Understanding the Relation of User and News Representations in Content-Based Neural News Recommendation

Lucas Möller, Sebastian Padó

Institute for Natural Language Processing, University of Stuttgart, Germany
{lucas.moeller, pado}@ims.uni-stuttgart.de

Abstract

A number of models for neural content-based news recommendation have been proposed. However, there is limited understanding of the relative importances of the three main components of such systems (news encoder, user encoder, and scoring function) and the trade-offs involved. In this paper, we assess the hypothesis that the most widely used means of matching user and candidate news representations is not expressive enough. We allow our system to model more complex relations between the two by assessing more expressive scoring functions. Across a wide range of baseline and established systems this results in consistent improvements of around 6 points in AUC. Our results also indicate a trade-off between the complexity of news encoder and scoring function: A fairly simple baseline model scores well above 68% AUC on the MIND dataset and comes within 2 points of the published state-of-the-art, while requiring a fraction of the computational costs.

1. Introduction

News recommender systems (NRS) guiding users to news items that are of interest to them are in widespread use [1, 2, 3, 4, 5]. Traditional approaches often relied on collaborative filtering and fought with a range of problems [6, 7, 8, 9, 10, 11]. In recent years *neural content-based* approaches have successfully addressed many prior challenges [12, 13, 14, 15]. Figure 1 shows the architecture shared by many of these systems. They typically consist of three components: (a), a *news encoder* which maps individual news articles onto embeddings; (b), a *user encoder* which produces user representations $\mathbf{u}$ as a function of their reading history $\mathbf{h}_t$; (c), a *scoring function* that maps a pair of a candidate news embedding $\mathbf{c}$ and a user representation $\mathbf{u}$ onto a scalar score s.

In this paper, we start from the observation that these three components of the NRS architecture have received very different amounts of attention. Regarding (a), the news encoder, there is a large amount of research, ranging from early applications of convolutional networks [16, 17] to the integration of additional features from topic models [18, 19, 20], or knowledge bases [21, 22, 23] to modern pre-trained language models [24]. As for (b), the user encoder, some systems use

Joint Proceedings of 10th International Workshop on News Recommendation and Analytics (INRA'22) and the Third International Workshop on Investigating Learning During Web Search (IWILDS'22) co-located with the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'22) July 15, 2022, Madrid, Spain

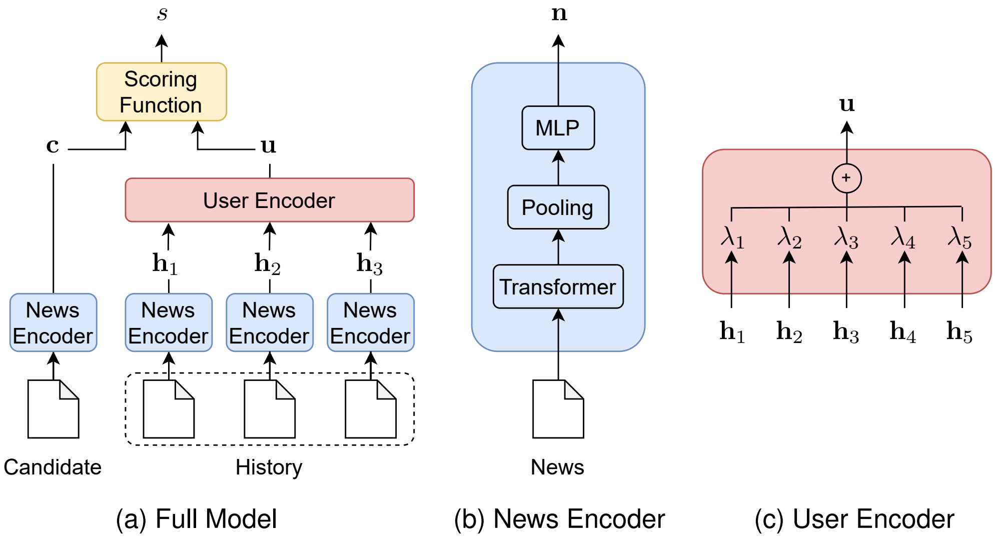

Figure 1: Components of a typical neural NRS. The full model (a) consists of a news encoder (b), a user encoder (b) and a scoring function. Details are given in Section 2.

recurrent models [20, 9, 25] or graph representations [26, 27, 28]. However, state-of-the-art models obtain user representations from additive combinations of the user's reading history [24]. In a large majority of models, the scoring function (c) is then instantiated by an inner product [16, 29, 19, 18, 21, 30, 31, 24, 32]. We believe this is not expressive enough: Given that, user representations are essentially averages of news embeddings, the use of a simple inner product entails that scores can only compare user and candidate news embeddings *within* but not *across* dimensions of the embedding space. This puts a large strain on the news encoder, since it needs to encode any such relevant interactions explicitly in some dimension of the news embedding.

Following up on this observation, we allow our system to model more complex relations between user and candidate news representations by systematically assessing more expressive scoring functions. We evaluate these scoring functions in combination with a number of baseline and SOTA news recommender systems.

Our results show that a more complex, yet relatively simple, scoring function consistently results in a large improvement of the overall performance. Furthermore, it can obviate the need for a complex news encoder and still perform at the state of the art. In this manner, we make a contribution to a better understanding of the roles and importances of the individual components in the general NRS architectures.

2. Method

We implement a neural content-based NRS with the components shown in Figure 1. Our news and user encoder closely follow previously published methods [16, 24]. The focus of our experiments is then on the scoring function, as it is this component that models the relation between candidate news and user representations.

2.1. News Encoder

The content of a news article is typically represented by obtaining embeddings e_i from a sequential model and subsequently pooling them into a fixed-length news vector $\mathbf{n}$. We use a pre-trained transformer [33, 34] for embeddings and an additive attention mechanism from previous studies for pooling [17, 21, 18, 16, 29]:

$$\begin{aligned} \mathbf{n} &= \sum_i \alpha_i \, \mathbf{e}_i \, , \\ \alpha_i &= \text{softmax}\left(\mathbf{q}^T \tanh\left(W \, \mathbf{e}_j + \mathbf{b}\right)\right)_i \end{aligned} \tag{1}$$

The indexes i and j range over all token embeddings. W, $\mathbf{q}$ and $\mathbf{b}$ are parameters. The pooled representation is further processed by two linear layers with ReLU activations. We initialize both the pooling mechanism and the linear layers randomly and train them together with the full model.

2.2. User Encoder

In line with previous studies [21, 16, 18, 22], we compute user embeddings $\mathbf{u}$ from their reading histories. We combine the vector representation $\mathbf{h}_t$ of respective news by means of an additive attention mechanism analogous to the one used in the news encoder:

$$\mathbf{u} = \sum_t \lambda_t \, \mathbf{h}_t \tag{2}$$

Here the index t is over the last T historic news a user has read, and λ_t are computed analogous to α_i in Equation 1.

2.3. Scoring Functions

We now define a series of scoring functions to compute a score s for the match between a user embedding $\mathbf{u}$ and a candidate news embedding $\mathbf{c}$. We focus on generalizations of the inner product with increasing expressiveness.

As discussed above, a simple scoring function is an inner product followed by a sigmoid transformation:

$$s\left(\mathbf{u}, \mathbf{c}\right) = \sigma\left(\mathbf{c}^T \mathbf{u}\right) \tag{3}$$

However, its limitation becomes clear when we plug in Equation 2,

$$\mathbf{c}^T \mathbf{u} = \sum_d c^d u^d = \sum_d \sum_t \lambda_t \, c^d h_t^d \, , \tag{4}$$

where d indexes the dimension of the embedding space: Only dependencies within identical dimensions of history and candidate news are considered for the computation of s. The score cannot depend on dependencies across different dimensions of the feature space.

We can remove this limitation by defining a scoring function based on a bilinear form where off-diagonal entries in the matrix A may capture interactions among different dimensions:

$$s = \sigma\left(\mathbf{c}^T A\,\mathbf{u}\right) \tag{5}$$

When A is treated as a parameter, the scoring function becomes a learnable component. If we add a bias $\mathbf{b}$ and an activation function a we obtain a non-linear version:

$$s = \sigma\left(\mathbf{c}^T a\left(A\,\mathbf{u} + \mathbf{b}\right)\right), \tag{6}$$

Finally, we consider a two-layer MLP acting on the concatenation $\mathbf{u}||\mathbf{c}$ of the two representation vectors:

$$s = \sigma\left(W_2\, a\left(W_1\left[\mathbf{u}||\mathbf{c}\right] + \mathbf{b}\right)\right) \tag{7}$$

3. Experiments and Results

3.1. Experimental Setup

Data. We carry out experiments on the widely used Microsoft News Dataset (MIND, Wu et al. [35]) for news recommendation. It consists of logs generated from one million randomly sampled users over a period of six weeks and contains approximately 160k news items. Notably, this dataset also contains cold start sessions, i.e. sessions for which no user history exists.

Task and Model. We train a binary click-prediction classification task with a standard cross-entropy objective. Each input is a news document presented to a user in a given session and the user's reading history up to this session. The output is whether the news document was clicked. For a fair comparison among model architectures, we use a pre-trained roBERTa transformer [36] to encode the news documents (cf. Section 2.1) throughout. We do not fine-tune the transformer, which permits us to train the full model on a single RTX 2070 GPU.

Training. In each training iteration, we sample one clicked news and a number of K negatives from a given session. In accordance with previous studies, we use $K\!=\!4$ and a batch size of 64. We use the Adam optimizer with a learning rate of $1e\!-\!4$. A maximum of the last $T\!=\!25$ news are used from a user's reading history. The embedding dimensionality for news and user vectors is set to 256. All trainings run for five epochs.

Evaluation. We evaluate our experiments with the standard ranking metrics *Area Under the Curve* (AUC), *Mean Reciprocal Rank* (MRR), and *Normalized Discount Cumulative Gain* up to position five (NDCG@5) and ten (NDCG@10). Cold start users receive random scores from a uniform distribution.
To test whether one model is significantly superior to another, we use del Barrio's test for stochastic dominance on the loss distributions of the respective models. This test is non-parametric and compares the percentile functions of two distributions [37]. For a detailed explanation and the test's suitability for the evaluation of deep models we refer to the work by Dror et al. [38]. We choose a maximum violation level of $\epsilon\!=\!0.33$ and a significance level of $\alpha\!=\!0.01$.

Table 1
Performances of different scoring functions and number of parameters for various models. Original model configurations are indicated by *(orig.)*. Results for the best scoring function in each model and metric are underlined. Figure 3 visualizes these results. Refer to the text regarding significance of improvements.

Model + Scoring Function	**AUC**	**MRR**	**NDCG@5**	**NDCG@10**	**params**
Base + inner	62.59	27.89	29.69	36.55	526k
Base + bilinear	67.50	32.43	35.58	41.95	591k
Base + nonlinear	68.66	32.66	36.06	42.45	657k
Base + mlp	67.99	32.42	35.71	42.09	592k
NPA + inner *(orig.)*	61.67	27.62	29.13	36.16	23.2M
NPA + bilinear	68.23	32.50	35.78	42.13	23.2M
NPA + nonlinear	68.23	32.67	36.02	42.35	23.3M
NPA + mlp	68.08	32.63	35.92	42.27	23.2M
NAML + inner *(orig.)*	62.21	26.67	28.17	35.35	1.06M
NAML + bilinear	67.89	32.49	35.81	42.23	1.13M
NAML + nonlinear	67.90	32.68	35.99	42.42	1.20M
NAML + mlp	67.89	32.02	35.29	41.82	1.13M
NRMS + inner *(orig.)*	68.57	33.02	36.20	42.78	3.15M
NRMS + bilinear	68.40	32.24	35.58	42.17	3.22M
NRMS + nonlinear	68.74	32.53	35.96	42.35	3.28M
NRMS + mlp	68.85	32.85	36.33	42.75	3.22M
NRMS ablation + inner	63.82	28.47	30.57	37.48	2.89M
NRMS ablation + bilinear	68.20	32.28	35.52	42.15	2.95M
Mean + inner	58.89	25.55	27.13	33.62	263k
Mean + bilinear	67.68	32.51	35.85	42.15	328k
Mean + nonlinear	67.81	32.52	35.64	42.02	394k
Mean + mlp	66.88	32.08	35.15	41.40	328k

3.2. Experiment 1: Comparing Scoring Functions

In our first experiment we evaluate the performance of our Base model from Section 2 in combination with all four scoring functions. The results are shown at the top of Table 1 (first group of results).

We find a reasonable baseline performance of around 62.6% AUC for the inner product score. The bilinear scoring function clearly outperforms the inner product by 5 points in AUC (67.5%). The non-linear scoring function further improves the performance by one point to 68.7%, which the MLP cannot surpass (68.0%). The improvement of the bilinear scoring function over the inner one and that of the nonlinear over the bilinear one are both significant ($\epsilon = 0$ and $\epsilon = 0.29$).

Figure 2 (Base) shows the loss distributions of all four models. Clearly, compared with the other models the inner product has a distinctively lower peak at low values and a much heavier tail towards higher values. It also has a sharp peak at $log(0.5) \approx 0.7$, indicating the model is uncertain about a substantial fraction of the data and places them right at the decision boundary.

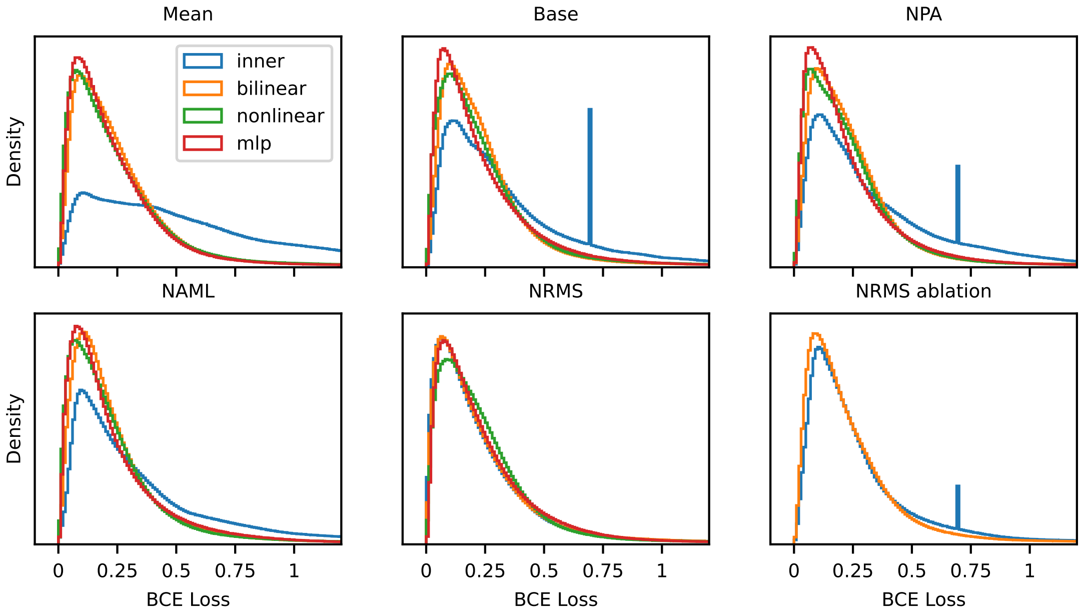

Figure 2: Test loss distributions for all models and scoring functions.

3.3. Experiment 2: SOTA Models with Different Scoring Functions

To investigate to what extent these patterns generalize beyond our Base model, we now vary the scoring function in three state-of-the-art NRS models: NPA personalizes the user encoder [16], NAML includes categorical and textual news features [18] and NRMS applies multi-head attention in the user- and news-encoder [29]. All standardly use an inner product score. We complete the set of models with a trivial Mean baseline which replaces the attention mechanisms in the news and user encoders (Equation 1, Equation 2) with simple averages. The results are shown in the rest of Table 1. Figure 2 visualizes the loss distributions of all combinations.

Strikingly, for all models except NRMS the bilinear scoring function largely outperforms the inner product. Moreover, the bilinear models are strictly stochastically dominant over the inner product models ($\epsilon = 0$), i.e. they outperform the latter at every percentile of the loss distribution [37, 38].

Second, our Base model from Experiment 1 performs within 0.2 percentage points AUC of the best overall model (68.7% vs. 68.9%). Even the Mean model in combination with the nonlinear scoring function comes to within 1 percentage point AUC of the best model (67.8%). This is especially interesting considering the poor performance of the Mean model in combination with an inner product scoring function (58.9%). By changing only the scoring function, this trivial baseline can compete with much more complex architectures.

In contrast, we cannot confirm a superiority of a nonlinear over a bilinear scoring function in this experiment. For the models tested here, the two show very similar results. Improvements, where present, are not significant.

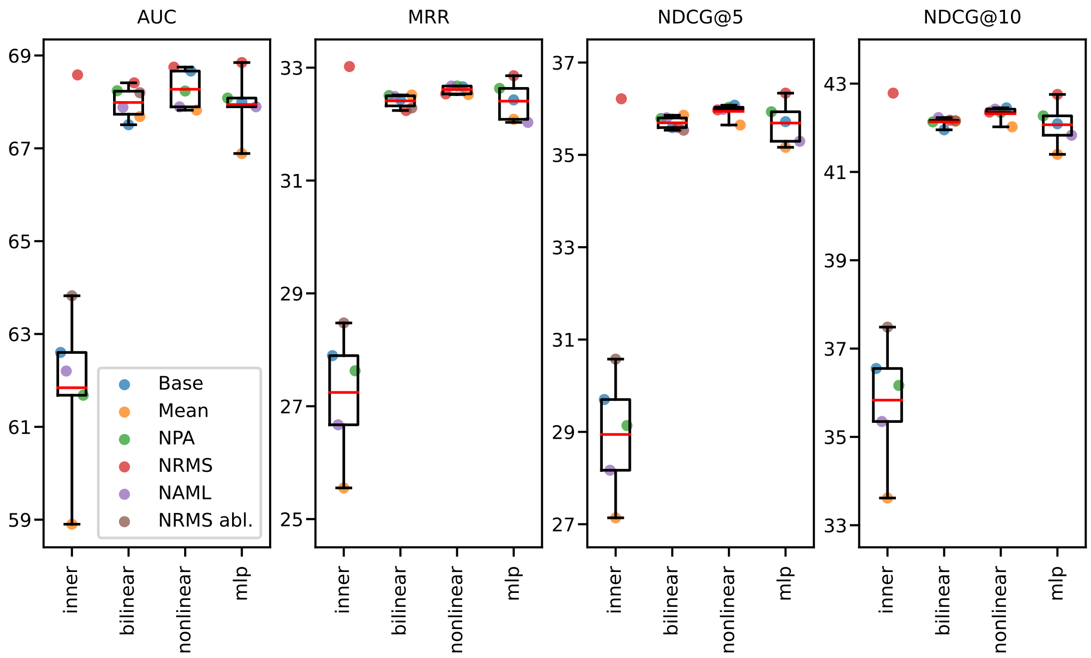

Figure 3: Boxplot visualizations of the four evaluation metrics for all combinations of models (legend) and scoring functions (x-axes). Red lines indicate means. Exact values can be found in Table 1

The outlier model in this experiment is NRMS, for which the choice of scoring function does not appear to matter much. We believe that this is the case because NRMS is the only model that transforms the news representations in a user's history before additively combining them to a user embedding. To test whether this transformation plays an important role, we remove it, obtaining the 'NRMS ablation' model. Indeed, this modification leads to a large drop in performance of almost 5 pp AUC to a level slightly above the NPA, NAML and Base model. When replacing the inner product with a bilinear score the performance recovers to 68.2% AUC. This performance is not significantly worse than that of the original NRMS model ($\epsilon = 0.45$).

3.4. Meta Analysis

We conclude by carrying out a meta analysis of the results across all combinations of scoring functions with the five implemented model architectures (Mean, Base, NPA, NAML, NRMS). Figure 3 visualizes the results from Tabel 1 using boxplots.

The bilinear scoring function accounts for an average improvement of 6.1 ± 1.5 points in AUC over a simple inner product[1]. Very much in parallel, MRR increases by 5.2 ± 1.1 pp, NDCG@5 by 6.7 ± 1.3 and NDCG@10 by 6.3 ± 1.3 points. On the contrary, there is hardly a difference between the bilinear, nonlinear or MLP scoring functions.

An interesting result is also that simple models (Mean and Base) in combination with more

[1] For the NRMS model in combination with an inner product, we consider the ablation described above.

powerful scoring functions perform better than models with complex news encoders (NPA, NAML, NRMS ablation) combined with an inner product score. A more expressive scoring function appears to be able to compensate for complexity in other parts of the model.

Finally, Table 1 also shows the number of parameters in every model. NPA and NRMS are especially parameter hungry due to their use of embedding and attention layers (20M and 3M, respectively). NAML has additional parameters for category embedding layers and a second news encoder for the abstract of the news. The Mean and Base models, on the other hand, only have some 100ks of parameters, meaning that they are cheaper and likely more robust to be learned.

4. Conclusion

In this paper, we have dissected the relation of user and candidate news representation in content-based neural NRS, which is modeled by the scoring function. On top of a range of baseline and SOTA models, we find a large improvement of 6.2 ± 1.4 points in AUC for moving from an inner product to a bilinear form, but no further improvements for moving to a nonlinear version or an MLP. These findings extend similar results on collaborative approaches by Rendle et al. [39] to neural content-based NRS.

By implementing a bilinear scoring function, a trivial baseline (Mean) can almost reach a 1 pp AUC proximity of our best model, while having an order of magnitude less parameters. Our slightly more complex Base model comes within a 2 pp AUC margin of the currently published state of the art [24].

We achieve these results without fine-tuning the transformer backbone of the news encoder. Together with their small number of parameters, these models require relatively little computational costs. Thus, they can serve as conceptually simple and cheap, yet powerful baselines [40].

Overall, we conclude that representing users by means of an additive combination of historic news embeddings and subsequently using an inner product to model the relation with candidate news is not sufficient — A more expressive relation between user and candidate news representations can enhance the performance of NRS by a large margin and can even compensate for complex news encoders.

We believe our study is a first step towards a systematic understanding of the importances of the individual components of NRS for their overall performance.

References

[1] L. Li, W. Chu, J. Langford, R. E. Schapire, A contextual-bandit approach to personalized news article recommendation, in: Proceedings of the 19th International Conference on World Wide Web, WWW '10, Association for Computing Machinery, New York, NY, USA, 2010, p. 661–670. URL: https://doi.org/10.1145/1772690.1772758.

[2] S. Okura, Y. Tagami, S. Ono, A. Tajima, Embedding-based news recommendation for millions of users, in: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '17, Association for Computing Machinery, New York, NY, USA, 2017, p. 1933–1942. URL: https://doi.org/10.1145/3097983.3098108. doi:10.1145/3097983.3098108.

[3] E. Kirshenbaum, G. Forman, M. Dugan, A live comparison of methods for personalized article recommendation at forbes.com, in: P. A. Flach, T. De Bie, N. Cristianini (Eds.), Machine Learning and Knowledge Discovery in Databases, Springer Berlin Heidelberg, Berlin, Heidelberg, 2012, pp. 51–66. doi:10.1007/978-3-642-33486-3_4.

[4] A. Said, J. Lin, A. Bellogín, A. de Vries, A month in the life of a production news recommender system, in: Proceedings of the 2013 Workshop on Living Labs for Information Retrieval Evaluation, LivingLab '13, Association for Computing Machinery, New York, NY, USA, 2013, p. 7–10. URL: https://doi.org/10.1145/2513150.2513159.

[5] F. Garcin, B. Faltings, O. Donatsch, A. Alazzawi, C. Bruttin, A. Huber, Offline and online evaluation of news recommender systems at swissinfo.ch, in: Proceedings of the 8th ACM Conference on Recommender Systems, RecSys '14, Association for Computing Machinery, New York, NY, USA, 2014, p. 169–176. URL: https://doi.org/10.1145/2645710.2645745.

[6] J. Lian, F. Zhang, X. Xie, G. Sun, Towards better representation learning for personalized news recommendation: a multi-channel deep fusion approach, in: Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18, International Joint Conferences on Artificial Intelligence Organization, 2018, pp. 3805–3811. doi:10.24963/ijcai.2018/529.

[7] Z. Lu, Z. Dou, J. Lian, X. Xie, Q. Yang, Content-based collaborative filtering for news topic recommendation, in: Proceedings of the AAAI Conference on Artificial Intelligence, 2015. doi:10.1609/aaai.v29i1.9183.

[8] M. Karimi, D. Jannach, M. Jugovac, News recommender systems – survey and roads ahead, Information Processing & Management 54 (2018) 1203–1227. URL: https://www.sciencedirect.com/science/article/pii/S030645731730153X. doi:10.1016/j.ipm.2018.04.008.

[9] A. Lommatzsch, B. Kille, S. Albayrak, Incorporating context and trends in news recommender systems, in: Proceedings of the International Conference on Web Intelligence, WI '17, Association for Computing Machinery, New York, NY, USA, 2017, p. 1062–1068. URL: https://doi.org/10.1145/3106426.3109433.

[10] Ö. Özgöbek, J. A. Gulla, R. C. Erdur, A survey on challenges and methods in news recommendation, in: WEBIST, 2014. doi:10.5220/0004844202780285.

[11] J. Domann, J. Meiners, L. Helmers, A. Lommatzsch, Real-time news recommendations using apache spark, in: CLEF, 2016. URL: http://ceur-ws.org/Vol-1609/16090628.pdf.

[12] K. Park, J. Lee, J. Choi, Deep neural networks for news recommendations, in: Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, CIKM '17, Association for Computing Machinery, New York, NY, USA, 2017, p. 2255–2258. URL: https://doi.org/10.1145/3132847.3133154.

[13] V. Kumar, D. Khattar, S. Gupta, M. Gupta, V. Varma, Deep neural architecture for news recommendation., in: CLEF (Working Notes), 2017. URL: http://ceur-ws.org/Vol-1866/paper_85.pdf.

[14] S. Raza, C. Ding, News recommender system: a review of recent progress, challenges, and opportunities, Artificial Intelligence Review (2021) 1–52. doi:10.1007/s10462-021-10043-x.

[15] C. Wu, F. Wu, Y. Huang, X. Xie, Personalized news recommendation: A survey, arXiv 2106.08934 (2021). URL: https://arxiv.org/abs/2106.08934.

[16] C. Wu, F. Wu, M. An, J. Huang, Y. Huang, X. Xie, Npa: Neural news recommendation with personalized attention, in: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD '19, Association for Computing Machinery, New York, NY, USA, 2019, p. 2576–2584. URL: https://doi.org/10.1145/3292500.3330665.

[17] X. Wang, L. Yu, K. Ren, G. Tao, W. Zhang, Y. Yu, J. Wang, Dynamic attention deep model for article recommendation by learning human editors' demonstration, in: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '17, Association for Computing Machinery, New York, NY, USA, 2017, p. 2051–2059. URL: https://doi.org/10.1145/3097983.3098096.

[18] C. Wu, F. Wu, M. An, J. Huang, Y. Huang, X. Xie, Neural news recommendation with attentive multi-view learning, in: Proceedings of the 28th International Joint Conference on Artificial Intelligence, IJCAI'19, AAAI Press, 2019, p. 3863–3869. URL: https://dl.acm.org/doi/10.5555/3367471.3367578. doi:10.5555/3367471.3367578.

[19] C. Wu, F. Wu, M. An, Y. Huang, X. Xie, Neural news recommendation with topic-aware news representation, in: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Florence, Italy, 2019, pp. 1154–1159. URL: https://aclanthology.org/P19-1110.

[20] M. An, F. Wu, C. Wu, K. Zhang, Z. Liu, X. Xie, Neural news recommendation with long- and short-term user representations, in: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Florence, Italy, 2019, pp. 336–345. URL: https://aclanthology.org/P19-1033.

[21] H. Wang, F. Zhang, X. Xie, M. Guo, Dkn: Deep knowledge-aware network for news recommendation, in: Proceedings of the 2018 World Wide Web Conference, WWW '18, International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE, 2018, p. 1835–1844. URL: https://doi.org/10.1145/3178876.3186175.

[22] D. Liu, J. Lian, S. Wang, Y. Qiao, J.-H. Chen, G. Sun, X. Xie, Kred: Knowledge-aware document representation for news recommendations, in: Fourteenth ACM Conference on Recommender Systems, 2020, p. 200–209. doi:10.1145/3383313.3412237.

[23] T. Qi, F. Wu, C. Wu, Y. Huang, Personalized News Recommendation with Knowledge-Aware Interactive Matching, Association for Computing Machinery, New York, NY, USA, 2021, p. 61–70. doi:10.1145/3404835.3462861.

[24] C. Wu, F. Wu, T. Qi, Y. Huang, Empowering news recommendation with pre-trained language models, in: SIGIR '21: The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, Canada, July 11-15, 2021, ACM, 2021, pp. 1652–1656. URL: https://doi.org/10.1145/3404835.3463069.

[25] L. Zhang, P. Liu, J. A. Gulla, Dynamic attention-integrated neural network for session-based news recommendation, Machine Learning 108 (2019) 1851–1875. doi:10.1007/s10994-018-05777-9.

[26] L. Hu, S. Xu, C. Li, C. Yang, C. Shi, N. Duan, X. Xie, M. Zhou, Graph neural news recommendation with unsupervised preference disentanglement, in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Online, 2020, pp. 4255–4264. URL: https://aclanthology.org/2020.acl-main.392.

[27] C. Wu, F. Wu, Y. Huang, X. Xie, User-as-graph: User modeling with heterogeneous graph

pooling for news recommendation, in: Z.-H. Zhou (Ed.), Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21, International Joint Conferences on Artificial Intelligence Organization, 2021, pp. 1624–1630. doi:10.24963/ijcai.2021/224.

[28] S. Ge, C. Wu, F. Wu, T. Qi, Y. Huang, Graph Enhanced Representation Learning for News Recommendation, Association for Computing Machinery, New York, NY, USA, 2020, p. 2863–2869. doi:10.1145/3366423.3380050.

[29] C. Wu, F. Wu, S. Ge, T. Qi, Y. Huang, X. Xie, Neural news recommendation with multi-head self-attention, in: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Association for Computational Linguistics, Hong Kong, China, 2019, pp. 6389–6394. doi:10.18653/v1/D19-1671.

[30] R. Xie, C. Ling, Y. Wang, R. Wang, F. Xia, L. Lin, Deep feedback network for recommendation, in: C. Bessiere (Ed.), Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20, International Joint Conferences on Artificial Intelligence Organization, 2020, pp. 2519–2525. doi:10.24963/ijcai.2020/349, main track.

[31] C. Wu, F. Wu, T. Qi, Y. Huang, Feedrec: News feed recommendation with various user feedbacks, in: Proceedings of The Web Conference 2022, 2022. URL: https://arxiv.org/abs/2102.04903, to appear.

[32] T. Qi, F. Wu, C. Wu, P. Yang, Y. Yu, X. Xie, Y. Huang, HieRec: Hierarchical user interest modeling for personalized news recommendation, in: Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), Association for Computational Linguistics, Online, 2021, pp. 5446–5456. URL: https://aclanthology.org/2021.acl-long.423.

[33] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, BERT: Pre-training of deep bidirectional transformers for language understanding, in: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), Association for Computational Linguistics, Minneapolis, Minnesota, 2019, pp. 4171–4186. URL: https://aclanthology.org/N19-1423.

[34] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, I. Polosukhin, Attention is all you need, in: I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, R. Garnett (Eds.), Advances in Neural Information Processing Systems, volume 30, Curran Associates, Inc., 2017. URL: https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

[35] F. Wu, Y. Qiao, J.-H. Chen, C. Wu, T. Qi, J. Lian, D. Liu, X. Xie, J. Gao, W. Wu, M. Zhou, MIND: A large-scale dataset for news recommendation, in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Online, 2020, pp. 3597–3606. URL: https://aclanthology.org/2020.acl-main.331.

[36] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, V. Stoyanov, Roberta: A robustly optimized bert pretraining approach, arXiv 1907.11692

(2019). URL: https://arxiv.org/abs/1907.11692.

[37] E. del Barrio, J. A. Cuesta-Albertos, C. Matrán, An Optimal Transportation Approach for Assessing Almost Stochastic Order, Springer International Publishing, Cham, 2018, pp. 33–44. doi:10.1007/978-3-319-73848-2_3.

[38] R. Dror, S. Shlomov, R. Reichart, Deep dominance - how to properly compare deep neural models, in: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Florence, Italy, 2019, pp. 2773–2785. URL: https://aclanthology.org/P19-1266.

[39] S. Rendle, W. Krichene, L. Zhang, J. Anderson, Neural collaborative filtering vs. matrix factorization revisited, in: Proceeedings of the Fourteenth ACM Conference on Recommender Systems, Association for Computing Machinery, New York, NY, USA, 2020, p. 240–248. doi:10.1145/3383313.3412488.

[40] B. Kille, A. Lommatzsch, Defining a meaningful baseline for news recommender systems, in: INRA@RecSys, 2019. URL: http://ceur-ws.org/Vol-2554/paper_04.pdf.