What is on People's Minds?

Alexei Chen and **Zedong Chen** and **Tian Xia** and **Harry Yang** and **Qisi Yang**
University of Michigan

1 Project Description

1.1 Current Problem

Selling books on Amazon is a venture filled with potential gains and risks. On the one hand, a significant portion of the U.S. adult population (about 65%) prefers print books and Amazon takes at least 40% of the print book market in the USA, selling millions of books annually (McLoughlin, 2023). This presents a lucrative opportunity to sell books and gain profit from this vast market. On the other hand, being an Amazon bookseller also involves risks such as investing in unpopular titles or missing out on the sudden surges in demand for bestsellers. The key to success is aligning with market demands, and constantly monitoring the trends in books that people love and are interested in, understanding what titles consumers are most eager to read at any given moment.

However, there are too many ratings and reviews coming from various platforms every day which will influence a potential reader's decision-making. It is a big challenge for Amazon booksellers to get the most updated trends from so many platforms and understand current customer needs. Our project aims to equip booksellers with advanced analytical tools by recommending them with possible future bestsellers. We will first aggregate and analyze data from diverse sources such as Goodreads, social media commentary, and book reviews. We then want to prioritize books with emerging trends, indicated by the prevalence of specific keywords to make a dynamic and reliable recommendation system. Sellers could get advice from our model to stock books that are likely to be in high demand and help themselves optimize their inventory and maximize potential profits.

1.2 Proposed Solution

In order to provide better recommendations for book categories, it is necessary to propose a classification system that reflects current trends to better categorize books. We believe that analyzing current trends can help us understand what people are thinking about and what they are focusing on at a specific time. Therefore, we not only utilize datasets related to books but also gather data from social media through web crawling. We attempt to extract the categories we need from the summaries of these books and the Wikipedia entries of trending words on the internet. Book summaries and word synopses often contain words indicative of broader categories related to books and words, which could be a good potential way to extract trends.

We then preprocess these texts. Firstly, we preprocess the text by removing some common words and stop words, retaining those words that may represent categories, and only keeping some of the more frequently occurring words of this kind as our provisional categories. Then, we utilize BART for zero-shot text classification, using the summaries as text and the provisional categories as our possible classes. We eliminate the 50 categories with the lowest total probabilities for all summaries every round. Through iterations, we obtain the final 10 categories we need (ten most popular words for every two year), which are not only related to our dataset of books but also to some extent reflect contemporary trends, providing a feasible platform for recommending book categories.

Currently, through text classification of book summaries, we identify the most popular categories for every two years for those books and utilize the RoBERTa model to predict popular book categories for next year. We will use the most 10 popular book

categories from the previous year as inputs, along with directives and guiding words as prompts, to recommend potential popular categories for the future. Moreover, we use 1-shot example to help the model to predict what we want. Thus, we can recommend future trends and preferred directions based on past data, providing a solution for recommending books that align with current trends. Furthermore, we could use the predicted most popular book category to find suitable books via the Google API, which can further recommend a specific book to user.

2 Related Work

As the world becomes increasingly connected over the Internet, more and more information is being shared on various social media platforms. Websites like X/Twitter provide a place for people to share their reactions to viral trends as they happen, which captures the broader topics that society cares about at a given time and can be used to provide recommendations for media like books.

Althoff et al. (2013) approached the task of trend analysis by crawling Google, Twitter, and Wikipedia to retrieve popular terms and articles for each day in a year-long period, which was then clustered by a Levenshtein edit distance threshold to consolidate similar terms into their overarching topic. They assigned scores for each topic based on its rank each day, which allowed the detection of the overall most trending topics for September 2011-September 2012, as well as individual time series for specific terms like “Neil Armstrong”. In doing so, they found that different sources tend to specialize in different topics, so analyzing only one set of data may skew one’s impression of which topics are most trendy; for instance, Twitter contained the most trends related to products and technology, while Wikipedia had many trending topics on natural disasters. Therefore, social media sites like Twitter would be most useful for us to gain information on product-related trends.

Recently, Ding et al. (2023) explored various methods of modeling trend recommendations, aiming to identify items that may become popular in the near future, rather than only promoting items that are already popular. They describe "TrendRec", a neural network model that contains two components: a sequential recommender and a time series forecasting model, which can be integrated to decide which item should be recommended next and when. They use TaoBao behaviors, Netflix recommendations, and Microsoft News data to train their model. However, we feel that it would be more informative to combine reviews (in our case, GoodReads reviews) with social media trends, since the instantaneous nature of social media better reflects how fast trends can change.

One challenging aspect of our project is how to recommend (i.e. predict) what trends may become popular. A potentially useful model is RoBERTa (Liu et al., 2019), which is based on the BERT machine learning model. RoBERTa is trained with masked language modeling, which involves the prediction of a randomly masked word during the training. We aim to repurpose this training for the prediction of trends, rather than the prediction of the next word in a sentence. To improve the prediction, we utilize the concept of "in-context learning". This is a technique in which a few training examples are provided in the prompt, which also contains a test instance that the model is supposed to respond to (Brown et al., 2020). Unlike fine-tuning, in-context learning does not update the weights of the model, which means that it is comparatively less resource-intensive. However, in-context learning can be a delicate process. (Liu et al., 2021) showed that for a GPT-3-based model, factors ranging from the number of in-context examples to the order in which they were provided all had an effect on the model’s performance, demonstrating the importance of prompt design when using large language models.

Another method that was shown to improve performance is to use expert prompting, which asks the large language model to act like a highly knowledgeable expert in a field related to the task at hand. By doing so, (Xu et al., 2023) were able to outperform other chat assistants like Vicuna and Alpaca, indicating that including a statement about being an expert in the prompt is a simple yet effective way to improve accuracy. Thus, we decided to utilize this approach and combine it with the previously described approach of in-context learning, both of which should lead to improved performance.

3 Data Collection and Annotation

We collect our data from two main sources: Twitter trending topics and books from the past 15 years.

3.1 Social Media Trends Dataset

Data Collection Approach: We utilized a web crawling technique to gather data on prevailing social media trends. Initially, we considered using Twitter's API for direct scraping. However, due to the API's limitations on accessing historical data and rate limits, we opted for an alternative approach.

Source: We collected data from the Trend Calendar website (https://us.trend-calendar.com/), which archives daily top trending topics on Twitter. This allowed us to access and analyze past trends for specific dates. The URLs for the website are deterministic, such as https://us.trend-calendar.com/trend/2022-01-01.html, where only the date portion of the URL changes depending on the date.

Data Processing: Using the Python library Beautiful Soup, we programmatically navigated and parsed the website to extract the top ten trending topics for a given range of dates.

Additional Data Retrieval: For a better understanding of each trending topic, we queried Wikipedia's MediaWiki API to fetch summaries of the topics derived from the trends. In particular, the summary function, such as mediawikiapi.summary("Github"), provides a summary of the given string, assuming that the string can be found on Wikipedia.

Challenges: A notable challenge was parsing Twitter trends, especially when they appeared in hashtag form (e.g., "#dejoyhearing"). Hashtags often merged multiple words into a single string without clear delimiters, complicating the extraction of searchable terms. Methods like splitting by capitalization were considered but proved ineffective for acronyms (e.g., "#STP500"). Consequently, we decided to exclude trends consisting of densely connected words without spaces, as these also typically lacked corresponding articles on Wikipedia.

3.2 Book Summaries And Reviews Dataset

Data collection: Due to the lack of direct API access to Goodreads for extracting book descriptions, we resorted to manual data scraping using Selenium, which is an effective tool for automating web browsers. Additionally, we discovered that the Google Books API could be utilized to obtain book summaries, allowing us to compile two distinct versions of the summaries.

Challenges: The GoodReads website frequently displayed instability issues, such as frequent errors, loading problems, and occasional browser crashes. To mitigate these issues and ensure successful data retrieval, our script was designed with a '$max\_attempts$' limit. This retry mechanism allowed the script to make multiple attempts to connect and scrape data until successful or until the attempts exceeded the predefined limit.

3.3 Data Annotation

For data annotation, we manually observe all the categories produced from scraping the different websites after preprocessing the data. Given that we limit the number of categories, this allows for manual checking and provides us with better adjustments to our algorithm. Since the categories come from the highest score words from summaries or introductions, there is the possibility of irrelevant or meaningless categories that our algorithm produces. Therefore, we manually adjust and rerun our category generation algorithm until the categories provided are meaningful and provide useful distinctions between other categories.

4 Method Description

Our objective is to identify significant trends for each year, drawing from both books and social media sources. Our model is designed to summarize and identify trending categories every two years. We input Google Books summaries along with an initial set of 210 categories to our model. Below is the detailed methodology we follow:

4.1 Label Extraction

Data Preparation TF-IDF Calculation: We establish our categories using a specific computation—the weighted term frequency (TF) multiplied by the inverse document frequency (IDF). This calculation is applied to three different sources of text: Google Books summaries, Goodreads summaries, and Wikipedia descriptions of Twitter trending words.

Data Cleaning To enhance the quality of our analysis, we clean the data by removing

noise/uninformative elements. This involves using from nltk.corpus import stopwords to filter out non-essential words such as names, common pronouns, and prepositions. This step is crucial to focus on the most meaningful words in our summaries.

Part-of-Speech Tagging Identifying Word Types: We utilize spaCy with the en_core_web_sm model to perform part-of-speech tagging. Each word is tagged to identify whether it is a noun, verb, or adjective. This helps us understand the grammatical structure and emphasize nouns in our analysis, as they tend to carry more informational weight.

Weighting Words Term Frequency Adjustment: In our TF calculations, we focus exclusively on nouns due to their higher informational value in determining content relevance and category significance.

Category Synthesis Combining Sources: Finally, we synthesize and combine categories derived from the three different texts into our initial set of categories.

4.2 Data Processing Approach

We use a zero-shot text classification model called "bart-large-mnli" from Hugging Face (https://huggingface.co/facebook/bart-large-mnli) for our project. This type of model categorizes texts into specific classes based on a given set of labels, allowing us to match texts with relevant categories based on their content. Specifically, we classify book summaries against a set of chosen keywords or phrases to gauge their relevance, which from both book dataset and social media dataset. Those labels are from Google Books descriptions, Goodreads book summaries and Twitter trending words with Wikipedia descriptions (See table 1 below)

Here we present a brand new algorithm to find which categories should be used: Given N texts and X classes, we apply a zero-shot text classification model, "bart-large-mnli", to each of the N texts to ascertain their relevance to each category. The (X-50) most relevant classes will then be selected as the new classes. After that, repeat the reduction process until only 10 classes remain from 210 classes initially (Figure 1).

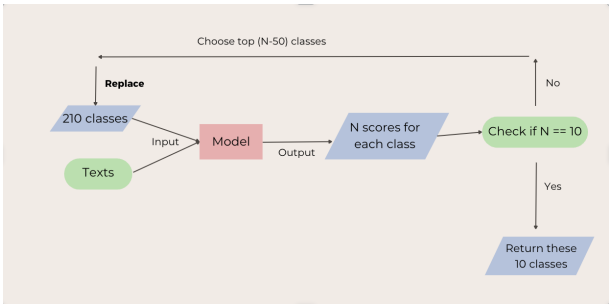

Figure 1: Flowchart of the algorithm

To predict future trends, we will employ a fill-mask model called RoBERTa (https://huggingface.co/FacebookAI/roberta-base), designed to predict missing words in a sentence, allowing us to generate forecasts based on the categories identified. For example, if we provide a prompt like "trending prediction: category1, category2, <mask>.", the model will return an appropriate word which will complete this sentence, and make the sentence make sense at the same time. We use 1-shot example to help the model to better understand this task and expert instruction to improve the performance.

In this project, we face many difficulties but finally solved them. We used three data sources and they are Goodreads, Google API and Twitter posts. For Goodreads and Google API, we extracted book name and book summarization, and for Twitter, we needed post title and post content. We did not have a unified interface so we needed to write different functions to read data from txt or json files. To adopt multiple data sources, after cleaning the data, we unified the interfaces so that all of the data sources could be read in similar approaches for better reading and processing by converting them to JSON format.

Preparing for the classification, we put all candidate labels in a txt file to get ready for the filter. We realized that it is important to prepare the label file because some words like "many" or "name" could never be future labels. Originally, we assigned a weight of 1 for each term. After that, we adjusted the weights according to the syntax meaning and sentence structure. We finally added more weights to words that have more potential to become future trend labels (like nouns). When

Input	Source	Texts	Classes
	Google Books API, Goodreads, Twitter and Wikipedia	N book Google summaries	210 most important words from Goodreads summaries, Google summaries and Wikipedia description(applying techniques such as case folding, stop-word elimination, and text normalization)

Table 1: Input for model

we ran the classification model, we also faced the problem that the classification stage took hours to filter and return the result. We have limited calculation resources, and our approach was too slow even after switching to Google Colab's GPU, so we had to find an alternative approach to speed up the classification. On one hand, we divided the data sources into chunks every two years like 2009-2010, 2011-2012, and 2013-2014... Each team member took a piece of smaller data chunk size to run, and each person took less than 30 minutes to finish. On the other hand, we adjusted the parameters by turning on the cuda device and removing more labels for each iteration.

After we get future trend labels, we make predictions based on the result. We added a feature which recommends users books and visualizes books in UI. We adopted the Google Book API which could return the most relevant books given a specific category. For example, given a label/category like "flowers", the API could return information of 800 books about flowers in sorted order (starting from the most relevant). Users could decide how many books they want to get recommended. For example, they can choose 10 books for each future trend label. We filtered the JSON result and only displayed useful info like book name, author name, prices and so forth. Book info and pictures will be delivered to the frontend webpage.

5 Evaluation and Integration

Overall, we will have two methods for social media. We will assess the effectiveness of both methods independently, as well as their combined results. To integrate the two, we will select 5 categories from each method, merging them to form a final list of 10 categories. This integrated approach allows us to refine our analysis and select the most representative keywords for each year, based on a comprehensive review of both book and social media trends.

To assess the accuracy and reliability of our method, we initiate the evaluation process by comparing outcomes from a smaller dataset with results obtained through manual categorization of summaries. This comparative analysis enables us to refine our approach, ensuring that our automated categorization closely aligns with human judgment. This step is crucial for validating the effectiveness of our data processing techniques and making necessary adjustments before scaling up to larger datasets.

The results of our first method of evaluation can be seen in Table 2. We can see that, despite providing the bart-large-mnli model with various book summaries and categories, the final categories deduced were not similar to the categories deduced manually. For example, for the years 2017-2019, there was only one match, "attention", between the manually selected categories and the final categories produced by the model, although there are a couple other categories that have similar themes or meanings. For the summaries provided of books from 2015-2016, we have two matches, "children" and "kids". Thus, the effectiveness of the bart-large-mnli model can be seen to differ greatly when compared to the manual evaluation of book summaries.

In regards to our second method of evaluation, there are clear patterns with the categories that the RoBERTa model predicted, with "media" and "marketing" showing up in all five of the predictions, and "categories" or "category" showing up in four of the five predictions (Table 3). Since we also have the calculated categories of those previous years, we can see that the actual categories for these two year intervals provides us with a more informative

Table 2: Comparison of Manual and Model-Based Categorizations for Randomly Selected Books

Year Range	Manual Categorization	Model Final Categorization
2017-2019	business, life, house, world, **attention**, secrets, decisions, control, power, women	classic, success, number, record, fans, championship, media, **attention**, leader, old
2015-2016	cooking, **children**, family, meals, attention, health, comedy, house, **kids**, food	advice, movement, groups, nomination, young, guide, company**children**, **kids**, program
2011-2012	adult, murder, president, campaign, history, power, death, creatures, world, **tale**	young, movement, company, position, **tale**, work, passion, action, friend, brand

Table 3: Predicted and Actual Categories from Models RoBERTa and BART

Years	RoBERTa Predictions	BART Calculations
2009-2010	-	thinking, tale, host, sequel, attention, power
2011-2012	media, category, marketing, categories, news	young, power, attention, thinking, tale, drama
2013-2014	**media**, category, marketing, categories, trends	**media**, power, young, host, company, variety
2015-2016	**media**, content, category, categories, marketing	**media**, classic, power, attention, groups, success
2017-2019	**media**, content, marketing, data, media	**media**, power, areas, classic, thinking, attention
2020-2021	media, marketing, style, business, category	-

idea of the topics of those years, as opposed to the topics that the RoBERTa model predicted.

6 Conclusions

In our approach, we ended up testing various datasets and used different models to provide us with a recommendation system that attempts to provide more recent books given the trends of books predicted. We began with crawling data from multiple sources, including a Twitter trends website, Goodreads, and Google. We also utilized pre-existing datasets to provide us with a corpus of books in which we used for our methods. After gathering all the data, we would preprocess and clean the data, then utilize tf-idf in order to establish our initial batch of categories. With these categories, we would provide them, along with book summaries of books with two year intervals, into the bart-large-mnli model to eventually provide us with the ten most impactful labels. These labels are then put into the RoBERTa model, providing us with predictions of trends of later years.

6.1 Limitations

Although we performed well on this task, we encountered certain limitations. One such limitation pertains to the reliability of employing in-context learning for future prediction. The model relies solely on implicit knowledge within its parameters rather than external data for prediction, which may introduce uncertainty. However, more data available, such as the top 10 most popular categories for each month, could enable us to fine-tune the model. This refinement process would facilitate the model in learning from a more comprehensive set of data.

Another limitation encountered was the amount of time it took to perform certain tasks. For example, web scraping was a task that ended up consuming many hours at a time, and there would always be new issues that would show up. This was resolved with using try statements, but still required multiple attempts to properly catch the various types of exceptions that came from web crawling. Furthermore, the speed and computational power at which the bart-large-mnli model would take to get the weights of the various categories was a large issue that made testing very difficult and inefficient. If we were given significantly more computational power, we could have run then model at a significantly faster pace and generated significantly more impactful and meaningful categories that we were

unable to produce due to our limited resources.

6.2 Considerations for Future Work

There are two main considerations that we could employ in the future to yield better results. The first would be our computational power. Due to the amount of computing some of the models we used required, we were unable to achieve as meaningful results as we hoped. With more computational power, this allows us to perform more iterations and change the input more, which could provide better results. The second consideration is our category selection system. We had to manually remove many non meaningful generated categories, such as verbs. Additionally, there are still some categories that may not describe genres of books properly, despite being a noun. With a better filtering system when generating our initial pool of categories, we could have more meaningful categories that will let the bart-larger-mnli yield better results.

7 Individual Contributions

Alexei Chen was responsible for web scraping the Twitter trends website for the trends, and to find the corresponding trends' summaries on Wikipedia. He converted the trends into a JSON format and exported the trends, rankings, summaries of the trends, and date of trend to a JSON file. Additionally, he was part of the team that evaluated the data. He also wrote the data collection, evaluation, and conclusion of the report.

Zedong Chen was responsible for scraping book summaries and book reviews from GoodRead. He also handled parts of the data conversion such as txt_to_json.py and category score calculation. Besides, Zedong collaborated with other team members on the evaluation process and composing the README and the final project report.

Tian Xia was responsible for handling data processing and trend prediction. He implemented the algorithm outlined in our paper and wrote interaction with those two models. Additionally, he designed the prompt for trend word prediction and conducted experiments to obtain the final labels. Furthermore, Tian Xia collaborated on writing sections of the introduction and method, and also contributed to writing the README.

Qisi Yang was responsible for getting book descriptions from Google API. She wrote the interface which read data from multiple data sources and produced term labels for each data source. She merged multiple toolkits like nltk, a self-constructed stopwords library and phrasemachine to tokenize the text and rank labels. She collaborated to write the zero-shot classification function and wrote the recommendation functions and front-end page to display the results.

Harry Yang contributed to compiling the Twitter data and creating the manual dataset for evaluating the effectiveness of our approach. He was also responsible for reviewing the relevant literature for techniques like expert prompting to improve performance and wrote the methods sections.

References

Tim Althoff, Damian Borth, Jörn Hees, and Andreas Dengel. 2013. Analysis and forecasting of trending topics in online media streams. In *Proceedings of the 21st ACM international conference on Multimedia*, pages 907–916.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *34th Conference on Neural Information Processing Systems (NeurIPS 2020)*.

Hao Ding, Branislav Kveton, Yifei Ma, Youngsuk Park, Venkataramana Kini, Yupeng Gu, Ravi Divvela, Fei Wang, Anoop Deoras, and Hao Wang. 2023. Trending now: Modeling trend recommendations. In *Proceedings of the 17th ACM Conference on Recommender Systems*, pages 294–305.

Jiachang Liu, Dinghan Shen, Yizhe Zhang, Bill Dolan, Lawrence Carin, and Weizhu Chen. 2021. What makes good in-context examples for GPT-3? *arXiv*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A robustly optimized BERT pretraining approach. *arXiv*.

Danny McLoughlin. 2023. Amazon print book sales statistics. WordsRated. Available online: https://wordsrated.com/amazon-print-book-sales-statistics/ [Accessed January 13, 2023].

Benfeng Xu, An Yang, Junyang Lin, Quan Wang, Chang Zhou, Yongdong Zhang, and Zhendong Mao. 2023. ExpertPrompting: Instructing large language models to be distinguished experts. *arXiv*.