ALDi score $ALDi_{u,i}$ quantifies the level of dialectal Arabic in the review written by user u for item i. Sampling is performed with a probability proportional to the ALDi score:

$$P(i \mid u) = \frac{ALDi_{u,i}}{\sum_{j \in \mathcal{I}_u} ALDi_{u,j}} \quad (4)$$

This ensures that items with stronger dialectal signals are prioritized. The sampled items $\mathcal{S}_u \subseteq \mathcal{I}_u$ are processed to create an additional vector $\mathbf{z_u}$, which augments the user representation $\mathbf{x_u}$.

For *Implicit*$_{uniform}$, the contribution of sampled items is calculated as:

$$\mathbf{z_u} = \frac{1}{|\mathcal{S}_u|} \sum_{i \in \mathcal{S}_u} \mathbf{y_i} \quad (5)$$

While for *Implicit*$_{weighted}$, we let the user attends to the sampled items differently, and computes a weighted sum based on attention scores, and the created vector is:

$$\mathbf{z_u} = \sum_{i \in \mathcal{S}_u} \alpha_i \mathbf{y_i} \quad (6)$$

Where α_i represents the attention score of item i. The augmented user representation becomes:

$$\mathbf{x_u^{AUG}} = \mathbf{x_u} \oplus \mathbf{z_u} \quad (7)$$

For both *Implicit*$_{uniform}$ and *Implicit*$_{weighted}$ this augmented representation is used in the final relevance scoring function:

$$F_s(\mathbf{x_u^{AUG}}, \mathbf{y_i}) = \mathrm{MLP}_s(\mathbf{x_u^{AUG}} \oplus \mathbf{y_i}) \quad (8)$$

4 Experimental Setting

Based on the approaches described above we incorporate the dialect information into the recommendation model and evaluate their benefits using the following experimental setup.

Datasets. We conduct the experiments on two public benchmark datasets: Book Reviews in Arabic Dataset (BRAD) (Elnagar et al., 2018; Elnagar and Einea, 2016) and Large-scale Arabic Book Reviews Dataset (LABR) (Aly and Atiya, 2013). Because we're interested in the top k recommendation task, we use a common pre-processing method to convert these datasets into implicit feedback. As in (He et al., 2017; Zhang et al., 2020; Wang et al., 2021a; Sun et al., 2021; Cheng et al., 2021), we divide all the datasets into training, validation and test datasets according to the leave-one-out setting. Detailed statistics about the datasets used in our experiments can be found in Appendix A.

Evaluation Criteria. To assess the performance of the model, we utilize two key metrics: Hit Rate @ k (HR@k) and Normalized Discounted Cumulative Gain @ k (NDCG@k). The Hit Rate evaluates whether a candidate item appears within the top k recommendations list, whereas NDCG measures the position of the recommended item relative to the top of the list.

Implementation Details. The implementation details can be found in Appendix B.

5 Results

We present the top-K recommendation results for all design options in Table 1. The table includes HR@10 and NDCG@10 for both the BRAD and LABR datasets, with the percentage difference relative to the base recommendation model without dialect information indicated between parentheses. In addition to the design options that incorporate the Arabic dialect information, we include a design variant, termed '*Ar-Review*', where we process the Arabic review associated with the positive user-item interaction through a sentence transformer (Reimers and Gurevych, 2019) finetuned specifically for Arabic (Nacar, 2024). The processed review in this design option is passed as a user feature, with the hope that it will enhance the model's understanding of user preferences. We notice the following from the results:

- Including the dialect information, either as explicit or implicit signal, improves the recommendation performance for both datasets. This is evident from the improvement of Multi-Label++, Implicit$_{\text{uniform}}$ and Implicit$_{\text{weighted}}$ over the base recommendation model with no dialect information (No-Dialect). Specifically, there is a significant improvement in both HR@10 and NDCG@10 when dialect information is used as an implicit signal to the model. In the next section, we will present an ablation study to further investigate this observation.

- Using dialect information directly as a user feature in the model provides an improvement in recommendation performance, but only when extracting multiple dialect labels from the review and selecting the two most frequent dialects for both positive and negative interactions (Multi-Label++). This result suggests that the model can benefit from including dialect information as an explicit signal.

- It is worth noting that using different sets of dialects for positive and negative edges, as in the 'Single-Label' and 'Multi-Label' design options,

Model	BRAD HR@10	BRAD NDCG@10	LABR HR@10	LABR NDCG@10
No-Dialect	0.05033	0.02432	0.06754	0.03411
Ar-Review	0.02055 (-59%)	0.01160 (-52%)	0.03659 (-46%)	0.01571 (-54%)
Single-Label	0.01834 (-64%)	0.00802 (-67%)	0.03846 (-43%)	0.01578 (-54%)
Multi-Label	0.01495 (-70%)	0.00614 (-75%)	0.05722 (-15%)	0.02695 (-21%)
Multi-Label++	0.05202 (+3%)	0.02554 (+5%)	0.07129 (+6%)	0.03586 (+5%)
Implicit$_{\text{uniform}}$	**0.06411 (+27%)**	**0.03171 (+30%)**	0.07880 (+17%)	**0.03729 (+9%)**
Implicit$_{\text{weighted}}$	0.06099 (+21%)	0.03040 (+25%)	**0.08068 (+19%)**	0.03664 (+7%)

Table 1: Comparison between design variations by HR@10 and NDCG@10 for BRAD and LABR.

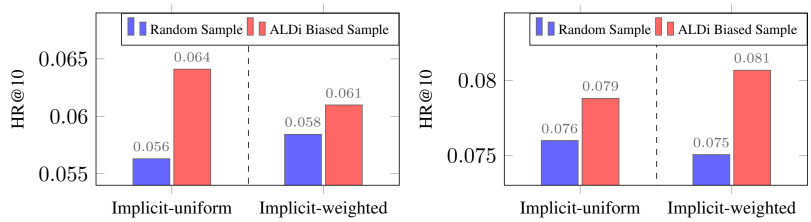

Figure 1: Ablation on ALDi score importance during sampling showing HR for BRAD (left) and LABR (right).

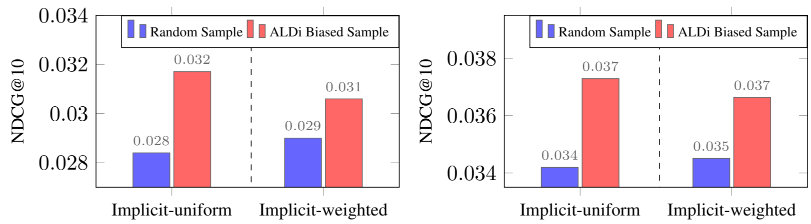

Figure 2: Ablation on ALDi score importance during sampling showing NDCG for BRAD (left) and LABR (right).

degrades model performance. This result may occur because the model might struggle to learn a unified user representation due to inconsistencies in the feature space between positive and negative interactions. This could explain why reviews are rarely used in an implicit feedback setting, which is partly attributed to the lack of reviews associated with negative interactions (Sachdeva and McAuley, 2020; Wan and McAuley, 2018). The same observation and reasoning apply to using an Arabic sentence transformer to process user reviews and incorporate them into the recommendation model, as demonstrated by the inferior results of 'Ar-Review'.

Ablation on ALDi Biased Sampling. Augmenting the user representation with sampled item representations, where each item's sampling probability follows the ALDi score of a user's written review, results in a significant improvement in recommendation performance. In this ablation study, we aim to investigate the specific effect of using ALDi scores as the sampling probability. Figure 1 and Figure 2 show the HR@10 and NDCG@10 results of our ablation experiments, respectively. It is clear that using ALDi scores as the sampling probability leads to better recommendation performance for both Implicit$_{\text{uniform}}$ and Implicit$_{\text{weighted}}$ compared to random sampling, demonstrating that the model benefits from the implicit dialect signal introduced during the sampling process.

6 Conclusion

We presented how to use Arabic dialect information extracted from a user's publicly visible review as valuable signals in recommendation systems. By demonstrating both explicit and implicit design options for incorporating dialect information into the model, we have shown that leveraging dialects from user reviews can improve modeling the user behavior. Our findings aim to encourage further research in the intersection of natural language processing and recommendation systems, particularly within the Arab multicultural world.

7 Limitations

While this paper explores the use of Arabic dialect information to enhance recommendation system performance, there are opportunities for further improvement. Specifically, the extent to which the ALDi score influences sampling probability in the implicit signal experiments could be controlled and investigated. Additionally, while the study focused on recommendation system performance, future work could take an orthogonal approach by assessing the ALDi estimation model's performance if finetuned jointly along with the recommendation model.

8 Ethical Considerations

This paper focused on incorporating Arabic dialects when recommending items to users. We used publicly available datasets containing text reviews written in Dialectal Arabic; however, we aim to draw the research community's attention to the need for more datasets that include a wider range of micro-dialects. Expanding these resources would better serve the Arab multicultural community while minimizing the risk of unintentional bias.

In addition, one might expect that users who use Modern Standard Arabic in the majority of their reviews may not benefit from incorporating dialect information into the model through implicit signal design choices, especially since we sample items with a probability based on the ALDi scores. This could unintentionally create bias against those users and, consequently, degrade the model's performance for them. To investigate this, we report in Figure 3 the percentage improvement of HR@10 for the implicit signal design choices, $\text{Implicit}_{\text{uniform}}$ and $\text{Implicit}_{\text{weighted}}$, relative to the base model with no dialect information for the following division of users:

- Group1: Users with an average ALDi score less than 0.1 for their reviews.
- Group2: Users with an average ALDi score greater than 0.1 for their reviews.

From Figure 3, we can see that there is no bias against users who mostly use Modern Standard Arabic in their reviews when including dialect information as an implicit signal, as the model performance improved for both Group 1 and Group 2, and specifically that Group 1, who has an average ALDi score less than 0.1, did not experience any degradation in model performance.

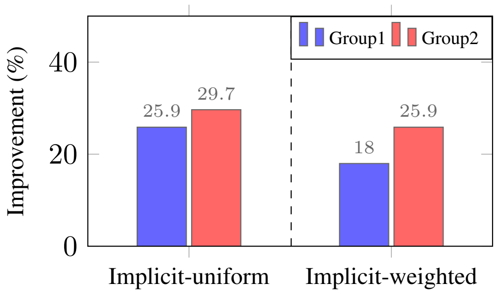

Figure 3: HR@10 percentage improvement of implicit signal design choices relative to the base model with no dialect information for BRAD dataset considering users with varying levels of dialect usage.

Acknowledgment

We sincerely thank all the reviewers for their time and constructive comments. This material is based upon work supported by REAL@USC-Meta center, and a Broadcom gift. The views, opinions, and/or findings expressed are those of the author(s) and should not be interpreted as representing the official views or policies of our sponsors.

References

Muhammad Abdul-Mageed, Abdelrahim Elmadany, Chiyu Zhang, Houda Bouamor, Nizar Habash, et al. 2023. Nadi 2023: The fourth nuanced arabic dialect identification shared task. In *Proceedings of Arabic-NLP 2023*, pages 600–613.

Muhammad Abdul-Mageed, Amr Keleg, Abdelrahim Elmadany, Chiyu Zhang, Injy Hamed, Walid Magdy, Houda Bouamor, and Nizar Habash. 2024. Nadi 2024: The fifth nuanced arabic dialect identification shared task. In *Proceedings of The Second Arabic Natural Language Processing Conference*, pages 709–728.

Muhammad Abdul-Mageed, Chiyu Zhang, Abdelrahim Elmadany, Houda Bouamor, and Nizar Habash. 2021. Nadi 2021: The second nuanced arabic dialect identification shared task. In *Proceedings of the Sixth Arabic Natural Language Processing Workshop*, pages 244–259.

Muhammad Abdul-Mageed, Chiyu Zhang, Abdelrahim Elmadany, Houda Bouamor, and Nizar Habash. 2022. Nadi 2022: The third nuanced arabic dialect identification shared task. In *Proceedings of the Seventh Arabic Natural Language Processing Workshop (WANLP)*, pages 85–97.

Amjad Almahairi, Kyle Kastner, Kyunghyun Cho, and Aaron Courville. 2015. Learning distributed representations from reviews for collaborative filtering. In *Proceedings of the 9th ACM Conference on Recommender Systems*, pages 147–154.

Maha Jarallah Althobaiti. 2020. Automatic arabic dialect identification systems for written texts: A survey. *International Journal of Computational Linguistics (IJCL)*, 11(3):61–89. Revised: 31-10-2020; Published: 01-12-2020.

Mohamed Aly and Amir Atiya. 2013. Labr: A large scale arabic book reviews dataset. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 494–498.

Keshav Balasubramanian, Abdulla Alshabanah, Elan Markowitz, Greg Ver Steeg, and Murali Annavaram. 2024. Biased user history synthesis for personalized long-tail item recommendation. In *Proceedings of the 18th ACM Conference on Recommender Systems*, RecSys '24, page 189–199, New York, NY, USA. Association for Computing Machinery.

Reem Bassiouney. 2006. *Functions of code switching in Egypt: Evidence from monologues*, volume 46. Brill.

A. Bergman and Mona Diab. 2022. Towards responsible natural language annotation for the varieties of Arabic. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 364–371, Dublin, Ireland. Association for Computational Linguistics.

Houda Bouamor, Sabit Hassan, and Nizar Habash. 2019. The MADAR shared task on Arabic fine-grained dialect identification. In *Proceedings of the Fourth Arabic Natural Language Processing Workshop*, pages 199–207, Florence, Italy. Association for Computational Linguistics.

Rose Catherine and William Cohen. 2017. Transnets: Learning to transform for recommendation. In *Proceedings of the eleventh ACM conference on recommender systems*, pages 288–296.

Li Chen, Guanliang Chen, and Feng Wang. 2015. Recommender systems based on user reviews: the state of the art. *User Modeling and User-Adapted Interaction*, 25:99–154.

Mingyue Cheng, Fajie Yuan, Qi Liu, Shenyang Ge, Zhi Li, Runlong Yu, Defu Lian, Senchao Yuan, and Enhong Chen. 2021. Learning recommender systems with implicit feedback via soft target enhancement. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '21, page 575–584, New York, NY, USA. Association for Computing Machinery.

Paul Covington, Jay Adams, and Emre Sargin. 2016. Deep neural networks for youtube recommendations. In *Proceedings of the 10th ACM Conference on Recommender Systems*, New York, NY, USA.

Ashraf Elnagar and Omar Einea. 2016. Brad 1.0: Book reviews in arabic dataset. In *2016 IEEE/ACS 13th International Conference of Computer Systems and Applications (AICCSA)*, pages 1–8. IEEE.

Ashraf Elnagar, Leena Lulu, and Omar Einea. 2018. An annotated huge dataset for standard and colloquial arabic reviews for subjective sentiment analysis. *Procedia computer science*, 142:182–189.

Charles A. Ferguson. 1959. Diglossia. *WORD*, 15(2):325–340.

Carlos A. Gomez-Uribe and Neil Hunt. 2016. The netflix recommender system: Algorithms, business value, and innovation. *ACM Trans. Manage. Inf. Syst.*, 6(4).

Asnat Greenstein-Messica and Lior Rokach. 2018. Personal price aware multi-seller recommender system: Evidence from ebay. *Know.-Based Syst.*, 150(C):14–26.

Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. 2017. Neural collaborative filtering. *CoRR*, abs/1708.05031.

Dan Hendrycks and Kevin Gimpel. 2016. Bridging nonlinearities and stochastic regularizers with gaussian error linear units. *CoRR*, abs/1606.08415.

Barbara Johnstone and Judith Mattson Bean. 1997. Self-expression and linguistic variation. *Language in Society*, 26(2):221–246.

Amr Keleg, Sharon Goldwater, and Walid Magdy. 2023. ALDi: Quantifying the Arabic level of dialectness of text. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 10597–10611, Singapore. Association for Computational Linguistics.

Amr Keleg and Walid Magdy. 2023. Arabic dialect identification under scrutiny: Limitations of single-label classification. In *Proceedings of ArabicNLP 2023*, pages 385–398, Singapore (Hybrid). Association for Computational Linguistics.

Amr Keleg, Walid Magdy, and Sharon Goldwater. 2024. Estimating the level of dialectness predicts interannotator agreement in multi-dialect Arabic datasets. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 766–777, Bangkok, Thailand. Association for Computational Linguistics.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

G. Linden, B. Smith, and J. York. 2003. Amazon.com recommendations: item-to-item collaborative filtering. *IEEE Internet Computing*, 7(1):76–80.

Omer Nacar. 2024. Arabic matryoshka embeddings model - arabic minilm l12 v2 all nli triplet.

Maxim Naumov, Dheevatsa Mudigere, Hao-Jun Michael Shi, Jianyu Huang, Narayanan Sundaraman, Jongsoo Park, Xiaodong Wang, Udit Gupta, Carole-Jean Wu, Alisson G. Azzolini, Dmytro Dzhulgakov, Andrey Mallevich, Ilia Cherniavskii, Yinghai Lu, Raghuraman Krishnamoorthi, Ansha Yu, Volodymyr Kondratenko, Stephanie Pereira, Xianjie Chen, Wenlin Chen, Vijay Rao, Bill Jia, Liang Xiong, and Misha Smelyanskiy. 2019. Deep learning recommendation model for personalization and recommendation systems. *CoRR*, abs/1906.00091.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. 2019. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32.

Gang Qiu, Yanli Guo, and Changjun Song. 2022. Tann: Text-based attention neural network recommendation model. In *2022 IEEE 8th Intl Conference on Big Data Security on Cloud (BigDataSecurity), IEEE Intl Conference on High Performance and Smart Computing,(HPSC) and IEEE Intl Conference on Intelligent Data and Security (IDS)*, pages 119–124. IEEE.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Noveen Sachdeva and Julian McAuley. 2020. How useful are reviews for recommendation? a critical review and potential improvements. In *proceedings of the 43rd international ACM SIGIR conference on research and development in information retrieval*, pages 1845–1848.

Xuehan Sun, Tianyao Shi, Xiaofeng Gao, Yanrong Kang, and Guihai Chen. 2021. *FORM: Follow the Online Regularized Meta-Leader for Cold-Start Recommendation*, page 1177–1186. Association for Computing Machinery, New York, NY, USA.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *CoRR*, abs/1706.03762.

Mengting Wan and Julian McAuley. 2018. Item recommendation on monotonic behavior chains. In *Proceedings of the 12th ACM Conference on Recommender Systems*, RecSys '18, page 86–94, New York, NY, USA. Association for Computing Machinery.

Jianling Wang, Kaize Ding, and James Caverlee. 2021a. Sequential recommendation for cold-start users with meta transitional learning. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '21, page 1783–1787, New York, NY, USA. Association for Computing Machinery.

Xi Wang, Iadh Ounis, and Craig Macdonald. 2021b. Leveraging review properties for effective recommendation. In *Proceedings of the Web Conference 2021*, pages 2209–2219.

Xinyang Yi, Ji Yang, Lichan Hong, Derek Zhiyuan Cheng, Lukasz Heldt, Aditee Ajit Kumthekar, Zhe Zhao, Li Wei, and Ed Chi, editors. 2019. *Sampling-Bias-Corrected Neural Modeling for Large Corpus Item Recommendations*.

Rex Ying, Ruining He, Kaifeng Chen, Pong Eksombatchai, William L. Hamilton, and Jure Leskovec. 2018. Graph convolutional neural networks for web-scale recommender systems. *CoRR*, abs/1806.01973.

Yin Zhang, Derek Zhiyuan Cheng, Tiansheng Yao, Xinyang Yi, Lichan Hong, and Ed H. Chi. 2020. A model of two tales: Dual transfer learning framework for improved long-tail item recommendation. *CoRR*, abs/2010.15982.

A Datasets Details

Table 2 shows the statistics of the datasets used in our experiments.

	BRAD	LABR
Number of Users	33,986	5891
Number of Items	4978	1333
Number of Interactions	455,912	45,681
Sparsity	99.73%	99.42%

Table 2: Statistics of the datasets used in the experiments.

B Implementation Details

Our model is implemented entirely in PyTorch (Paszke et al., 2019) and all experiments are conducted on a server with an Nvidia RTX 5000 GPU and an AMD EPYC 7502 32-Core CPU. We use the GeLU activation function (Hendrycks and Gimpel, 2016), layernorms between layers and an embedding dimension of 96. Specifically, we adopt a common tower structure for the TTNN, wherein higher-layer hidden dimensions contain half the number of neurons compared to their lower layers. The Adaptive Moment Estimation (Adam) (Kingma and Ba, 2015) is chosen as the optimizer. Learning rate and batch size are determined via grid search within the ranges of {0.2, 0.02, 0.002, 0.0002, 0.00002} and {32, 64, 128, 512, 1024}, respectively. Specifically, we use a batch size of 1024 and a learning rate of 0.00002 for the design choices we had in the paper. We use 10 sampled items for the Arabic dialect as an implicit signal experiments. We use a max-margin-based ranking loss function, where the objective is to maximize the score of positive examples and minimize the scores of negative examples, therefore, we sample 20 negative edges for each positive edge during training. We used about 18 GPU hours to train the 1.18M parameter recommendation model, given that training takes 3 minutes on average for each epoch, and we train for a maximum of 100 epochs. We take the mean of three runs for each data point.