KEYTREND: AUTOMATED KEYWORD SYNTHESIS VIA LLMS FOR DEMAND FORECASTING WITH GOOGLE TRENDS

Ali Barooni
GIGL, Polytechnique Montréal & GERAD
Montréal, Canada
ali.barooni@etud.polymtl.ca

Patrick Munroe
GERAD, HEC Montréal
Montréal, Canada
patrick.munroe@gerad.ca

Frédéric Quesnel
AOTI, Université du Québec à Montréal & GERAD
Montréal, Canada
frederic.quesnel@uqam.ca

ABSTRACT

Google Trends (GT) can provide useful external signals for demand forecasting, but choosing the right keywords is usually done manually and depends heavily on domain knowledge. In this work, we introduce **KeyTrend**, a framework where an LLM generates GT keyword candidates from e-commerce websites, and a correlation-based filter selects the most relevant ones. We evaluate on four real-world daily logistics shipment datasets from a major Canadian freight forwarder and find that a fixed set of 25 filtered GT keywords consistently improves XGBoost SMAPE across all datasets, even without any hyperparameter tuning. We also benchmark against foundation models (Chronos, TimesFM) and classical methods, and observe that no single approach dominates across all domains. The performance of the foundation models is similar to the statistical baseline. We also discuss practical lessons about prompt strategies and keyword selection that can guide future work in LLM-augmented forecasting.

Track: Industry & Applications

1 INTRODUCTION

In traditional demand forecasting, exogenous variables typically come from the physical environment like weather, calendar effects, holidays, and promotions. While informative for brick-and-mortar retail, these signals are less relevant for e-commerce, where demand is mediated through digital channels. Google Trends (GT), which captures search query volume over time, is a more natural exogenous signal for e-commerce forecasting (Choi & Varian, 2012; Al-Basha, 2021; Bangwayo-Skeete & Skeete, 2015; Lind & Ramesh, 2025). However, most studies select keywords manually, which is subjective and hard to scale. Recent work shows that LLMs can extract keywords from text with good quality (Ben Mansour et al., 2025), but this has not been applied to demand forecasting.

At the same time, time series foundation models such as TimesFM (Das et al., 2024) and Chronos (Ansari et al., 2024) have shown strong zero-shot ability, performing on par with traditional methods though no single family dominates (Puvvada & Chaudhuri, 2024). However, these models are primarily univariate and cannot natively incorporate exogenous covariates. To address the challenges of manual keyword selection and domain knowledge, we propose **KeyTrend**, a framework that uses LLMs to generate and filter GT keywords as exogenous covariates for XGBoost. We evaluate on four real-world logistics datasets and benchmark against statistical methods and foundation models.

2 THE KEYTREND FRAMEWORK

Our framework has three steps (details in the appendix). In **Step 1**, e-commerce website content is fed to an LLM to generate $\sim$200 search-ready keywords per dataset. In **Step 2**, each keyword is extracted from Google Trends; the resulting weekly series are resampled to daily frequency and feature-engineered with lags and growth rates. In **Step 3**, keywords are filtered using correlation-based stability analysis: we compute Pearson and Spearman correlations on the training set (60%), verify sign consistency on the validation set (20%), and rank by a composite stability score. The top-25 keywords serve as exogenous past covariates; the model is retrained on train+validation (80%) and evaluated on the held-out test set (20%).

3 EXPERIMENTS AND RESULTS

Data. We evaluate on real-world logistics data from a major Canadian freight forwarder. The target variable is *daily shipment count*. We use four e-commerce clients spanning different domains: A (clothing, 408 days), B (cosmetics, 499 days), C (sporting equipment, 730 days), and D (women's clothing, 238 days), totalling $\sim$290,000 shipments. While we use a single temporal split, the four datasets span different industries and time ranges (training periods from 2020–2022), providing natural diversity in seasonal and business conditions. Full dataset details are in Appendix E.

Results. Table 1 reports SMAPE (Symmetric Mean Absolute Percentage Error), a scale-independent metric that enables fair comparison across datasets with different demand volumes (Hewamalage et al., 2023). We also include Seasonal Naïve (repeat last week) as a simple baseline. Foundation models use rolling 7-day forecasts. We use XGBoost with default hyperparameters (no tuning) to isolate the GT contribution: its determinism (std=0) ensures observed improvements are attributable to GT covariates rather than seed variance. We also evaluated neural models (N-BEATS, N-HiTS, TFT) but found them unsuitable for this ablation due to high seed sensitivity and inconsistent results across implementations.

Table 1: Forecasting SMAPE% (lower is better). Foundation models use rolling 7-day forecasts; Random GT averages 20 draws. Best per column in **bold**.

Category	Method	A	B	C	D
Baseline	Seasonal Naïve	51.18	61.25	82.16	72.98
	ETS	**31.13**	63.83	73.18	81.61
Foundation	Chronos	33.39	62.68	56.88	87.57
	TimesFM	52.30	**52.21**	57.33	**47.56**
XGBoost	No GT	59.67	61.51	57.97	59.57
	+ Random GT (K=25)	61.1$\pm$3.4	57.0$\pm$1.8	56.7$\pm$2.3	60.4$\pm$2.8
	+ GT (K=25)	55.07	56.28	**55.64**	59.29

GT contribution. Correlation-selected GT covariates consistently improve XGBoost SMAPE on all four datasets: A ($-$4.6pp), B ($-$5.2pp), C ($-$2.3pp), and D ($-$0.3pp), notably with default hyperparameters and no model tuning, suggesting further gains are achievable. Randomly selected keywords (20-draw average) actually hurt on A ($+$1.5pp) and D ($+$0.8pp), confirming that the correlation-based filtering step is essential. In other words, not just any GT data helps.

Sensitivity to the number of keywords. Table 2 reports SMAPE for $K \in \{10, 25, 50\}$ under both random sampling (mean$\pm$std over 20 seeds) and KeyTrend correlation-based selection. KeyTrend matches or beats random in 11 of 12 cells. On A, KeyTrend's benefit grows monotonically with K while random hovers near zero. On B, both converge around $+$7-9% improvement as the dataset has a strong base signal across most keywords. On C and D the best operating point is K=25; adding more keywords beyond this point yields diminishing or negative returns. For C at K=10, the small filtered set appears to overfit to a few highly correlated keywords that do not generalize to the test period. Overall, based on sensitivity analysis on K, performance is stable in the $K \in [15, 30]$ range, indicating that KeyTrend is not fragile to a specific keyword-count choice.

Table 2: Sensitivity to the number of selected keywords. SMAPE% (lower is better) on the test set for $K \in \{10, 25, 50\}$, comparing *Random* keyword selection (mean±std over 20 seeds) against *KeyTrend* correlation-based selection. Best per row in **bold**.

	No GT	K=10		K=25		K=50	
Dataset	XGB	Random	KeyTrend	Random	KeyTrend	Random	KeyTrend
A	59.67	63.0±6.1	57.93	61.1±3.4	55.07	59.7±4.1	**52.25**
B	61.51	57.2±1.7	56.76	57.0±1.8	**56.28**	56.9±1.3	56.32
C	57.97	58.7±4.2	70.45	56.7±2.3	**55.64**	57.7±2.7	59.65
D	59.57	61.4±3.2	64.06	60.4±2.8	**59.29**	60.5±4.6	61.00

Foundation model comparison. No single model family dominates: ETS achieves the best SMAPE on A (31.13), TimesFM wins on B (52.21) and D (47.56), and XGBoost+GT is best on C (55.64). TimesFM is the overall most consistent performer across datasets.

Why intermediate-specificity keywords work best. Very generic terms (e.g., "beauty products") have stable search volumes that do not capture demand fluctuations. Very specific terms may have too little search volume. The most useful keywords are at an intermediate level (e.g., "pink gel polish", "custom sweatshirts"), capturing seasonal patterns that align with purchasing behavior.

4 Conclusion

We presented KeyTrend, a framework that uses LLMs to generate and filter Google Trends keywords for demand forecasting. Using a correlated set of 25 keywords, GT covariates consistently improve out-of-the-box XGBoost (untuned) SMAPE across all four e-commerce datasets, and a random-keyword ablation confirms the correlation-based filtering is an essential step. Foundation models are competitive with classical methods on real logistics data, though no model family dominates. Since current foundation models are primarily univariate, a natural next step is integrating GT covariates with foundation models that support exogenous inputs. Also, investigating the temporal stability of selected keywords and optimal refresh frequency is left for future work.

Acknowledgements

We would like to sincerely thank the anonymous reviewers for their valuable feedback and constructive comments, which helped improve the quality of this paper. We are also grateful to François Soumis for his valuable insights throughout this project.

References

Feras Al-Basha. Forecasting retail sales using Google Trends and machine learning. Master's thesis, HEC Montréal, 2021.

Abdul Fatir Ansari, Lorenzo Stella, Caner Turkmen, Xiyuan Zhang, Pedro Mercado, Huibin Shen, Oleksandr Shchur, Syama Sundar Rangapuram, Sebastian Pineda Arango, Shubham Kapoor, et al. Chronos: Learning the language of time series. *arXiv preprint arXiv:2403.07815*, 2024.

Prosper F Bangwayo-Skeete and Ryan W Skeete. Can Google data improve the forecasting performance of tourist arrivals? Mixed-data sampling approach. *Tourism Management*, 46:454–464, 2015.

Nacef Ben Mansour, Hamed Rahimi, and Motasem Alrahabi. How well do large language models extract keywords? A systematic evaluation on scientific corpora. In *Proceedings of the 1st Workshop on AI and Scientific Discovery*, pp. 13–21, 2025.

Hyunyoung Choi and Hal Varian. Predicting the present with Google Trends. *Economic Record*, 88 (s1):2–9, 2012.

Abhimanyu Das, Weihao Kong, Andrew Leber, Rajat Mathews, and Rihan Sen. A decoder-only foundation model for time-series forecasting. *arXiv preprint arXiv:2310.10688*, 2024.

Hansika Hewamalage, Klaus Ackermann, and Christoph Bergmeir. Forecast evaluation for data scientists: common pitfalls and best practices. *Data Mining and Knowledge Discovery*, 37(2): 788–832, March 2023. ISSN 1573-756X. doi: 10.1007/s10618-022-00894-5. URL https://doi.org/10.1007/s10618-022-00894-5.

Gary Lind and K Ramesh. Using internet search data to predict aggregate retail sales and enhance firm-level revenue expectations. *Contemporary Accounting Research*, 42(3):1557–1588, 2025.

Santosh Kumar Puvvada and Satyajit Chaudhuri. Critical evaluation of time series foundation models in demand forecasting. In *NeurIPS Workshop on Time Series in the Age of Large Models*, 2024. URL https://openreview.net/forum?id=TS42sRKINd.

A KeyTrend Pipeline Overview

Figure 1 shows the full KeyTrend pipeline: (1) LLM-based keyword generation from e-commerce websites, (2) Google Trends extraction and feature engineering, and (3) correlation-based filtering and forecasting.

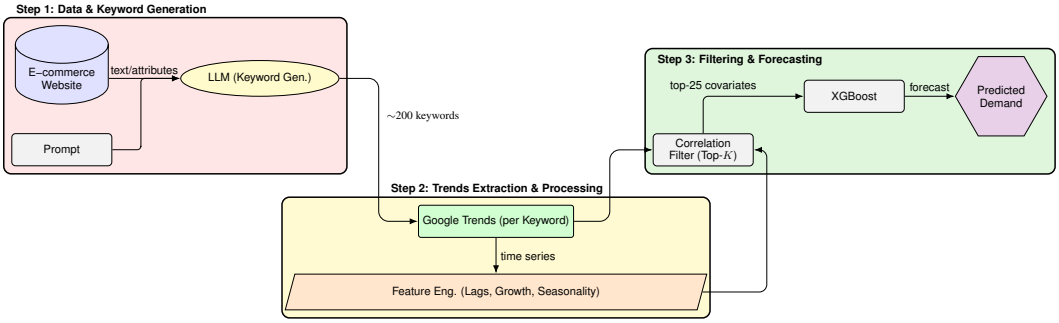

Figure 1: KeyTrend pipeline overview. **Step 1**: E-commerce website content is processed by an LLM (Claude Opus, product-focused prompt) to generate ∼200 search-ready keywords. **Step 2**: Keywords are queried against Google Trends API, and the resulting time series undergo feature engineering. **Step 3**: Keywords are filtered by correlation-based stability analysis; the top-25 serve as exogenous covariates for XGBoost.

B Step 1: Data Collection & Keyword Extraction

Website content collection. For each client, we collected product catalog pages including titles, descriptions, and category information. Where available, we used archived website snapshots (via the Wayback Machine) to ensure the product catalog reflects the historical training period rather than the current website.

LLM keyword generation. We tested three prompt strategies with four LLMs. Table 3 shows the results. The *product-focused* prompt with Claude Opus generated the highest quality keywords: specific, search-ready, and closely matching queries real consumers would use on Google. This configuration was used for all experiments reported in this paper.

Prompt strategies. The *SEO-style* prompt asks the LLM to act as an SEO (Search Engine Optimization) specialist and generate keywords that potential customers would search. The *category-focused* prompt organizes keywords by product category. The *product-focused* prompt provides individual product titles and descriptions and asks for the most relevant search terms per product. This last strategy produced the most specific, actionable keywords.

Table 3: Keyword extraction across prompt strategies and LLMs. The product-focused prompt with Claude Opus (used in all experiments) produced the best results.

Prompt Strategy	**Model**	**Keywords**	**Quality**
SEO-style	Claude Sonnet 3.5	103	Good
Category-focused	GPT-4 Turbo	102	Systematic
Product-focused	Claude Opus 4.5	100	Best
Product-focused	Llama 3.1 70B	30	Failed

C Step 2: Google Trends Extraction & Feature Engineering

Trends extraction. Each keyword is queried against the Google Trends for both country-specific (Canada) and worldwide trends, yielding $\sim$200 keyword time series per dataset.

Feature engineering. Weekly Google Trends data is (1) resampled to daily frequency via forward-fill, (2) augmented with 1–4 week lag features, and (3) enriched with week-over-week growth rates. All features are standardized using only training-period statistics (μ, σ) to prevent data leakage.

D Step 3: Keyword Filtering & Forecasting

Not all LLM-generated keywords improve forecasting; some of them introduce noise. We use correlation-based stability analysis with a strict 60/20/20 train/validation/test split to prevent selection bias.

Correlation-based filtering. For each keyword, we compute Pearson and Spearman correlations with demand on the training set (60%), using raw daily values, first-differenced values, and weekly aggregates, at lags of 0–28 days. We repeat on the validation set (20%). A keyword is "stable" if it has the same correlation sign on both sets and exceeds a minimum magnitude threshold. Keywords are ranked by a composite stability score and the top-25 are selected.

Forecasting. The top-25 filtered keywords serve as past covariates. The model is retrained on 80% of the data (train + validation) with calendar features and selected GT keywords, then evaluated on the held-out 20% test set.

E Dataset Details

Table 4 summarizes the four datasets.

Table 4: Dataset characteristics

	A	**B**	**C**	**D**
Domain	Clothing	Cosmetics	Sporting Equip.	Women's clothing
Daily avg. shipments	291.5	177.4	57.0	189.7
Time span (days)	408	499	730	238
Total shipments	117,165	88,538	38,862	43,817
Test period (days)	82	100	146	48
Train / Val / Test	60% / 20% / 20% (temporal split)			
Extracted GT keywords	197	205	210	184

All the datasets show weekly seasonality (lower volumes on weekends) and occasional promotional spikes. Data was anonymized by the freight forwarder; only aggregated daily shipment counts and dates were provided.

F GLOBAL VS. COUNTRY-LEVEL GOOGLE TRENDS

Each keyword is fetched at two geographic levels: worldwide (*global*) and Canada-specific (*country*). Table 5 separates the top-25 KeyTrend keywords by geo level and evaluates each subset independently. Neither level consistently dominates: country-level keywords contribute most for B, global keywords are more informative for C, while both contribute roughly equally for A. The correlation-based filter implicitly selects the best geographic mix per domain, removing the need for manual geo-level tuning.

Table 5: SMAPE% by geographic level of GT keywords (lower is better).

GT Source	**A**	**B**	**C**	**D**
No GT (baseline)	59.67	61.51	57.97	59.57
Global only	54.71	59.68	**55.78**	62.01
Country only	54.61	**54.25**	59.43	60.20
Both (K=25)	**55.07**	56.28	55.64	**59.29**

G USE OF LLMS

We used LLMs for language polishing and improving the clarity of the manuscript. Additionally, LLM-based coding assistants were used to assist with code debugging and implementation. All research ideas, experimental design, methodology, analysis, and interpretation of results were conceived and carried out independently by the authors. The authors take full responsibility for the content of this paper.