RAGnaREC

Pranav Setlur, Calvin Qin, Rosy Chen, and Ayush Kumar
School of Computer Science, Carnegie Mellon University
Pittsburgh, USA
{psetlur, calvinq, dingdinc, ayushkum}@andrew.cmu.edu

1 Introduction

This project develops a recommendation system that applies sentiment analysis and natural language processing (NLP) techniques to provide personalized suggestions based on user input. It extracts insights from unstructured text in user reviews to generate recommendations aligned with individual preferences and sentiments.

This system employs a multi-stage pipeline that includes clustering, sentiment analysis, and Retrieval-Augmented Generation (RAG). Reviews are clustered into categories to group similar content, followed by sentiment analysis to gauge the tone and intent of each review. The RAG component retrieves relevant reviews and integrates them with input queries to generate contextually appropriate responses. This ensures balanced recommendations reflecting both positive and negative sentiments.

In today's era of information overload, personalized recommendations are essential. Our system moves beyond popularity-based suggestions, capturing nuanced options to deliver relevant results quickly. For businesses, it provides insights into customer preferences and pain points, enhancing service and satisfaction. Our project showcases the potential of combining multiple NLP techniques to create more intelligent and user-centric information retrieval systems. It enhances personalized recommendation systems and advances our understanding of how complex NLP techniques can better serve user needs.

2 Data Summary

We use the Yelp Reviews dataset, featuring user reviews on topics like restaurants, services, and healthcare. Each review includes a star rating initially ranging from 1 to 5 but normalized to a 0-4 scale. This dataset's diversity across topics makes it ideal for training a system to categorize and recommend items across domains.

However, the dataset contains a few biases. As a subset of Yelp's full data, this dataset may under-represent certain regions or business types. Ratings are also skewed towards higher values, which could impact the model's ability to capture negative sentiments and recommend items to avoid. Finally, some fake reviews might persist despite Yelp's mitigation efforts, potentially influencing the overall sentiment and recommendations.

3 Methodology

Our system uses a multi-stage pipeline that integrates various NLP and machine learning techniques. The pipeline consists of three primary components: clustering, sentiment analysis, and RAG.

3.1 Preprocessing

Our system preprocesses reviews to standardize and reduce the dimensionality of the dataset. We use Named Entity Recognition (NER) [4] to filter reviews by identifying entities relevant to recom-

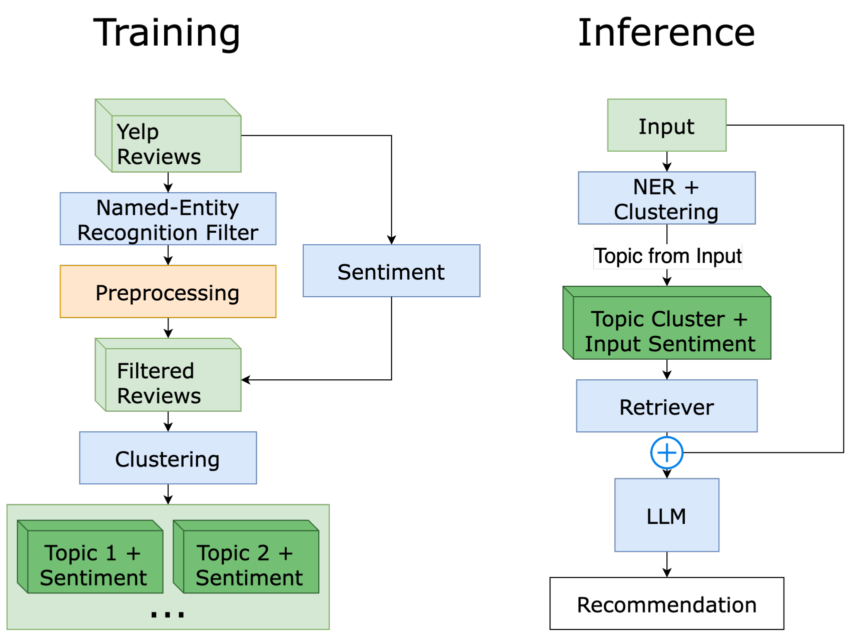

Figure 1: Entire System Diagram for Training and Inference

mendation. After isolating a subset of 'useful' reviews, we further experimented with text-based preprocessing techniques.

Named Entity Recognition

Named Entity Recognition (NER) identifies entities within each Yelp review that correspond to recommendable items. Reviews that do not include such entities are removed. Regardless of uniqueness or detail, a review must contain something recommendable to be considered useful.

We first tried using the Natural Language ToolKit (NLTK) to perform part of speech (POS) tagging. NTLK produces POS tags from the Penn Treebank Project, and we planned to filter for proper nouns (NNP and NNPS). This method identified reviews that contain named entities, but also produced poor results (2). Therefore, we experimented with spaCy [7] to include the semantics and context of each word. spaCy provides pretrained models for NER that identify entities from a pre-trained list (3). We are interested in tags that represent recommendable entities, such as `ORG`, `FAC`, and `PERSON`. Experimentation (3) proved that the best option is to tag `FAC`, `ORG`, `PERSON`, `GPE`, `EVENT`, `WORK_OF_ART`, `PRODUCT`, `MONEY`, and `PERCENT`.

Basic Preprocessing

We then run NLP preprocessing on the resulting tagged reviews, including removing punctuation, tokenization, lemmatization, and filtering out stopwords/verbs/single character tokens. These steps help reduce the size of our review vocabulary while also standardizes the words used in each review. We took special care to preprocess each review while preserving the special marked tokens, like `<person>` or `<fac>`. We therefore use regex matching to identify words and bracketed tokens.

3.2 Topic Modeling

We implemented domain-specific clustering to categorize reviews and streamline downstream tasks. After evaluating several topic modeling approaches like Latent Dirichlet Allocation (LDA) [2] and its guided variant [9], we selected BERTopic [5] as our primary clustering method.

BERTopic leverages sentence transformers and c-TF-IDF to create dense, interpretable clusters while preserving important words in topic descriptions. We explored guided topic modeling with seed topics, and seed word weighting. The guided topic modeling takes a list of seed topics, each comprising representative keywords. The seed topics need to be carefully crafted since if they do not exist or are too general, the model ignores them. The seed word variation allows users to identify and up-weight domain-specific keywords in topic representations. This approach minimized risks associated with incorrect keyword selection by preserving the low importance of irrelevant words.

After extensive evaluation, guided BERTopic delivered the most contextually relevant cluster representations. The model's effectiveness stems from its ability to capture semantic relationships between reviews, moving beyond simple keyword-based clustering. The guided topic modeling demonstrated notably better performance than the seed word approach, likely because single words prove insufficient in capturing the nuanced complexity of topics. Another advantage is that the model provides cluster assignment probabilities during inference, offering a measure of classification confidence.

Our methodology for identifying effective seed topics followed a two-step process: initial clustering without guidance to generate candidate topics, followed by iterative refinement of representative words. However, we encountered a significant challenge: despite extensive hyperparameter tuning of `min_cluster_size`, `num_topics`, and seed topic lists, approximately 40% of the dataset remained unclassifiable. We developed mitigation strategies for this issue during the inference phase, which we discuss in detail in our results section.

3.3 Sentiment Analysis Model

After clustering the reviews, we used a supervised sentiment analysis pipeline to classify the user reviews as positive, negative, or neutral. We did this to capture review *alignment*, i.e. does user A's review convey an overall positive experience? This information allowed for more informed recommendation, particularly in aiding our RAG generator for "also try" or "also avoid" recommendations. We explored various sentiment modelling approaches, each representing different paradigms. We found that explicitly providing the LLM with review sentiment boosts the specificity of generated recommendations while better aligning it with the user's intent.

Initially, we experimented with rule-based sentiment classifiers. VADER [8] uses lexical analysis along with generalizable grammar and syntactical conventions to predict the sentiment of text. To explore more complex patterns in the review text, we implemented a Long Short-Term Memory (LSTM) network [6]. LSTMs, a type of recurrent neural network (RNN), are effective in capturing long-range dependencies in sequential data, making them well-suited for sentiment analysis tasks in longer review texts. Finally, we evaluated several state-of-the-art transformer models, including DistilBERT [14], RoBERTa [11], and FinBERT [1]. These pre-trained models were fine-tuned on our preprocessed dataset to adapt to the specific language and sentiment patterns present in Yelp reviews.

Among these models, DistilBERT performed the best, achieving the highest accuracy. Our final sentiment analysis pipeline outputs both a sentiment label and a confidence score for each review. The confidence score played a critical role in aligning user intent, helping to distinguish between "also try" and "avoid" lists in our recommendation system.

3.4 Retrieval-Augmented Generation (RAG)

Extending the traditional RAG model [10], which retrieves relevant documents from a large corpus and concatenates them with the input query to generate responses, we developed an enhanced RAG pipeline for personalized recommendations, focusing on efficient query processing, review retrieval, and response generation. We utilized a VectorStoreRetriever to identity relevant reviews within the most appropriate cluster. We then conducted extensive experiments with distilGPT [14], Gemma 2-2B [12], Llama 3.2-1B [3], and Bloomz-560M [13] for the generation phase. We tested for fluency, relevance, and coherence of the generated recommendations. We selected Gemma as our primary generation model because of its superior performance in producing contextually appropriate and fluent responses.

Third, we developed an iterative prompt engineering process to optimize output quality. This step involved testing numerous prompt templates and refining them based on output analysis. Our final prompt template was designed to minimize hallucinations and maximize recommendation coherence. For example, we explicitly guided the model to highlight relevant points from similar reviews and suggest actionable recommendations.

The generation component takes a user review and the retrieved relevant reviews as input. It then process this information using Gemma to produce a generated recommendation text. Throughout the development of our RAG pipeline, we encountered several challenges. Initially, we attempted a single-step retrieval over the entire corpus. However, this was computationally expensive and often retrieved irrelevant reviews. Instead, we developed the clustered retrieval approach. Additionally, our early experiments resulted in occasional hallucinations and inconsistent recommendations, which motivated our iterative prompt engineering process. These mechanisms allowed our RAG pipeline to provide more context-aware and efficient recommendations compared to the traditional RAG model.

4 Results

We adopted a component-wise evaluation strategy, testing each module independently before integrating them and assessing the system as a whole. This approach allowed us to isolate and optimize individual components while ensuring they worked well together.

4.1 Clustering

Through comprehensive evaluation of multiple methodologies, we selected Guided BERTopic as our primary approach. Training the model on our preprocessed dataset of approximately 400,000 reviews yielded 665 distinct clusters, including a special cluster (-1) designated for unclassifiable reviews. This unclassifiable category comprised roughly 40% of all reviews, a significant proportion we attribute to the nature of Yelp reviews, which often express personal sentiments without providing objective descriptions of the target.

To address this challenge, we implemented a novel solution for handling unclassified reviews during testing. If a review is assigned to the -1 cluster, we use a keyword matching algorithm that compares the review content with topic keywords across all clusters. The review is then assigned to the cluster containing the matching keyword with the highest probability, ensuring more comprehensive coverage of our dataset. Figure 2 shows the intertopic distance map of our final clusters.

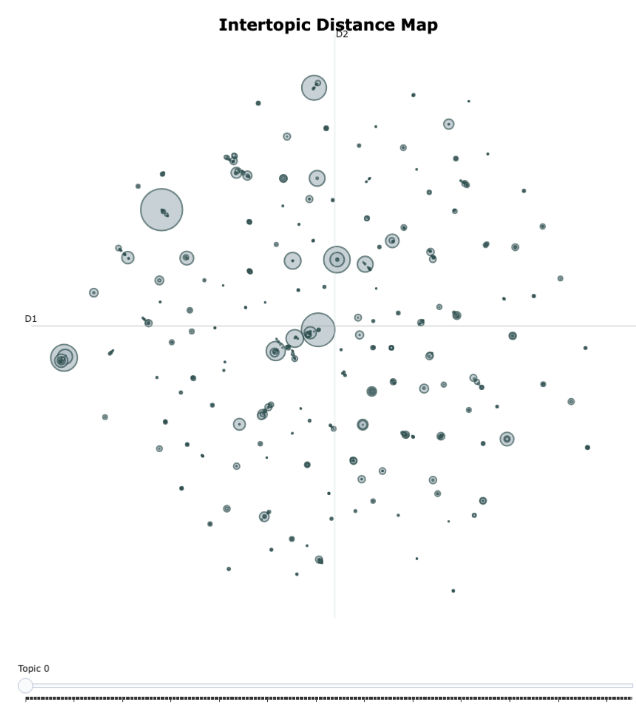

Figure 2: Intertopic Distance Map of the Final 665 Clusters

4.2 Sentiment Analysis

Table 1 presents a comprehensive comparison of the sentiment analysis models evaluated on our dataset. The results showcase the performance of each model across different sentiment categories, providing insights into their strengths and limitations. One notable observation was the inability of the LSTM model to generalize well on the validation dataset. Although providing strong interpretability, the LSTM's poor performance underscores the challenges traditional deep learning models face in handling the complexities of sentiment analysis in large, nuanced datasets.

Among the evaluated models, DistilBERT demonstrated the best overall performance, with a high confidence score and reliable predictions across both positive and negative sentiments. Its binary classification framework provided clear and consistent sentiment differentiation. The average labels associated with its predictions are closely aligned with the review ratings, suggesting that DistilBERT effectively captured the sentiment nuances in the dataset.

Although we proceeded with DistilBERT for our final implementation, it's worth noting that VADER could be a good alternative for applications requiring quick results with reasonable accuracy. Despite its bias towards positive sentiments, it is efficient and doesn't require extensive training or fine-tuning.

Sentiment Models Performance					
Model	**Model Type**	**Sentiment**	**Sentiment Frequency**	**Average Confidence**	**Average Label**
VADER	Rule-based	Positive	0.80	0.83	2.36
		Negative	0.19	0.63	0.79
		Neutral	0.01	0.99	1.46
LSTM*	RNN	Positive	0.36	0.88	3.12
		Negative	0.69	0.84	0.88
RoBERTa	Transformer	Positive	0.59	0.86	2.91
		Negative	0.33	0.76	0.68
		Neutral	0.08	0.47	1.48
DistilBERT*	Transformer	Positive	0.50	0.97	3.07
		Negative	0.50	0.97	1.03
FinBERT	Transformer	Positive	0.30	0.96	2.80
		Negative	0.10	0.91	0.93
		Neutral	0.70	0.94	1.89

Table 1: Performance comparison of sentiment analysis models.
**For better performance, DistilBERT and LSTM only categorize reviews as positive or negative.*

4.3 Retrieval-Augmented Generation (RAG)

The RAG pipeline was evaluated on the quality and efficiency of its generated recommendations. In the generation phase, we compared the performance of Gemma 2-2B, DistilGPT, Llama 3.2-1B, and Bloomz-560M on metrics like fluency, relevance, and coherence.

Gemma consistently produced the most fluent and contextually relevant recommendations. It effectively synthesized retrieved reviews and generated responses that aligned closely with the user query. However, these advantages came with the trade-off of higher computational overhead and longer generation times. DistilGPT, on the other hand, offered faster response times and required fewer computational resources. While its outputs were less sophisticated than those from Gemma, it maintained an acceptable level of fluency and relevance, especially for less complex queries. Bloomz, however, tended to hallucinate, particularly in cases where input queries contained a negative review with a positive intent (e.g., "I did not enjoy this experience, tell me places I might enjoy"). Meanwhile, we found that Llama often provided minimal elaboration, simply listing places of interest without explaining why they might appeal to a user.

We observed that improved prompting strategies and chain-of-thought reasoning enhanced the performance of simpler models like Bloomz and Llama, especially in zero-shot scenarios. Exploring techniques to bridge the gap between smaller, more efficient models and their larger counterparts remains an open area of research. Ultimately, while DistilGPT offers a good balance between efficiency and quality, we chose Gemma for our final implementation due to its superior performance.

4.4 Integrated System

To evaluate the system as a whole, we designed a comprehensive integrated system testing strategy. The process involved conducting end-to-end evaluation using a stratified set of user queries. We evaluated the recommendations based on the following qualitative methods.

- **Recommendation Relevance:** Measured by the overlap between recommended reviews and the user's query intent. All four generator models performed well on this metric.
- **Sentiment Appropriateness:** The alignment of recommended sentiment with the user's preferences, using sentiment analysis scores to quantify agreement between user input and system output. The models generally performed well on this metric as well.
- **Intent Recognition:** The system's inferred intent compared with the intent of the prompt. The smaller models performed poorly here.
- **Response Coherence:** The fluency, contextual alignment, and logical flow in response. This is something language models excel at, so the generators had no issue with this metric.

We then applied the metrics to four categories of queries:

1. Positive review seeking similar experiences: Testing the system's ability to generate recommendations that align with a positive sentiment and experience - our expected primary use case.
Example prompt: "I had a fantastic experience at Bright Smiles Dental. The staff was friendly, and the cleaning was thorough. The ambiance was great and fun, and I felt relaxed the entire time I was there."
Response:"Based on your review, it sounds like you had a fantastic experience at Bright Smiles Dental! It's great to hear you enjoyed the friendly staff and thorough cleaning. Here are a few other dental practices that have received positive feedback and might be a good fit for you:
Smile Dental Care: This practice has consistently received high ratings for its friendly staff and comfortable atmosphere. Many reviewers mention the dentists are gentle and take the time to explain procedures.
Aspen Dental: Aspen Dental is known for its affordable prices and convenient locations. Reviewers have praised the quality of care and the efficiency of their services. Let me know if you'd like more recommendations!"

2. Positive review wanting to avoid specific aspects: Evaluating the system's ability to filter out unwanted elements while providing relevant alternatives.
Example prompt: "SushiFuku's food itself tasted good, decently fresh fish. That said the rolls are made worse than some make-at-home kit and free all feel apart nearly instantly. Great value for the cost, as long as you can deal with the rolls falling apart, this place is a great, and quick meal for the cost. Are there other spots with better put-together rolls?
Response: "Based on your review, it sounds like you're looking for sushi places that offer good value and fresh fish, but with rolls that hold together better. Here are a few places you might enjoy:
Sushi Roku: This restaurant is known for its high-quality sushi and creative rolls. They have a great reputation for fresh ingredients and a good balance of flavor.
Kaito Sushi: This spot is a local favorite for its delicious sushi and friendly service. They have a wide variety of rolls and sashimi, and their prices are very reasonable.
Sushi House: This place is a bit more casual, but they have a great selection of sushi and sashimi, and their rolls are always fresh and flavorful.
Let me know if you'd like more recommendations!"

3. Negative review aiming to avoid similar experiences: Assessing the system's ability to identify and exclude negative recommendations effectively.
Example prompt: "The food at Bob's Diner was terrible, and the staff was rude. Can you tell me other places to avoid?"
Response: "Based on your review, it sounds like you're looking for places to avoid. Here are a few places that have received negative feedback:
Jack's Pub: The user mentioned that the drinks were bad, the food was lousy, and the karaoke was

horrible.
Yolie's: While some reviewers enjoyed Yolie's, others have mentioned cold cheese and rude service."

4. Negative review looking for better alternatives: Testing the system's ability to balance user concerns with constructive, relevant recommendations.
Example prompt: "The bakery on Fifth Avenue had stale pastries and no seating. Their muffins were dry on the inside! Any good bakeries where I can enjoy fresh treats comfortably?"
Response: "Based on your review, it sounds like you're looking for a bakery that offers fresh pastries and comfortable seating. Here are a few places that might be a good fit:
German Bread Bakery: This bakery has consistently received positive reviews for its fresh pastries and welcoming atmosphere. They are known for their delicious apple strudel and other German-inspired treats.
Bakeshop: This bakery is known for its high-quality ingredients and delicious baked goods. They have a cozy and inviting atmosphere, and the staff is friendly and knowledgeable.
Carlo's Bakery: This bakery is a New York institution known for its authentic New York-style baked goods. They have a long line, but it's worth the wait for their delicious pastries."

This stratified approach allowed us to assess the system's ability to handle nuanced user intents and generate appropriate recommendations across different scenarios. By analyzing the results, we compared the system's performance with the expectations set by individual component testing. The results showed that the RAG pipeline was effective in providing contextually relevant and coherent recommendations across all categories.

5 Discussion and Analysis

Our multi-stage pipeline, integrating clustering, sentiment analysis, and RAG, demonstrated promising results in generating personalized recommendations based on user reviews. This method proved appropriate for our task of understanding user intent and providing context-aware suggestions. However, as with any complex system, there are areas where improvements can be made, and insights have been gained regarding the limitations and future directions.

5.1 Appropriateness of Methodology

Our pipeline's modular design integrates multiple components, each selected to enhance system performance and contribute to the final output. The preprocessing step prioritized precision over computational efficiency, standardizing and cleaning reviews to improving clustering performance and reduce errors in subsequent steps. The clustering effectively grouped semantically similar reviews, reducing computational overhead and enhancing retrieval accuracy. We chose Guided BERTopic as our clustering methodology for its comprehensive capabilities: it leverages semantic similarity for grouping, accepts keyword lists to guide topic formation, and provides confidence scores during inference. Furthermore, BERTopic prioritizes precision over exhaustive classification. By allowing some reviews to remain unclustered, the resulting output contains higher-quality, more semantically cohesive topic clusters, enhancing performance in downstream tasks.

The choice of DistilBERT for sentiment analysis proved effective in capturing contextual sentiment nuances, particularly for classifying reviews into sentiment categories that aligned well with user intent. Similarly, the RAG pipeline produced context-aware and coherent responses, while our modifications reduced hallucinations and enhanced output quality. The iterative prompt engineering we employed in the RAG pipeline played a critical role in mitigating errors and improving coherence. By fine-tuning the system's prompts, we ensured that recommendations were both relevant and actionable. This approach optimized performance while also highlighting the importance of human oversight in developing robust AI systems.

5.2 Limitations and Constraints

Despite its strengths, our approach has notable limitations that warrant discussion.

Generator Model Constraints: While effective, Gemma lacks the advanced capabilities of state-of-the-art models like GPT-4 or Claude-3.5 which could potentially enhance coherence, fluency, and context awareness in generated recommendations. However, these alternatives would increase computational costs.

Subjectivity in BERTopic: Manual curation of seed topics introduces subjectivity and limits scalability, particularly in dynamic environments.

Computational Intensity: Despite clustering optimizations, the multi-stage pipeline remains computationally intensive, challenging real-time applications in resource-constrained settings.

5.3 Insights and Future Improvements

The results of our study provide valuable insights into the strengths and weaknesses of multi-stage NLP pipelines. While the modular design proved effective for our task, it also exposed the complexities and interdependencies of each stage. For instance, the accuracy of clustering directly influenced retrieval quality, which in turn affected the coherence of generated recommendations.

To address these limitations, future work could incorporate state-of-the-art models for all stages of our system. Fine-tuning spaCy's NER models could provide more nuanced entity tags, such as `RESTAURANT` or `OFFICE` as opposed to the generic `ORG` tag. Moreover, better embedding or clustering models could reduce our unclustered set, improving data efficiency and diversity. Advanced generator models like GPT-4 could offer significant improvements in handling nuanced queries. Furthermore, employing self-supervised learning methods in the clustering mechanism could reduce the need for manual intervention in topic modeling.

Optimizing the computational efficiency of the pipeline is another critical area for improvement. Techniques such as model distillation, pruning, or quantization could be applied to reduce resource consumption while maintaining performance. Additionally, implementing caching mechanisms for frequently accessed clusters or precomputed embeddings could further improve efficiency, but this comes with the trade off of a higher memory overhead.

6 Conclusion

Our work contributes to the broader effort of making AI systems more intuitive and responsive to human needs. Our goal was to design and implement a multi-stage pipeline integrating clustering, sentiment analysis, and Retrieval-Augmented Generation (RAG) to generate personalized recommendations from user reviews. Through a structured approach, we demonstrated the pipeline's ability to interpret nuanced user intents and provide contextually relevant, actionable suggestions. Our methodology, combining state-of-the-art techniques in NLP and sentiment analysis, proved effective in addressing the core challenges of this task.

The results showed that the modular design of our pipeline, particularly the integration of clustered retrieval with RAG, significantly improved relevance and coherence while reducing computational complexity. The sentiment analysis step using DistilBERT was great at capturing the subtleties of reviews, and the iterative prompt engineering improved the overall fluency of our recommendations. These results validated the feasibility of our approach for personalized recommendation systems in real-world applications. Nevertheless, we highlighted several areas for improvement. The choice of generator model, reliance on manually guided clustering, and computational intensity pose notable challenges.

All in all, our work demonstrates the potential of integrating advanced NLP techniques into a cohesive framework for recommendation systems. By addressing the identified limitations and expanding the scope of the pipeline, future iterations can achieve even greater levels of adaptability and scalability.

GitHub Repository

References

[1] Dogu Araci. Finbert: Financial sentiment analysis with pre-trained language models. *CoRR*, abs/1908.10063, 2019.

[2] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3(null):993–1022, March 2003.

[3] Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, and Ahmad Al-Dahle et. al. The llama 3 herd of models, 2024.

[4] Ralph Grishman and Beth Sundheim. Message Understanding Conference- 6: A brief history. In *COLING 1996 Volume 1: The 16th International Conference on Computational Linguistics*, 1996.

[5] Maarten Grootendorst. Bertopic: Neural topic modeling with a class-based tf-idf procedure. *arXiv preprint arXiv:2203.05794*, 2022.

[6] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 11 1997.

[7] Matthew Honnibal and Ines Montani. spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing. To appear, 2017.

[8] C. Hutto and Eric Gilbert. Vader: A parsimonious rule-based model for sentiment analysis of social media text. *Proceedings of the International AAAI Conference on Web and Social Media*, 8(1):216–225, May 2014.

[9] Jagadeesh Jagarlamudi, Hal Daumé III, and Raghavendra Udupa. Incorporating lexical priors into topic models. In Walter Daelemans, editor, *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 204–213, Avignon, France, April 2012. Association for Computational Linguistics.

[10] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive nlp tasks. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 9459–9474. Curran Associates, Inc., 2020.

[11] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692, 2019.

[12] Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, Pouya Tafti, Léonard Hussenot, Aakanksha Chowdhery, Adam Roberts, Aditya Barua, Alex Botev, Alex Castro-Ros, Ambrose Slone, Amélie Héliou, Andrea Tacchetti, Anna Bulanova, Antonia Paterson, Beth Tsai, Bobak Shahriari, Charline Le Lan, Christopher A. Choquette-Choo, Clément Crepy, Daniel Cer, Daphne Ippolito, David Reid, Elena Buchatskaya, Eric Ni, Eric Noland, Geng Yan, George Tucker, George-Cristian Muraru, Grigory Rozhdestvenskiy, Henryk Michalewski, Ian Tenney, Ivan Grishchenko, Jacob Austin, James Keeling, Jane Labanowski, Jean-Baptiste Lespiau, Jeff Stanway, Jenny Brennan, Jeremy Chen, Johan Ferret, Justin Chiu, and et al. Gemma: Open models based on gemini research and technology. *CoRR*, abs/2403.08295, 2024.

[13] Niklas Muennighoff, Thomas Wang, Lintang Sutawika, Adam Roberts, Stella Biderman, Teven Le Scao, M Saiful Bari, Sheng Shen, Zheng-Xin Yong, Hailey Schoelkopf, Xiangru Tang, Dragomir Radev, Alham Fikri Aji, Khalid Almubarak, Samuel Albanie, Zaid Alyafeai, Albert Webson, Edward Raff, and Colin Raffel. Crosslingual generalization through multitask finetuning, 2023.

[14] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. In *NeurIPS EMC*$^2 Workshop,$ 2019.

Appendix

A Example NLTK Tagging Outputs

Examples of Useless Reviews that are Filtered Out	- found a bug in my rice... - Incredibly rude employees. Will never be back. - They do not respect our flag!!! A business should have more respect for our country and address issues like this.
Examples of Useless Reviews that are Kept	Last minute **Halloween** costume shopping, they didn't have any of the costumes I wanted. **Wario** was nowhere to be found, unfortunately. Service sucked. The girl working there had moved like a snail and clearly didn't want to be there.
Examples of Good Reviews that are Kept	OK, so I admit that I was underwhelmed by **Cafe Zinho** when I first went, a few years ago... In short, I owe **Cafe Zinho** an apology.

Table 2: While the Halloween review was not filtered out because the bolded nouns were detected, this review still does not provide anything we can recommend. Therefore we would like to avoid this review in our filtered output.

B Tagging Performance Experimentation

Baseline Clustering Tagging Performance					
Tagging Method	Inlcuded Tags	Inter-Cluster Distance	Intra-Cluster Distance	Silhouette Coefficient	Unclustered Percentage
Baseline	FAC, ORG, PERSON	0.685	**0.921**	0.022	45.9%
Numeric	*Baseline* + MONEY, PERCENT	0.697	0.923	**0.030**	45.7%
Verbose	*Baseline* + GPE, EVENT WORK_OF_ART, PRODUCT	**0.706**	0.922	0.019	**44.1%**
Numeric + Verbose	*Numeric* + *Verbose*	0.690	0.935	0.027	45.2%
VAD	*Baseline* + Removed Verbs	0.673	0.940	0.025	45.5%
VAD + Numeric	*VAD* + *Numeric*	0.690	0.934	0.028	45.6%

Table 3: Performance comparison of various tagging/preprocessing methods.
**Results found by averaging 50 randomized unguided clusterings.*

```
CARDINAL: Numerals that do not fall under another type
DATE: Absolute or relative dates or periods
EVENT: Named hurricanes, battles, wars, sports events, etc.
FAC: Buildings, airports, highways, bridges, etc.
GPE: Countries, cities, states
LANGUAGE: Any named language
LAW: Named documents made into laws.
LOC: Non-GPE locations, mountain ranges, bodies of water
MONEY: Monetary values, including unit
NORP: Nationalities or religious or political groups
ORDINAL: "first", "second", etc.
ORG: Companies, agencies, institutions, etc.
PERCENT: Percentage, including "%"
PERSON: People, including fictional
PRODUCT: Objects, vehicles, foods, etc. (not services)
QUANTITY: Measurements, as of weight or distance
TIME: Times smaller than a day
WORK_OF_ART: Titles of books, songs, etc.
```

Figure 3: List of Tags and Descriptions provided by spaCy