Supplementary materials B2: Topic model selection

Author note:

This document contains supplementary materials for the article titled:

"Norm repair as an extension to the deliberative dialogue model"

The document details the topic model selection process and results. Further details and code for replication are available in a Python notebook (Supplementary Materials B).

We used a topic model to identify clusters of interview responses by Reddit moderators. To do this, we used a BERTopic approach (Grootendorst, 2022). We chose the default sentence embedding (Reimers & Gurevych, 2019), "all-MiniLM-L6-v2" [1], which processes each text into a 384 dimensional vector. We then used the Uniform Manifold Approximation and Projection (UMAP, McInnes et al., 2020) algorithm to reduce the dimensions of the embeddings, followed by the Hierarchical Density Based Clustering (HDBSCAN, McInnes et al., 2017) algorithm to perform the clustering.

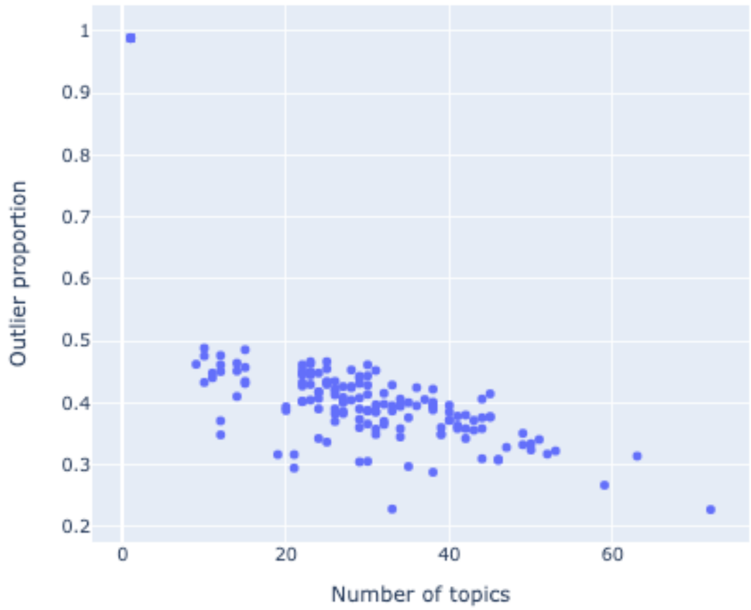
The resulting clusters of documents (the "topic") are then processed using a class-based term frequency inverse document frequency (cTF-IDF) matrix to produce a list of words that represent each topic. Finally, we fine-tuned the word list by, first, passing them to KeyBERT (Grootendorst, 2021) to identify the most meaningful words per topic. We then passed this word list alongside the representative documents for each topic to OpenAI's ChatGPT (OpenAI, 2023; see also Brown et al., 2020) generative language model to generate a unique name for each topic.

To select the most optimal hyper-parameters, we ran 157 models and tracked the number of topics they produced, the number of outliers, and the model coherence. To quantify the coherence, we used the C_V statistic, which, in a systematic study of coherence measures, was found to be the one that had the highest correlation with human ratings (Röder et al., 2015)[2]. Selecting the optimal topic model is difficult because the method is unsupervised (see Terragni et al., 2021; Laureate et al., 2023). As well as selecting optimal

[1] https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2
[2] See: https://github.com/MaartenGr/BERTopic/issues/90 for implementation with BERTopic.

hyper-parameters, running many models allowed us to observe the distribution of the number of topics (M=29.26, SD=12.67) and outliers (M=500.85, SD=159.61), providing us with an indication of where most models may lie (see Figure 1).

Figure B1: Plot of the number of topics against the proportion of outliers produced across 157 model sweeps.



Across the sweeps we altered the number of neighbors, the number of components, and the minimum distance for UMAP, the minimum cluster size for HDBSCAN, and the maximum ngram size for the Count Vectorizer. We only found the n-gram size ($r=0.79$, $p < 0.001$) and minimum cluster size ($r=0.17$, $p=0.05$) to have a statistically significant correlation with C_V at a 5% conventional level. All other parameter correlations were insignificant. The high correlation for the n-gram size is to be expected, as the higher the ngram, the more precise the terms in the final cTF-IDF matrix that is used during the calculation of C_V. Figure 2 plots the relationship between C_V and minimum cluster size parameter, where it can be observed that the coherence becomes asymptotic when the cluster size passes 9 or 10. Our highest performing model (C_V=0.92) produced a single topic and

1180 outliers (99% of the data). As this model has no interpretive value, we used the parameters of our second best/highest performing model to run our final topic model (table 1).

Figure B2: Plot of the coherence CV against the HDBSCAN minimum cluster size parameter across 157 model sweeps.

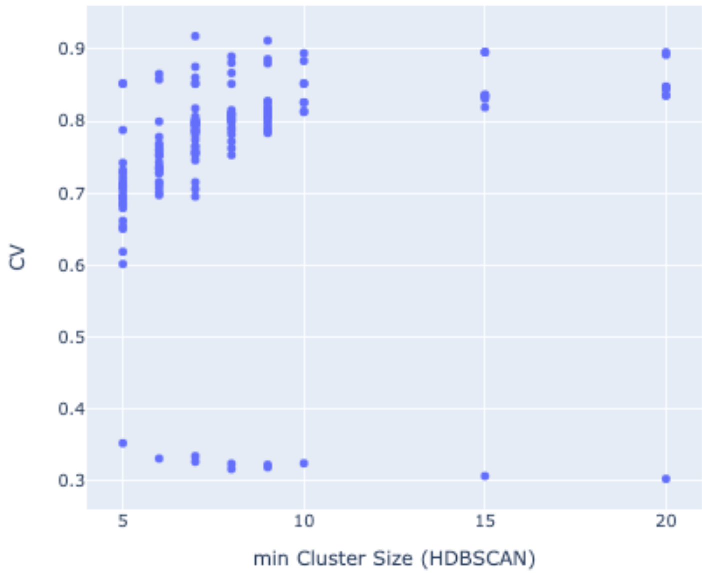


Table B1: Best performing model hyper-parameters selected for the final topic model (C_V=0.91, number of topics=25, number of outliers=519 (43%))

Hyper-parameter	Value
UMAP n neighbors	10
UMAP n components	4
UMAP minimum distance	0.00
Count Vectorizer ngram range	1-3
HDBSCAN minimum cluster size	9

References

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D., Wu, J., Winter, C., … Amodei, D. (2020). Language models are few-shot learners. *Advances in Neural Information Processing Systems*, *33*, 1877–1901. https://proceedings.neurips.cc/paper/2020/hash/1457c0d6bfcb4967418bfb8ac142f64a-Abstract.html

Grootendorst, M. (2021). *KeyBERT: Minimal keyword extraction with BERT.* [Computer software]. Zenodo. https://doi.org/10.5281/zenodo.4461265

Grootendorst, M. (2022). *BERTopic: Neural topic modeling with a class-based TF-IDF procedure* (arXiv:2203.05794). arXiv. https://doi.org/10.48550/arXiv.2203.05794

Laureate, C. D. P., Buntine, W., & Linger, H. (2023). A systematic review of the use of topic models for short text social media analysis. *Artificial Intelligence Review*. https://doi.org/10.1007/s10462-023-10471-x

McInnes, L., Healy, J., & Astels, S. (2017). hdbscan: Hierarchical density based clustering. *Journal of Open Source Software*, *2*(11), 205. https://doi.org/10.21105/joss.00205

McInnes, L., Healy, J., & Melville, J. (2020). *UMAP: Uniform manifold approximation and projection for dimension reduction* (arXiv:1802.03426). arXiv. https://doi.org/10.48550/arXiv.1802.03426

OpenAI. (2023). *ChatGPT* (22 August) [Python]. https://chat.openai.com

Reimers, N., & Gurevych, I. (2019). Sentence-BERT: Sentence embeddings using siamese BERT-networks. *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural*

Language Processing (EMNLP-IJCNLP), 3982–3992. https://doi.org/10.18653/v1/D19-1410

Röder, M., Both, A., & Hinneburg, A. (2015). Exploring the space of topic coherence measures. *Proceedings of the Eighth ACM International Conference on Web Search and Data Mining*, 399–408. https://doi.org/10.1145/2684822.2685324

Terragni, S., Fersini, E., Galuzzi, B. G., Tropeano, P., & Candelieri, A. (2021). OCTIS: Comparing and optimizing topic models is simple! *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*, 263–270. https://doi.org/10.18653/v1/2021.eacl-demos.31