Transformers cannot Recommend Love, not yet anyway

Aditya Somasundaram [*1] **Pranav Kumar** [*1]

Abstract

By enhancing personalization and privacy, recommender systems have been able to expand their reach thanks to the development of artificial intelligence (AI) systems. In this work, we aim to combine the representation of textual and visual data modalities into a single latent space and examine how the recommendation system functions (Liu et al., 2024). We employ cosine similarity (Fkih, 2022) as the recommendation method in conjunction with the Flava model to extract multimodal embeddings. Using styleGANs for image generation and random selection from the typical dating app questions, we create artificial personalities for users. Additionally, we experiment with five text-only models including meta-llama-3-8B. Through feature analysis, we provide a framework for assessing a model's comprehension of the underlying structure of data. Despite their shortcomings, we discover that our results offer a fresh approach to user profile recommendations.

1. Introduction

Neural networks (Portugal et al., 2018; Zhang & Liu, 2021) have helped recommendation systems (Resnick & Varian, 1997; Isinkaye et al., 2015) to better capture relationships between users and items. Current systems even go so far as to consider weather and time. It's not uncommon to see food delivery apps suggest a hot snack to areas that are raining. With federated learning (Yang et al., 2020), autoencoders (Li & She, 2017) to handle sparsity and learn latent representations for users and items, recurrent neural networks to handle sequential recommendation tasks (Liu & Singh, 2016), and many more, the evolution of AI systems has enabled recommender systems to go farther by enhancing personalization and privacy (Javeed et al., 2023).

When deciding whether or not to like a profile on dating apps, users consider two types of information: the profile's images and the introduction (van Kooten & Schouten, 2021). According to (Brozovsky & Petricek, 2007), a conventional recommendation system would treat profiles as a single entity and function accordingly. Our goal is to integrate the representation of both data modalities into a single latent space and examine how the recommendation system functions (Liu et al., 2024). A more recent development of transformer models' capabilities is the integration of data from various modalities (Rahman et al., 2020). It's intriguing to investigate whether doing so would enhance recommender systems and, in turn, the dating app user experience. While our research focuses on dating apps, it can also be applied to other applications that use data with multiple modalities, such as movie recommendations (video and text data), grocery and clothing retail (product images and descriptions), etc.

2. Related Works

2.1. Multimodal Transformer Models

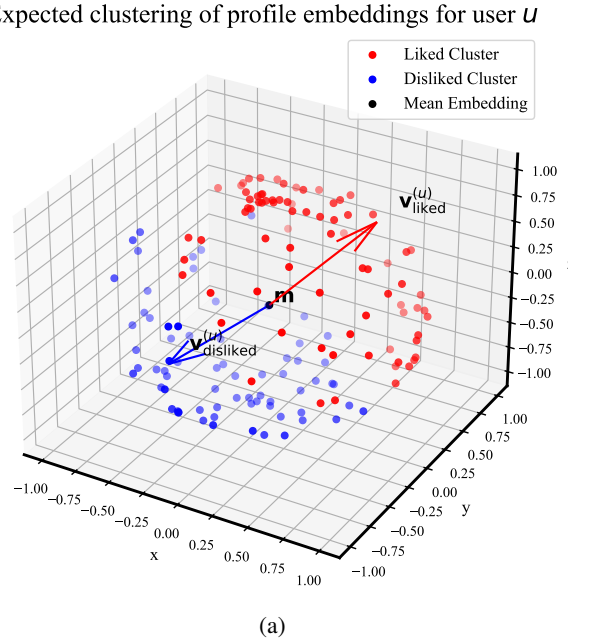
Since their inception, transformer models have changed to account for various data modalities (Vaswani, 2017). The most recent models on the market (Achiam et al., 2023; Touvron et al., 2023; Team et al., 2024) effortlessly incorporate data from text, images, audio, video, and other sources. These models capture relationships and alignments across various data types by expanding on the concepts of transformers, which were initially created for NLP tasks. Each modality—text as tokens, images as patches, and audio as spectrograms—is represented as a series of embeddings that are transformed into a single latent space (Gao et al., 2020). To align features across modalities, these embeddings are then subjected to co-attention and cross-attention mechanisms (Chen et al., 2021). Semantically aligned features are near to one another in the joint representation space created by the transform layers, which gradually improve the embeddings ("dog" in text and dog in image). For example, CLIP aligns image and text embeddings by maximizing similarity between matching pairs and minimizing similarity between mismatching ones. These models accomplish this by using contrastive loss or alignment loss (Radford et al., 2021). Image captioning, visual question answering, image-text retrieval, multimodal search engines, healthcare (combining textual notes and image data for diagnosis), robotics, and other fields are just a few of the real-world applications for multimodal models.

2.2. Recommendation Systems

Information retrieval and collaborative filtering served as the foundation for the idea of recommender systems (Baeza-Yates et al., 1999; He et al., 2017). The art of recommendation is used extensively in contemporary society today, including in personalized ads, product recommendations, social media content suggestions, meal recommendations from delivery services, etc. Recommender systems use user data to make recommendations and forecast user preferences. The objective is to create a system that is accurate, efficient, and customized while guaranteeing originality and exploration in recommendations. Both explicit input data, likes and ratings for products, and implicit input data, like clicks and past purchases, are used by modern systems (Sharma & Singh, 2016; Stankevičius & Lukoševičius, 2024). To improve the recommendation process, the collaborative filtering (CF) mechanism employs item-based CF (items similar to those a user has previously liked) and user-based CF (items based on users with similar tastes). Genre, keywords, hashtags, and other item attributes are used in content-based filtering in modern systems as well. These days, the development of better and more reliable systems has been facilitated by graph-based techniques, reinforcement learning, and privacy-preserving learning through decentralized data processing (Singhal et al., 2017).

2.3. Multimodal transformers in recommendation systems

(Nikzad-Khasmakhi et al., 2021) offers the first multimodal representation method for developing a recommender system that uses the Scopus Dataset to identify subject matter experts. (Gu et al., 2020) introduces a recommender system for e-commerce that uses a mixture-of-experts transformer model to generate suggestions based on clicks and the order of items in the cart. A conversational recommender system (CRS) has been developed by (Zou et al., 2022) to assist users with genres of films, books, television series, and more. In their transformer block, (Zou et al., 2022) and (Gu et al., 2020) both employ a single modality. The majority of contemporary recommender systems employ multimodal encoders (coarse-grained, fine-grained, and attention) and refined feature interactions to obtain the optimal latent space representation of data, as (Liu et al., 2024) notes in their survey. In our work, we employ cosine similarity (Fkih, 2022) as the recommendation method and the Flava model (Singh et al., 2022) to extract multimodal embeddings. We examine whether Flava can complete the recommendation task without further fine-tuning or training. Additionally, we test the text modality on five text-only models, including meta-llama-3-8B.



(a)

Figure 1. A visual on how the clustering of embeddings should work in 3 dimensions. The multimodal embeddings from the model would, informally speaking, contain more information as it has a higher dimension (768). Note that the vector cloud is shifted to the new origin (See 3.3).

3. Methodology

Since user information is confidential across these apps, we use openly accessible generative AI tools to create our own dataset.

3.1. Dataset creation

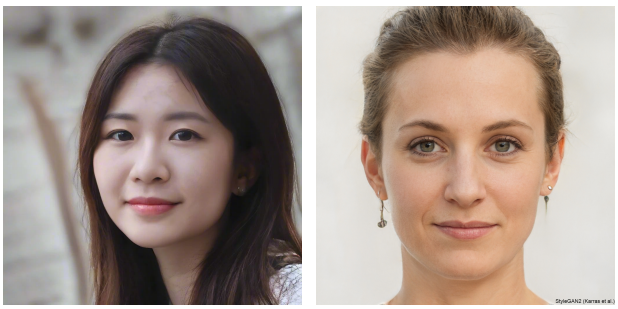
3.1.1. Images of users

Modern generative AI was used to create the user profile data. SytleGAN was used to create each user's image (Karras et al., 2020). High-level characteristics of the image, such as the face's position, expression, hairstyle, freckles, and other details, can be defined by this algorithm. This was used to generate data that is diverse in terms of age, ethnicity, and style. A selection of the generated images are displayed in Figure 2.

3.1.2. Personalities of users

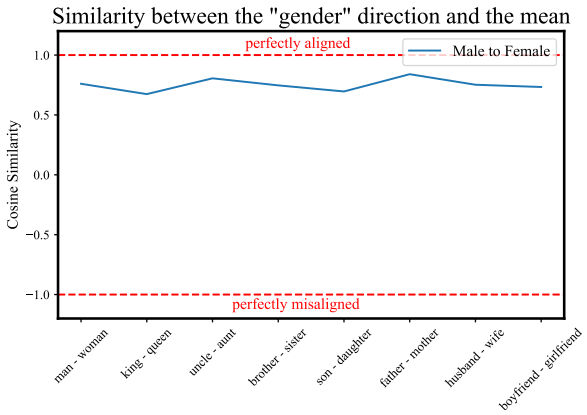
We make the crucial supposition that a person's hobbies and interests are unrelated to their physical characteristics. This would enable text to be generated at random and independent of user-generated images. We create synthetic user personas by selecting users at random from the typical questions they ask on dating apps. • Age • Height • Hobbies • Drinking preference • Smoking preference • Looking

(a) Male, age 25 (b) Male, age 23

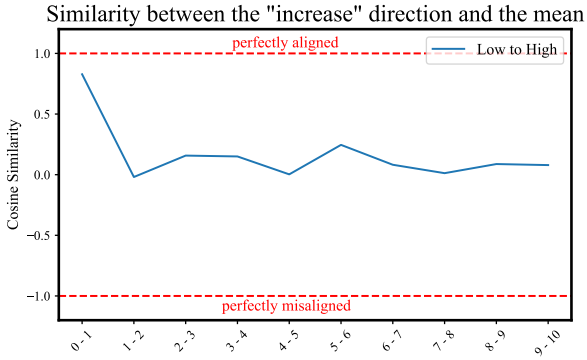


(c) Female, age 22 (d) Female, age 25

Figure 2. Images generated by styleGAN for dataset curation. Although these closely resemble real human images, they are different from the those typically found on dating applications (example: images with a background scenery, people doing a hobby or an activity).



(a) We analyze the clustering of directions which take masculine term to its feminine counterpart



(b) We analyze the clustering of directions which take a lower number to a higher number

Figure 3. Analyzing the understanding of gender and inequalities by the model

for. These keywords are then passed through GPT-4o to obtain the personality and a profile introduction. We also add text traits which are implicit to make the personalities more evident and unique.

Example 1. The string "gender: male, height: 6'0, hobbies: singing, reading, baking, gaming, playing an instrument, drinking: never, smoking: socially, looking for: something casual, text traits: Laid-back, Empathetic" is sent to the generative model. The output is as follows:

Profile: I'm someone who enjoys the simpler things in life, like singing, reading, and baking when I'm not getting lost in a good game or strumming my guitar. I prefer to take things easy and keep things relaxed, and I appreciate connecting with others who share a laid-back outlook. While I'm all about keeping it casual, I'm also the kind of person who values understanding and empathy, making sure that those around me feel heard and cared for.

Note that the model is instructed to weave the text traits into the profile introduction. The prompt used is provided in Appendix B.

3.1.3. The Preference Collection

Preference data b_{ij} is gathered from a survey. Additionally, the respondents are asked to write a description of their ideal partner; we use this information in our analysis as well as for a good initialization for the type vectors $\mathbf{v}$. The responses are manually filtered in order to further refine this preference data.

3.2. Building our recommendation system

Using the relative placement of user profiles in an embedding space, we suggest a way to personalize recommendations by creating "liked" and "disliked" vectors for every user. To shift the vector cloud to zero mean, we first calculate the mean embedding across all profiles, treating it as an "average profile," and then deduct this mean from each user's embedding (See Equations 1 and 2). By taking this step, the origin is guaranteed to reflect the typical profile. We assume that significant semantic variances, like a preference for humor or outdoor activities, are captured by the

relative directions of profiles to the mean (Li et al., 2020). The "disliked" and "liked" vectors are derived from the average direction of the embeddings corresponding to profiles that the user finds appealing. A new user's alignment is determined by subtracting the average profile from their embedding and comparing its orientation with the "liked" and "disliked" vectors using cosine similarity (Vozalis & Margaritis, 2003). We assume that users make selections based on some unquantifiable preferences. We attempt to use these profile embeddings to analyze possibilities of underlying structures based on the relative positioning from the mean profile.

3.3. The formulation

Let $\mathbf{v_i}$ be the embedding of the i-th user profile, where $i \in \{1, 2, \cdots, N\}$. Let $\mathbf{v}^{(\mathbf{u})}_{\text{liked}}$ and $\mathbf{v}^{(\mathbf{u})}_{\text{disliked}}$ be the "liked" and "disliked" vectors for user u. We obtain a mean vector $\mathbf{m} = \frac{1}{N}\sum_{i=1}^{N} \mathbf{v}_i$ and shift the vector cloud: $\mathbf{v}_i^{\text{centered}} = \mathbf{v}_i - \mathbf{m}$. For a given user u, let $\mathcal{L}_u$ be the set of embeddings of profiles the user likes, and $\mathcal{D}_u$ be the set of embeddings of profiles the user dislikes. We compute the "liked" and "disliked" vectors as the mean of the respective centered embeddings:

$$\mathbf{v}^{(u)}_{\text{liked}} = \frac{1}{|\mathcal{L}_u|} \sum_{\mathbf{v} \in \mathcal{L}_u} (\mathbf{v} - \mathbf{m}) \tag{1}$$

$$\mathbf{v}^{(u)}_{\text{disliked}} = \frac{1}{|\mathcal{D}_u|} \sum_{\mathbf{v} \in \mathcal{D}_u} (\mathbf{v} - \mathbf{m}) \tag{2}$$

For a new user, we first center their embedding $\mathbf{v}^{\text{centered}}_{\text{new}} = \mathbf{v}_{\text{new}} - \mathbf{m}$ and determine its alignment with $\mathbf{v}^{(u)}_{\text{liked}}$ and $\mathbf{v}^{(u)}_{\text{disliked}}$: $S_{\text{liked}} = \frac{\mathbf{v}^{\text{centered}}_{\text{new}} \cdot \mathbf{v}^{(u)}_{\text{liked}}}{\|\mathbf{v}^{\text{centered}}_{\text{new}}\| \cdot \|\mathbf{v}^{(u)}_{\text{liked}}\|}$ and $S_{\text{disliked}} = \frac{\mathbf{v}^{\text{centered}}_{\text{new}} \cdot \mathbf{v}^{(u)}_{\text{disliked}}}{\|\mathbf{v}^{\text{centered}}_{\text{new}}\| \cdot \|\mathbf{v}^{(u)}_{\text{disliked}}\|}$. The higher score determines the prediction. Figure 1 helps understand this with a visual.

3.4. The cold start problem

There are two aspects of the "cold start problem" in recommender systems (Lika et al., 2014). User cold start deals with recommending items to users with no prior data. Item cold start is the problem of recommending new items with no interaction history. We believe that the former will be solved with a good initialization, provided the description is detailed enough. Moreover, the latter will not be an issue as a new profile will receive a vector embedding $\mathbf{i}$ instantly, subsequently recommending it to users whose $\mathbf{v}^{(u)}_{\text{liked}}$ is proximal to $\mathbf{v_i}$.

4. Analysis and Results

We use the Flava model (Singh et al., 2022) to obtain the multimodal embeddings. It is imperative to analyze if this model understands semantics and in specific, differentiates between traits in the process of partner selection. For instance, some people prefer tall partners, making it crucial for the model to understand this trait (Ushio et al., 2021). Can the model understand numbers and cluster the text information based on traits (example: heights)? First, we send in individual words to the model to understand its capabilities.

4.1. Does the model understand gender?

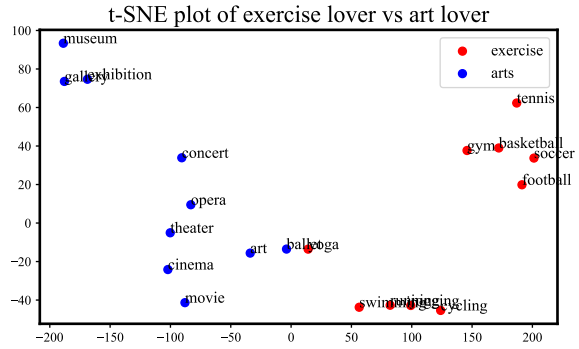
$\mathbf{king - queen \simeq man - woman}$ is a well-known example (Ethayarajh, 2019). Informally, the direction that corresponds to $\mathbf{man - woman}$ denotes a gender shift. We use additional gender-based pairs to test our model. The term $\mathbf{masculine - feminine}$ is calculated using the embeddings of the pairs {(man, woman), (king, queen), (uncle, aunt), (brother, sister), (son, daughter), (father, mother), (husband, wife), (boyfriend, girlfriend)}. The direction encoding gender is assumed to be the mean of these directions. Each direction's cosine similarity is calculated in relation to the mean and displayed below. Each tuple in our set has roughly the same vector difference in the embedding space, as shown in Figure 3a, i.e, there is a clustering of these vectors. We can draw the conclusion that the model does comprehend gender data well.

4.2. Does the model understand numbers?

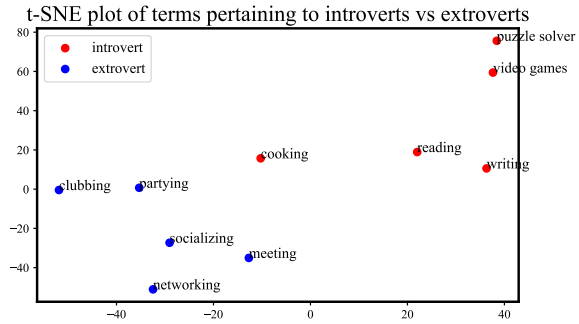
Transformer models have developed to comprehend entity growth and decrease. The analysis is repeated using the following numbers: $\{(0, 1), (1, 2), (2, 3), (3, 4), \cdots, (9, 10)\}$. Below is the cosine similarity between the mean direction and the calculated directions $\mathbf{high - low}$. Figure 3b shows that the model's comprehension of the terms "increase" and "decrease" is lacking. There is no sense of similarity because each individual change in direction is nearly orthogonal to the mean.

4.3. Does the model have a sense of traits?

Can the model distinguish between the following sets of words: {tennis, gym, jogging, running, swimming, cycling, yoga} and {museum, theater, concert, opera, ballet, cinema}? A person who enjoys exercising is represented by the former set of words, while a person who enjoys fine arts is represented by the latter. The t-SNE plots of several differentiators, such as introvert versus extrovert, exercise enthusiast versus fine art enthusiast, and luxury versus simplicity, are shown below. We can infer that the model can distinguish between terms linked to different personality traits and interests from the figures 4 and 8. This would



(a) We cluster individual words which pertain to people who exercise and play sports and terms associated with people who love fine arts and theater. Notice that the embeddings from the model have a clear clustering, indicating understanding between the two interests.



(b) We cluster terms associated with introverted personalities against those with extroverted ones. Once again, the embeddings from the model form separable clusters, indicating understanding of the terms associated with the two personality types.

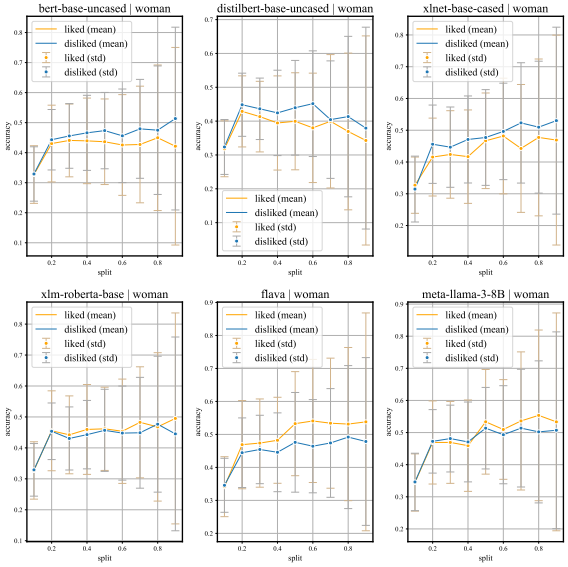
Figure 4. Analyzing the understanding of terms associated with opposing personalities

improve the recommender's performance by allowing the model to comprehend the interests connected to profiles.

4.4. Performance of the Recommender System

First, we use the Flava model to obtain embeddings for the entire profile. We then test our recommendation model on different train-test splits, examining performance for men's and women's profiles independently. Our findings suggest that Flava model is consistently poor at recommending profiles. In order to find out if the recommender can perform well when restricted to the text modality, we do experiments with five text-only models (bert, distilbert, xlnet, xlm-roberta, meta-llama-3-8B). Among our results, Flava is the only model accommodating multimodal data.

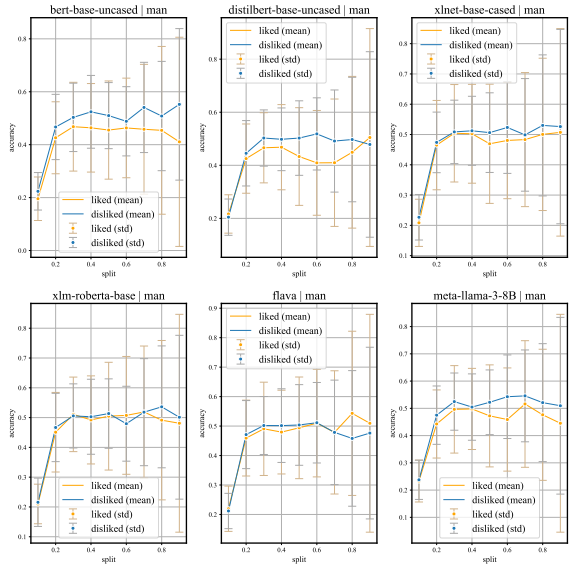
We plot the accuracy for both classes and vary the range from 10% to 90% to see how well we can perform across different train-test splits. Figures 6 and 5 illustrate the variance involved in the experiments, which were conducted



(a)

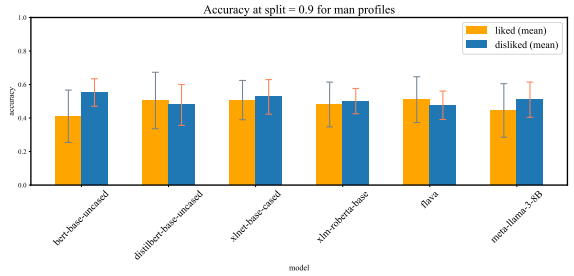
Figure 5. The performance of the recommendation system for women profiles for each model and each split is depicted above. Note that all models are unable to perform well, hovering around a random recommendation system (50%).

5 times to negate any randomness. We can see from the graphs that none of the models can function. The models hardly outperforms a random classifier; it cannot reliably distinguish between "liked" and "disliked.". This conclusion is supported by the variance. Regardless of the train-test split, all models have an accuracy of about 50% on average. It is interesting to note that Llama-3-8B is unable to perform well, despite being more powerful than the other models (See Figure 7). The accuracy for specific parameters of train-test split, gender, and model is available in Appendix E.

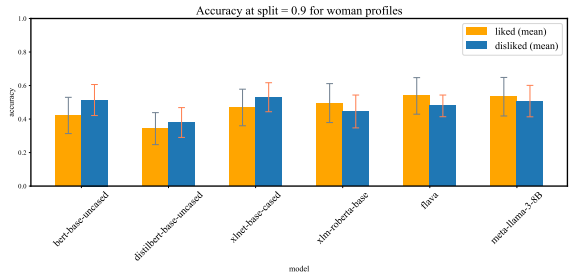


(a)

Figure 6. The performance of the recommendation system for men profiles for each model and each split is depicted above. Note that all models are unable to perform well, hovering around a random recommendation system (50%).



(a) Performance on profiles of men for all models



(b) Performance on profiles of women for all models

Figure 7. The recommendation system's effectiveness for every model with a 0.9 split between training and testing. Even though we observe that certain models perform better than others, we are unable to draw firm conclusions because the accuracy is only about 50%, which is equivalent to a random recommender.

5. Limitations

5.1. Curation of dataset

The dataset we curated deviate from real data present in dating applications. The images found in profiles are diverse and are focussed on more than just the face unlike images generated by styleGANs. We make an assumption that the images in the profiles has no correlation with the matching text, but this is not necessarily true. For example: a person who loves sports or museums would probably have a profile picture of them playing a sport or out at a museum.

5.2. Lack of responses

There is a lack of diversity in the age group of our respondents. We also focus on few profiles (30 for men and women each). Both of these could potentially lead to skewed data, consequently leading to a biased inferences. In order to test our process without bias, an increase in the number of profiles as well as number of responders is required. The authors have received reviews from responders stating quality and diversity could be improved.

5.3. No fine-tuning

Using these models' embeddings, we assess them on this downstream task without additional training or fine-tuning. Although we anticipate that the models will naturally recognize the features that some users find most appealing, there is a chance that fine-tuning the models could result in improved performance.

6. Conclusion and Future Work

While originally, we were aiming to find a golden recipe for recommender systems with multimodal models under the assumption that they understand the unquantifiable factors which humans look at while processing information, we show that there is no simple and straightforward answer. However, we do believe that there exists a deeper underlying perspective on the problem. We present a framework for assessing a model's capacity to understand the underlying structure of data by analyzing its features. Although constrained by limited computational resources and therefore relying on models that may be considered suboptimal by today's standards, we believe that our framework offers reliable method for evaluating a model's ability to recognize characteristics and subtleties that people naturally look for when processing data. In the future, we plan to improve this work by analyzing embeddings from state of the art language models and foundational models (Zhao et al., 2023).

The quality of the data may be one factor contributing to our subpar performance. While the generated data mimics that which is found in real world applications, there are nuances

which could cause the model to under perform (See section 5.1). The performance could also increase if we are able to receive a higher number of preferences from peers and responders.

We assume that humans rank these data based on some intangible factors such as humor, confidence, hobbies, expressiveness, and the way people portray themselves. By construction, a model which performs well inherently understands these "metrics" and therefore is able to understand such factors. It would be interesting to investigate if bigger models trained on more data with higher quality will be able to perform well.

Acknowledgements

We thank Prof. Micah Goldblum, whose discussions and insightful observations were invaluable during this research. We also acknowledge the assistance of our respondents in gathering data and assessing the accuracy of our findings.

References

Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F. L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Baeza-Yates, R., Ribeiro-Neto, B., et al. *Modern information retrieval*, volume 463. ACM press New York, 1999.

Brozovsky, L. and Petricek, V. Recommender system for online dating service. *arXiv preprint cs/0703042*, 2007.

Chen, C.-F. R., Fan, Q., and Panda, R. Crossvit: Cross-attention multi-scale vision transformer for image classification. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 357–366, 2021.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. An image is worth 16x16 words: Transformers for image recognition at scale. *CoRR*, abs/2010.11929, 2020. URL https://arxiv.org/abs/2010.11929.

Ethayarajh, K. Rotate king to get queen: Word relationships as orthogonal transformations in embedding space. *arXiv preprint arXiv:1909.00504*, 2019.

Fkih, F. Similarity measures for collaborative filtering-based recommender systems: Review and experimental comparison. *Journal of King Saud University-Computer and Information Sciences*, 34(9):7645–7669, 2022.

Gao, J., Li, P., Chen, Z., and Zhang, J. A survey on deep learning for multimodal data fusion. *Neural Computation*, 32(5):829–864, 2020.

Gu, Y., Ding, Z., Wang, S., Zou, L., Liu, Y., and Yin, D. Deep multifaceted transformers for multi-objective ranking in large-scale e-commerce recommender systems. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, CIKM '20, pp. 2493–2500, New York, NY, USA, 2020. Association for Computing Machinery. ISBN 9781450368599. doi: 10.1145/3340531.3412697. URL https://doi.org/10.1145/3340531.3412697.

He, X., Liao, L., Zhang, H., Nie, L., Hu, X., and Chua, T.-S. Neural collaborative filtering. In *Proceedings of the 26th international conference on world wide web*, pp. 173–182, 2017.

Isinkaye, F. O., Folajimi, Y. O., and Ojokoh, B. A. Recommendation systems: Principles, methods and evaluation. *Egyptian informatics journal*, 16(3):261–273, 2015.

Javeed, D., Saeed, M. S., Kumar, P., Jolfaei, A., Islam, S., and Islam, A. N. Federated learning-based personalized recommendation systems: An overview on security and privacy challenges. *IEEE Transactions on Consumer Electronics*, 2023.

Karras, T., Laine, S., Aittala, M., Hellsten, J., Lehtinen, J., and Aila, T. Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8110–8119, 2020.

Li, B., Zhou, H., He, J., Wang, M., Yang, Y., and Li, L. On the sentence embeddings from pre-trained language models. *arXiv preprint arXiv:2011.05864*, 2020.

Li, X. and She, J. Collaborative variational autoencoder for recommender systems. In *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 305–314, 2017.

Lika, B., Kolomvatsos, K., and Hadjiefthymiades, S. Facing the cold start problem in recommender systems. *Expert systems with applications*, 41(4):2065–2073, 2014.

Liu, D. Z. and Singh, G. A recurrent neural network based recommendation system. In *International Conference on Recent Trends in Engineering, Science & Technology*, 2016.

Liu, Q., Hu, J., Xiao, Y., Zhao, X., Gao, J., Wang, W., Li, Q., and Tang, J. Multimodal recommender systems: A survey. *ACM Computing Surveys*, 57(2):1–17, 2024.

Nikzad-Khasmakhi, N., Balafar, M., Reza Feizi-Derakhshi, M., and Motamed, C. Berters: Multimodal representation learning for expert recommendation system with transformers and graph embeddings. *Chaos, Solitons Fractals*, 151:111260, 2021. ISSN 0960-0779. doi: https://doi.org/10.1016/j.chaos.2021.111260. URL https://www.sciencedirect.com/science/article/pii/S0960077921006147.

Portugal, I., Alencar, P., and Cowan, D. The use of machine learning algorithms in recommender systems: A systematic review. *Expert Systems with Applications*, 97: 205–227, 2018.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PMLR, 2021.

Rahman, W., Hasan, M. K., Lee, S., Zadeh, A., Mao, C., Morency, L.-P., and Hoque, E. Integrating multimodal information in large pretrained transformers. In *Proceedings of the conference. Association for Computational Linguistics. Meeting*, volume 2020, pp. 2359. NIH Public Access, 2020.

Resnick, P. and Varian, H. R. Recommender systems. *Communications of the ACM*, 40(3):56–58, 1997.

Sharma, R. and Singh, R. Evolution of recommender systems from ancient times to modern era: a survey. *Indian Journal of Science and Technology*, 9(20):1–12, 2016.

Singh, A., Hu, R., Goswami, V., Couairon, G., Galuba, W., Rohrbach, M., and Kiela, D. Flava: A foundational language and vision alignment model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15638–15650, 2022.

Singhal, A., Sinha, P., and Pant, R. Use of deep learning in modern recommendation system: A summary of recent works. *arXiv preprint arXiv:1712.07525*, 2017.

Stankevičius, L. and Lukoševičius, M. Extracting sentence embeddings from pretrained transformer models. *Applied Sciences*, 14(19), 2024. ISSN 2076-3417. doi: 10.3390/app14198887. URL https://www.mdpi.com/2076-3417/14/19/8887.

Team, G., Mesnard, T., Hardin, C., Dadashi, R., Bhupatiraju, S., Pathak, S., Sifre, L., Rivière, M., Kale, M. S., Love, J., et al. Gemma: Open models based on gemini research and technology. *arXiv preprint arXiv:2403.08295*, 2024.

Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.-A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

Ushio, A., Espinosa-Anke, L., Schockaert, S., and Camacho-Collados, J. Bert is to nlp what alexnet is to cv: Can pre-trained language models identify analogies? *arXiv preprint arXiv:2105.04949*, 2021.

van Kooten, J. and Schouten, A. Visual and textual cues on online dating profiles: What makes you swipe? 2021.

Vaswani, A. Attention is all you need. *Advances in Neural Information Processing Systems*, 2017.

Vozalis, E. and Margaritis, K. G. Analysis of recommender systems algorithms. In *The 6th Hellenic European Conference on Computer Mathematics & its Applications*, pp. 732–745, 2003.

Yang, L., Tan, B., Zheng, V. W., Chen, K., and Yang, Q. Federated recommendation systems. *Federated Learning: Privacy and Incentive*, pp. 225–239, 2020.

Zhang, M. and Liu, Y. A commentary of tiktok recommendation algorithms in mit technology review 2021. *Fundamental Research*, 1(6):846–847, 2021.

Zhao, W. X., Zhou, K., Li, J., Tang, T., Wang, X., Hou, Y., Min, Y., Zhang, B., Zhang, J., Dong, Z., et al. A survey of large language models. *arXiv preprint arXiv:2303.18223*, 2023.

Zou, J., Kanoulas, E., Ren, P., Ren, Z., Sun, A., and Long, C. Improving conversational recommender systems via transformer-based sequential modelling. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '22, pp. 2319–2324, New York, NY, USA, 2022. Association for Computing Machinery. ISBN 9781450387323. doi: 10.1145/3477495.3531852. URL https://doi.org/10.1145/3477495.3531852.

A. About flava

Flava uses independent encoders to extract unimodal representations of images and text, followed by a multimodal encoder to fuse and align the representations. The unimodal encoders for image and text use the ViT-B/16 architecture (Dosovitskiy et al., 2020). The model is trained using global contrastive loss along with techniques of masked multimodal modeling (MMM) and image-text matching (ITM). Joint unimodal and multimodal embeddings help NLP, and is achieved by full pretraining of Flava.

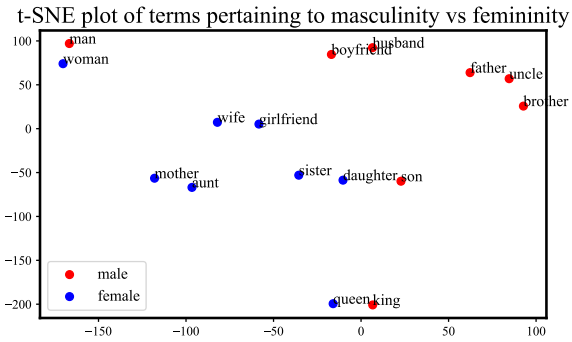
B. Prompt used

The following prompt was sent to gpt-4o: "I will give you a few key words related to a personality. I would like you to write a summary or an introduction which the person can use for their dating profile. I have also given a few text traits, weave them subtly into the text." A random selection of keywords was made from the following list. Each text generation was given two text traits and five hobbies.

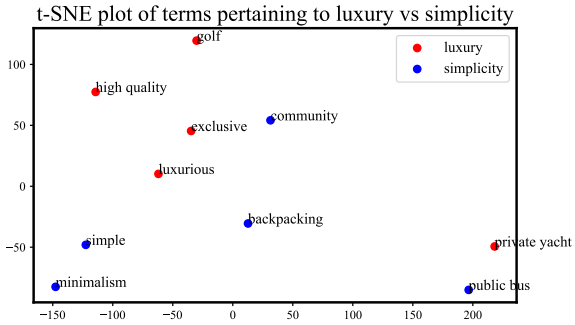
- **gender:** male, female
- **height:** 5'8, 5'9, 5'10, 5'11, 6'0, 6'1, 6'2
- **hobbies:** basketball, football, soccer, hiking, reading, writing, coding, swimming, running, gaming, cooking, baking, painting, drawing, dancing, singing, playing an instrument, yoga, meditation
- **drinking preference:** socially, rarely, never, often
- **smoking preference:** socially, rarely, never, often
- **looking for:** serious only, somewhat serious, something casual
- **text traits:** adventurous, compassionate, curious, empathetic, optimistic, introspective, witty, analytical, creative, generous, bold, laid-back, resilient, passionate, thoughtful, charming, sarcastic, sincere, playful, perceptive

C. More t-SNE plots to check differentiators in traits for Flava

The t-SNE plots are depicted for the masculine vs. feminine terms (Figure 8a) and the terms pertaining to luxury vs simplicity (Figure 8b) are depicted. Notice that Figure 8a has a cluster around the male entities while simultaneously having the masculine term close to its corresponding feminine term. This shows that the model understands not only the gender, but also the counterparts for the terms. On the other hand, from figure 8b we can infer that the model has little to no understanding of the terms associated with luxurious lifestyle vs a simple one.



(a) We cluster individual words which pertain to people who exercise and play sports and terms associated with people who love fine arts and theater. Notice that the embeddings from the model have a clear clustering, indicating understanding between the two interests.

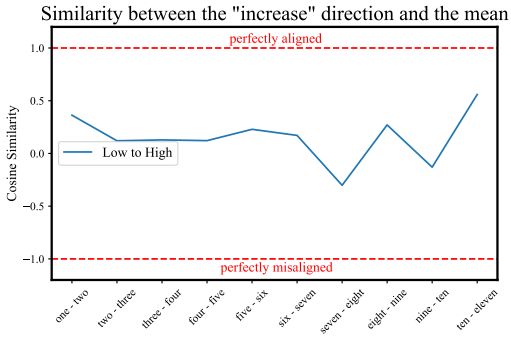


(b) We cluster terms associated with introverted personalities against those with extroverted ones. Once again, the embeddings from the model form separable clusters, indicating understanding of the terms associated with the two personality types.

Figure 8. Analyzing the understanding of terms associated with opposing personalities

D. Number analysis

To clarify that the Flava model is unable to understand numbers, we send the numbers in their word forms (Figure 9). We perform the same experiment mentioned in 4.2.



(a) We send in the numbers as “one”, “two”, ... to obtain the embeddings. The plot confirms our initial analysis that the model is unable to sense the concepts of increase or decrease.

Figure 9. Reaffirming that Flava is unable to understand numbers

E. Tabulated Results

The specific liked and disliked accuracies for each model, and train-test split is given below (Tables 1 and 2). Table 3 contains the accuracy for each gender.

Model	Split	Liked Accuracy	Disliked Accuracy
bert-base-uncased	0.1	0.2853	0.2816
bert-base-uncased	0.2	0.4402	0.4550
bert-base-uncased	0.3	0.4685	0.4867
bert-base-uncased	0.4	0.4437	0.4832
bert-base-uncased	0.5	0.4484	0.5106
bert-base-uncased	0.6	0.4538	0.5109
bert-base-uncased	0.7	0.4383	0.5113
bert-base-uncased	0.8	0.4745	0.4878
bert-base-uncased	0.9	0.4390	0.4900
distilbert-base-uncased	0.1	0.2713	0.2856
distilbert-base-uncased	0.2	0.4251	0.4493
distilbert-base-uncased	0.3	0.4272	0.4640
distilbert-base-uncased	0.4	0.4256	0.4602
distilbert-base-uncased	0.5	0.4080	0.4514
distilbert-base-uncased	0.6	0.3983	0.4506
distilbert-base-uncased	0.7	0.4035	0.4592
distilbert-base-uncased	0.8	0.3824	0.4513
distilbert-base-uncased	0.9	0.3910	0.4324
xlnet-base-cased	0.1	0.2731	0.2872
xlnet-base-cased	0.2	0.4592	0.4584
xlnet-base-cased	0.3	0.4590	0.4789
xlnet-base-cased	0.4	0.4340	0.4704
xlnet-base-cased	0.5	0.4729	0.5067
xlnet-base-cased	0.6	0.4957	0.5227
xlnet-base-cased	0.7	0.4714	0.5201
xlnet-base-cased	0.8	0.4816	0.5095
xlnet-base-cased	0.9	0.4686	0.5276
xlm-roberta-base	0.1	0.2897	0.2862
xlm-roberta-base	0.2	0.4585	0.4465
xlm-roberta-base	0.3	0.4717	0.4614
xlm-roberta-base	0.4	0.4806	0.4613
xlm-roberta-base	0.5	0.4902	0.4722
xlm-roberta-base	0.6	0.4800	0.4811
xlm-roberta-base	0.7	0.4714	0.4802
xlm-roberta-base	0.8	0.4750	0.4693
xlm-roberta-base	0.9	0.4800	0.4867

Table 1. Model-wise Accuracy Results by Split

Model	Split	Liked Accuracy	Disliked Accuracy
flava	0.1	0.2824	0.2860
flava	0.2	0.4481	0.4555
flava	0.3	0.4702	0.4719
flava	0.4	0.4680	0.4828
flava	0.5	0.5134	0.4862
flava	0.6	0.5282	0.4916
flava	0.7	0.5214	0.4882
flava	0.8	0.5316	0.4614
flava	0.9	0.5319	0.4757
meta-llama-3-8B	0.1	0.2873	0.2941
meta-llama-3-8B	0.2	0.4587	0.4812
meta-llama-3-8B	0.3	0.4771	0.4885
meta-llama-3-8B	0.4	0.4701	0.4947
meta-llama-3-8B	0.5	0.5063	0.5113
meta-llama-3-8B	0.6	0.5306	0.5121
meta-llama-3-8B	0.7	0.4967	0.5091
meta-llama-3-8B	0.8	0.5008	0.5173
meta-llama-3-8B	0.9	0.5233	0.5067

Table 2. Model-wise Accuracy Results by Split

Model	Gender	Liked Accuracy	Disliked Accuracy
bert-base-uncased	Man	0.4272	0.4780
bert-base-uncased	Woman	0.4299	0.4568
distilbert-base-uncased	Man	0.4229	0.4586
distilbert-base-uncased	Woman	0.3757	0.4089
xlnet-base-cased	Man	0.4593	0.4768
xlnet-base-cased	Woman	0.4451	0.4716
xlm-roberta-base	Man	0.4705	0.4775
xlm-roberta-base	Woman	0.4519	0.4349
flava	Man	0.4612	0.4615
flava	Woman	0.4856	0.4522
meta-llama-3-8B	Man	0.4471	0.4867
meta-llama-3-8B	Woman	0.4892	0.4746

Table 3. Model-wise Accuracy Results by Gender