Thanks for the context. I'll look into best practices for identifying and interpreting user behavior patterns from clickstream graphs—especially those that indicate dissatisfaction—and how to effectively map them to logical page groups like Account Access, Claims, Support, etc. I'll also explore statistical validation approaches (chi-squared vs. Fisher's) and how to structure these insights for business-facing reports.

I'll get back to you shortly with a summary of techniques and recommendations.

Identifying Significant User Behavior Patterns and Pain Points

Calculating Pattern Metrics (Support, Leverage, Growth, Interest)

After extracting all cyclic and simple path patterns from the clickstream sessions, the next step is to quantify how strongly each pattern is associated with dissatisfied sessions. We can compute several **interestingness metrics** for each pattern using the raw counts from the data (e.g. how many times the pattern occurs in dissatisfied vs. satisfied sessions). Key metrics include:

- **Support in Dissatisfied (support_d)**: The proportion of dissatisfied sessions containing the pattern. This is essentially the frequency of the pattern in the dissatisfied group divided by the total number of dissatisfied sessions.

- **Support in Satisfied (support_s)**: The proportion of satisfied sessions containing the pattern (frequency in satisfied group / total satisfied sessions).

- **Leverage**: The difference in support between the two groups, defined as *support_d – support_s*. This represents the absolute increase in likelihood of the pattern in dissatisfied sessions compared to satisfied sessions. A positive leverage means the pattern is more prevalent among unhappy users.

- **Growth Rate**: The ratio of the supports, defined as *(support_d + ε) / (support_s + ε)* (with a small ε to avoid division by zero). This **growth** metric (also known as *growth rate* in contrast mining) tells you how many times more frequently the pattern appears in dissatisfied sessions relative to satisfied sessions. A growth > 1 indicates the pattern is more common in dissatisfied users (an "emerging" pattern of dissatisfaction), whereas a growth < 1 would mean it's actually more common in satisfied users.

- **Interest**: A combined interestingness score defined (in your method) as *leverage × log(1 + growth)*. This heuristic gives higher weight to patterns that not only have a big absolute difference in occurrence (high leverage) but also a large relative difference (high growth ratio). In other words, a pattern that is both **much more frequent** in dissatisfied sessions *and* has a high multiplier compared to satisfied sessions will score a high interest value.

Computing these metrics for each pattern is straightforward arithmetic on your pattern frequency DataFrame and is very fast even for tens of thousands of patterns (100K+ rows). This quantitative scoring helps rank the patterns by their potential relevance to user dissatisfaction. For example, a cycle or path that appears in 5% of dissatisfied sessions but only 1% of satisfied sessions has support_d = 0.05, support_s = 0.01, leverage = 0.04, growth ≈ 5, and a sizable interest score. Such a pattern would stand out compared to one that appears 0.5% vs 0.1% (leverage 0.004, growth 5) or one that appears 5% vs 4.5% (leverage 0.005, growth ~1.11). By sorting patterns on these metrics (e.g. highest interest or highest leverage), you get a **preliminary list of candidate "pain point" patterns** – those that disproportionately occur in dissatisfied user sessions.

Statistical Significance Testing of Patterns

While the metrics above help rank patterns by magnitude of difference, it's crucial to verify that these differences are **statistically significant** and not just due to noise (especially given you have 250k sessions and potentially many patterns). The recommended approach is to perform a statistical hypothesis test for each pattern using its **2×2 contingency table** of presence/absence vs. satisfaction outcome. For each pattern, you can construct a table:

	Dissatisfied (Class 0)	**Satisfied (Class 1)**
Pattern occurs	a (e.g. count in dissat.)	b (count in sat.)
Pattern not occur	N_d – a	N_s – b

Where *N_d* is total number of dissatisfied sessions and *N_s* is total number of satisfied sessions. We then test the null hypothesis that the pattern occurrence is independent of satisfaction (i.e. no association). Two common tests are:

- **Chi-square test of independence** – suitable if expected counts are not too low. This tests whether the differences in proportions (supports) could be due to chance.

- **Fisher's Exact test** – a conservative choice especially when some counts are small. Fisher's test computes an exact p-value for the 2×2 table without relying on large-sample approximations. Many pattern mining frameworks prefer Fisher's test for reliability in these situations.

For each pattern, you would obtain a p-value indicating the likelihood of seeing a difference as extreme as *support_d vs support_s* if there were actually no real effect. After computing p-values for all patterns, **apply a multiple comparisons correction** such as the Benjamini–Hochberg procedure to control the False Discovery Rate (FDR). This step is essential because when you test thousands of patterns, some will show up as different just by chance. The Benjamini–Hochberg (BH) correction will adjust p-values so that you can use a threshold (e.g. $\alpha = 0.01$) while limiting the proportion of false positives among the patterns you deem significant.

Filter out patterns that do not meet significance or support thresholds. In practice, you would drop any pattern whose adjusted p-value exceeds your chosen α (say 0.01) or that has very low support in the dissatisfied group (e.g. occurs in only a handful of sessions). This two-stage approach (metric scoring then statistical testing) is grounded in best practices from data mining research: essentially a form of **contrast set mining** or **emerging pattern mining**. As noted in contrast set mining literature, we use statistical hypothesis testing to find real differences and control false positives. By doing so, we ensure the remaining patterns are **significantly associated with dissatisfaction** (not just random artifacts of the data).

After this filtering, you'll have a much shorter list of patterns (cycles/paths) that are statistically linked to user dissatisfaction. These are the *relevant patterns* likely pointing to user frustration or pain points.

Interpreting and Prioritizing Key Patterns of Frustration

The patterns that survive the significance filter can be interpreted as **pain point indicators** in the user journey. Look at each pattern's context and what it represents:

- **Cycles or Loops**: Repeated loops (e.g. A → B → A or A → A → A) often signal user frustration. If a user is cycling between the same two pages or revisiting the same page repeatedly in one session, it suggests they might be stuck, confused, or unable to complete what they intended. In UX analysis, **looped back-and-forth navigation is a red flag** – it "could indicate an issue" in the user experience. For example, a simple cycle like *Login Page → OTP Page → Login Page* appearing frequently in dissatisfied sessions may indicate problems in the authentication process (perhaps the one-time passcode wasn't working, forcing users to attempt login repeatedly). Another example might be users oscillating between a **Claims Submission** page and a **Help/FAQ** page (Claims → Help → Claims) – implying they had to seek help in the middle of filing a claim, a likely pain point.

- **Long or Unusual Paths**: A simple path that is significantly over-represented among unhappy users could highlight a problematic journey. For instance, a path like *Account Summary → Transfer Funds → Error Page → Transfer Funds → Support Page* (just as an illustration) would clearly indicate a failing transaction flow leading to frustration. When interpreting paths, check if they include error pages, multiple jumps between sections, or detours to support pages. These often reveal **friction in the workflow** (e.g. a user trying multiple times to accomplish a task or having to consult support content).

- **High interest score patterns**: The interest metric helps you find patterns that not only differ but *differ by a large factor*. Prioritize patterns with the highest interest (or highest leverage/growth) among those that are statistically significant. Those are likely the strongest candidates for true pain points. For example, a path that is 10× more likely in dissatisfied sessions (growth ~10) and also fairly frequent (say 4% of dissat users) is more critical than one that is 2× more likely and very rare.

By examining the content of these patterns, you can often hypothesize the underlying issue. It could be a **page that is hard to use**, a **feature that's broken or confusing** (causing users to bounce around), or a **missing piece of information** forcing users to detour to support. The combination of quantitative evidence (supports, growth, p-value) and qualitative reasoning about the page content will let you pinpoint what likely frustrates users.

It's also useful to cross-check these findings with other data if available – for example, session recordings, user feedback, or error logs – to validate that the patterns truly correspond to frustration (this helps "connect the dots" as to *why* those patterns are problematic).

Grouping Pages into Logical Categories for Presentation

Once you have the key problematic paths and loops, it's often **helpful to abstract and group the pages into higher-level categories** before presenting results to business stakeholders. In a banking/insurance context, individual page names or URLs might be too granular or technical for a business audience to digest. By categorizing pages into logical groups (such as by functionality or section of the site/app), you can convey the insights more clearly in business terms.

For example, you proposed grouping pages into categories like:

- **Account Access & Security** (login pages, authentication, profile security, password reset, etc.)

- **Balance / Statements** (account summary, balance inquiry, statements download)

- **Fund Transfers & Payments** (money transfer pages, bill pay, add payee, etc.)

- **Loan or Credit Applications** (pages for applying or checking loan/credit card status)
- **Claims Submission & Tracking** (insurance claim forms, claim status pages)
- **Support & Help** (help center, FAQ, contact support, chat support pages)

This kind of grouping is a well-established practice. In fact, analytics tools like Google Analytics support **Content Grouping** features to aggregate pages into broader categories for behavioral analysis. Using such groupings, you can **translate a raw page sequence into a sequence of category labels**.

How to group the sequences: Start by mapping each page in your dataset to one of these functional categories. Then for each session's page sequence, convert it into a sequence of categories. For example:

- A raw sequence `Login Page → 2FA Page → Overview Page → Transfer Funds Page → Transfer Confirmation Page → Logout` might map to `Account Access & Security → Account Access & Security → Balance/Statements → Fund Transfers & Payments → Fund Transfers & Payments → Account Access & Security`. You might even simplify consecutive repeats for clarity (the first two steps are both Account Access, the transfer pages are both Fund Transfers).
- A problematic loop like `Claim Form Page → FAQ Page → Claim Form Page → FAQ Page` would become `Claims Submission & Tracking → Support & Help → Claims Submission & Tracking → Support & Help`. It's immediately obvious in category terms that *users kept bouncing between a claim process and help*, a strong indicator of pain point.

By grouping in this way, **common patterns emerge at a higher level**. You might find, for instance, that many of the top dissatisfaction paths involve transitions between *Support & Help* and another category (suggesting users had to seek help mid-process), or multiple steps within *Account Access & Security* (suggesting login/authentication hurdles). Presenting the findings as "*Users who are frustrated often experience loops in the* ***Account Access*** *flow*" or "*Dissatisfied customers frequently navigate from* ***Claims*** *pages to* ***Help*** *content and back, indicating confusion in the claims process*" is much more digestible to a business audience than listing specific page URLs.

Moreover, grouping reduces the sheer number of distinct patterns you need to report. Instead of dozens of page-level sequences, you can summarize a handful of category-level patterns. This **storytelling approach** focuses on the *themes of friction* rather than the nitty-gritty. It answers the key question: *“Where in our online experience are customers hitting roadblocks?”* – e.g., in login/security, in making transfers, in filing claims, etc.

Communicating the Results and Next Steps

When reporting to the business team, highlight the patterns that meet all the criteria of importance: **high prevalence in dissatisfactions, large difference vs satisfied, and statistically significant**. For each such pattern (now described in terms of page categories), explain the likely pain point:

- **What the users were trying to do** (the user journey segment represented by the path) and
- **What went wrong** (inferred from the loop or detour).

For example, you might report: *“One significant pattern for dissatisfied users is a cycle between the password reset page and login page (Account Access & Security). 3.5% of unhappy sessions show users repeatedly going through password reset and back to login, compared to only 0.1% of satisfied sessions – a **35× higher occurrence**. This suggests many users struggle with logging in or resetting their password successfully, indicating a pain point in Account Access security flows.”* By backing such statements with the support and growth numbers (and noting the p-value < 0.001 for significance), you provide concrete evidence of the issue.

Another example: *“Dissatisfied users are **8× more likely** to navigate from a Fund Transfer page to the Help section and then back to the Fund Transfer (p < 0.005). This indicates a confusing funds transfer process where users leave to seek help mid-transaction. We should investigate the transfer UX and content.”* Framing it this way links the pattern to a business implication (a process that needs improvement).

In summary, the process to identify relevant patterns of frustration is:

1. **Compute support metrics** for each discovered path/cycle in both user groups and derive leverage, growth, interest scores.
2. **Rank and filter** patterns by these scores to focus on the most imbalanced ones favoring dissatisfied users.
3. **Perform statistical tests** (χ^2 or Fisher’s exact) on each pattern’s 2×2 table and apply a multiple-comparison correction (e.g. BH) to find which differences are statistically significant. Also drop patterns with trivial support counts.
4. **Interpret the remaining patterns** as potential pain points – identify what user action or feature they involve. Recognize that loops/backtracking behavior is a likely sign of

frustration.

5. **Group pages into logical categories** to abstract the patterns into business-friendly terms, e.g. grouping by functional areas like Login, Transfers, Claims, etc., as is common in customer journey analysis. Convert sequences into these categories to spot high-level problem areas.

6. **Present the findings** with an emphasis on the user's pain points: quantify how much more these patterns happen for unhappy users (support_d vs support_s), and describe in plain language what the users in those sessions were trying to do and where the experience failed them.

By following this approach, you leverage both data-driven rigor and domain knowledge to pinpoint the **behavioral patterns that correlate with dissatisfaction**. This provides the business a clear direction on where to focus improvements (e.g. fixing the pages or processes that lead to those frustrating loops and detours). The combination of statistical significance and logical grouping ensures that the pain points you report are not only real but also communicated in a meaningful way that stakeholders can act upon.

Sources:

- Hahsler, M. & Hornik, K. (2007). *New probabilistic interest measures for association rules*. Intelligent Data Analysis, 11(5). (For statistical filtering of association patterns with Fisher's test and FDR).

- Zaïane, O. (2007). *Lecture on Contrast Set Mining (STUCCO)* – contrast mining uses hypothesis testing to find differences between groups with control over false positives.

- Userpilot Blog (2023). *Optimizing User Journeys: How Users Navigate...* – notes that **looped back-and-forth behaviors indicate user frustration** in the journey.

- Bounteous Analytics Blog (2017). *Visualizing the Customer Journey* – recommends using **content grouping to group pages by type** (e.g. product pages, support pages) for flow analysis. This supports the idea of grouping banking pages into categories for clearer reporting of user journeys.