ReviewMirror– Tracking Opinion Drift in E-Commerce Reviews

Shubham (202411066), Ritwik (202411067), Dhairya (202411082)

1 Problem Statement & Objectives

We study how an individual user's sentiment and writing tone change across their review history. Our objectives are: (1) analyze temporal patterns of opinions in longitudinal review data [6] ; (2) define a representation to compare a user's sentiment/style over time; (3) quantify and visualize drift with interpretable linguistic or behavioral signals; (4) justify modeling and evaluation choices grounded in prior work.

2 Datasets

Primary: Amazon Review Data (2018, UCSD). The 2018 release contains ∼233.1M reviews spanning May 1996–Oct 2018 with fields such as `reviewerID`, `asin`, `reviewText`, `overall` (stars), and `unixReviewTime`. It also provides a *5-core* subset (users/items with ≥ 5 reviews) and per category splits , we will use the **Electronics 5-core** subset for computational tractability and longitudinal coverage [1]
Forward-compatibility. McAuley Lab released a 2023 expansion (571.5M reviews to Sep 2023) with richer metadata and maintained category mappings. We keep this as an upgrade path if broader coverage is needed [2].

Backup: Yelp Open Dataset. JSON files for reviews (`review_id`, `user_id`, `business_id`, `stars`, `date`, `text`), businesses, and users. Suitable for cross-domain validation in services settings [3].

3 Data Model & Schema

We will curate one tidy row per review:

Field	Description
`user_id`	Reviewer identifier (`reviewerID` / `user_id`)
`item_id`	Product/business identifier (`asin` / `business_id`)
`ts`	Timestamp (from `unixReviewTime` or `date`)
`text`	Raw review text
`stars`	Star rating (1–5)
`category`	(Optional) high-level category for stratification
`helpful_votes`	(Optional) helpfulness/engagement signal

Derived features for trajectories: (i) *text sentiment* (lexicon baseline, e.g., VADER; then transformer-based polarity), (ii) *style/tone* proxies (exclamation density, first-person ratio, subjectivity/readability), (iii) time indices (calendar month, time-since-first-review).

4 Preprocessing Pipeline

1. **Load the data.** Read the per category JSON dump (gzipped or plain) into a tidy dataframe with the key fields we need (user, item, text, stars, timestamp).
2. **Basic cleaning.** Keep reviews that look like English text, drop empty or boilerplate entries, and remove near duplicates.
3. **Make sure users have enough history.** Keep only users with at least k reviews (default k=5) so each user has a meaningful timeline to analyze.
4. **Put everything on a timeline.** Convert timestamps to UTC, bucket them by calendar month, and sort reviews within each user so trajectories are well ordered.
5. **Add lightweight labels.** Compute a quick text polarity score (VADER compound in $[-1, 1]$), map star ratings to $[-1, 1]$, and combine them into a simple hybrid sentiment. For comparability, also keep a within user z score version.
6. **Save the essentials.** Export two compact artifacts for the next steps: `reviews.parquet` (row level records) and `user_trajectories.parquet` (per user monthly sequences with features).

5 Representation of Opinion Trajectories

Trajectory: For each user u we build a time ordered sequence of monthly points $\{(t_i, s_i, r_i, \mathbf{f}_i)\}_{i=1}^{n_u}$, where t_i is the month, s_i is the text based sentiment score, r_i is the rating mapped to $[-1, 1]$, and $\mathbf{f}_i$ are simple style cues (e.g., exclamation rate, first person usage, capitalization). This gives us a compact, per user timeline we can compare and visualize.

- **Hybrid sentiment (single track to follow).** We blend text polarity and stars into one signal

$$h_i \;=\; \alpha\, s_i \;+\; (1-\alpha)\,\tilde{r}_i, \qquad \alpha \in [0,1],\ \tilde{r}_i \in [-1,1].$$

  Intuition: h_i is high if both the review *sounds* positive and the *stars* are high. (Default α=0.7; we will vary α in robustness checks.)

- **Smoothed path (for pictures, not decisions).** For cleaner plots we may show an exponentially weighted moving average $\bar{h}_t = \mathrm{EWMA}_\lambda(h_i)$. We keep all *metrics* (slopes, deltas, detectors) on the *monthly means* to avoid over smoothing.

- **Drift magnitude (how much did the opinion move).** We summarize change with two simple numbers:

  1. *Trend (slope):* fit a line to h_i versus true time (month ordinal), positive means upward drift.
  2. *Start→End delta:* $h_{\text{last}} - h_{\text{first}}$ (direction and size of net change).

- **Change points (when did the shift happen).** We run standard detectors on h_i:

  1. **ADWIN** for mean shifts without a fixed window size,
  2. **Page–Hinkley** for gradual, persistent drifts.

  Close-by alarms are merged (e.g., within ±1 month) to avoid duplicate events.

Together, this representation gives us a clear, single trajectory per user (the hybrid h_i), simple scalars to quantify *how much* it changed, and timestamps indicating *when* it changed while keeping the raw ingredients (text, stars, style) available for interpretation.

6 Quantifying & Visualizing Drift

What we measure: We quantify *how much* a user's opinion moved, *how wiggly* it was, and *when* any major shift happened. All metrics are computed on monthly hybrid sentiment h_t (with stars only variants for checks).

Quantification (small set of metrics)

- **Trend (signed slope).** Fit a line to h_t vs. true time (month ordinal):

$$\hat{\beta}_1 \;=\; \arg\min_{\beta_0,\beta_1} \sum_t \big(h_t - (\beta_0 + \beta_1\, t)\big)^2.$$

  Positive $\hat{\beta}_1$ means drifting upward, negative means drifting downward.

- **Net change (start→end).** $\Delta h = h_{\text{last}} - h_{\text{first}}$. Easy to interpret direction and magnitude.
- **Volatility (total variation).** $\text{TV} = \sum_t |h_t - h_{t-1}|$. Higher TV = more ups/downs.
- **Direction instability (sign flips).** $\text{FlipRate} = \frac{1}{T-1} \sum_t \mathbf{1}\{\text{sign}(h_t) \neq \text{sign}(h_{t-1})\}$.
- **Early vs. Late divergence.** Compare distributions of sentiment in the first K vs. last K months using a bounded, symmetric distance such as Jensen–Shannon divergence (or KL if desired):

$$D_{\text{JSD}}\big(P_{\text{early}}, P_{\text{late}}\big) \in [0, 1].$$

- **Change point timing.** From detectors (ADWIN/Page–Hinkley): (i) *count* of changes, (ii) *time-to-first-change* (months from start), and (iii) *dwell time* between changes.

Quality checks. We also report agreement with ratings: Spearman correlation between slope(h_t) and slope(stars), and the percentage of users where the signs match.

Visualization (make the patterns obvious)

- **Per user sparklines.** Small line plots of h_t over months with change points marked.
- **Cohort heatmaps.** Users grouped by first review year (or tenure deciles), showing average slope or share of "drift" users per cohort reveals period effects.
- **Rating transitions (alluvial / Sankey).** Early→Late star distributions to show how satisfaction shifts (e.g., 5→3, 2→4).
- **Wordshift / top terms.** Small wordshift style views highlighting tokens that most explain Early vs. Late differences in polarity.
- **Population summaries.** Histograms/violin plots of $\hat{\beta}_1$ and Δh, scatter of slope(h_t) vs. slope(stars) to visualize agreement.

7 Modeling & Evaluation Rationale

7.1 Modeling Rationale

- **The Problem:** User behavior isn't static, it changes over time. This is known as **concept drift** [4].
- **Our Hypothesis:** We believe models that *adapt* to these changes will be more accurate than models that treat all data as the same ("static").

- **The Test:** We will directly compare **static baseline models** (which ignore change) against **adaptive detectors** (like ADWIN [5] and Page-Hinkley) that are designed to spot these changes as they happen.
- **Making it Understandable (Interpretability):** When our model flags a "change-point," we'll confirm it's a *real* change by checking if the user's actual **writing style or word choice (n-grams)** also shifted at that same time.
- **Tracking Gradual Change:** We'll also explore methods like **Dynamic Topic Models** [7] to get a "big picture" view of how a user's discussion topics slowly evolve over their history.

7.2 Evaluation Rationale

We'll test our models in three ways to ensure the results are trustworthy:

1. **Internal Validity (Are we right?):** We need a "ground truth" to check our work. We'll use **star ratings** as our proxy. If our model detects a change in a user's *text* right when their *star ratings* suddenly jump or fall, we can be confident the detected change is meaningful.
2. **Robustness (Are we lucky?):** We will "stress-test" our models by tweaking their settings (like sensitivity α, bin width, etc.). This ensures our results aren't just a fragile fluke of one specific setup.
3. **External Validity (Does it work elsewhere?):** To prove our method is widely useful, we will experiment it on other datasets.

Our main success metric will be **user-level AUC**, which measures how well the model can correctly classify a user as having "changed" vs. "not changed."

8 Preliminary Visual Checks

To validate the pipeline, we plotted per-user monthly hybrid sentiment trajectories. Figure 1 shows representative patterns (steady increase, U-shape, volatile, plateau). Table 1 summarizes drift metrics for a small sample of users.

user_id	drift_slope	drift_delta	tv	flip_rate
A3UFCX1AE4TKZE	1.275	1.275	1.275	1.0
A11GHM8Q3IEXHI	0.561	1.123	1.123	0.5
A3PUCTT3HOZ77V	0.512	0.512	0.512	0.0
A2VB5FKZMKLVPV	0.511	0.511	0.511	0.0
A7M4QUHGIY69U	0.406	0.406	0.406	0.0
A3CXLR0Y89GTB5	-0.918	-0.918	0.918	1.0
A11D26I3MRPYUF	-0.846	-0.846	0.846	1.0
ARXL1PBCOYXL3	-0.828	-0.828	0.828	0.0
A17LCR0YZ0SOOP	-0.545	-0.545	0.545	0.0
A1517476XFOC7A	-0.448	-0.448	0.448	0.0

Table 1: Sample user drift metrics: slope (true time), start→end delta, total variation (TV), and sign-flip rate.

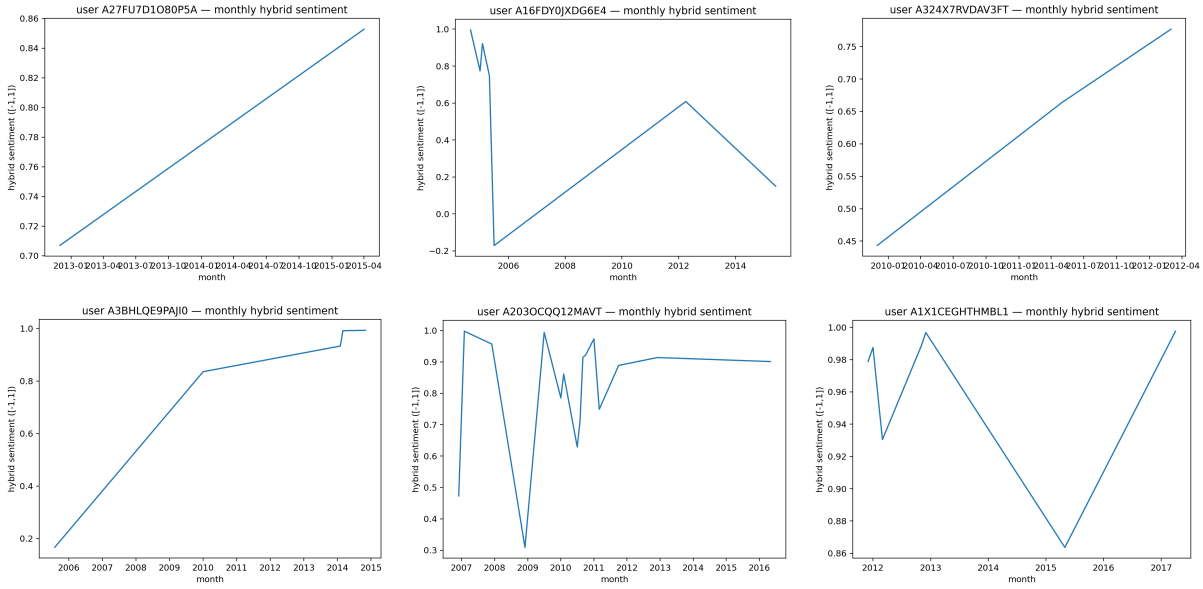

Figure 1: Representative per-user monthly hybrid sentiment trajectories.

References

[1] J. Ni, J. Li, J. McAuley. Amazon Review Data (2018). https://cseweb.ucsd.edu/~jmcauley/datasets/amazon_v2/.

[2] McAuley Lab. Amazon Reviews 2023 (571.5M reviews; May 1996–Sep 2023). https://amazon-reviews-2023.github.io/.

[3] Yelp Open Dataset. Reviews/business/users JSON schema and downloads. https://business.yelp.com/data/resources/open-dataset/.

[4] J. Gama, I. Žliobaite, A. Bifet, M. Pechenizkiy, A. Bouchachia. A Survey on Concept Drift Adaptation. *ACM Computing Surveys*, 2014.

[5] A. Bifet, R. Gavalda. Learning from Time-Changing Data with Adaptive Windowing (ADWIN). *SIAM International Conference on Data Mining*, 2007.

[6] Y. Koren. Collaborative Filtering with Temporal Dynamics. *Communications of the ACM*, 2010.

[7] D. Blei, J. Lafferty. Dynamic Topic Models. *ICML*, 2006.