Interpretable Satisfaction Modeling in Sales Dialogues: A Metrics-based Approach Validated Against Text-based Model

Natalia Krawczyk, Alicja Kasicka, Bartosz Przybył, Justyna Gromada
Orange Research
{natalia1.krawczyk, alicja.kasicka, bartosz.przybyl, justyna.gromada}@orange.com

Abstract

Understanding drivers of customer satisfaction in sales conversations is crucial for dialogue system optimization and customer experience improvement, yet existing neural approaches lack interpretability. We introduce a sales action annotation framework that tracks offers, cross-sells, and customer decisions in sales dialogues. From these structured annotations, we derive interpretable metrics spanning conversion rates, timing and negotiation patterns, and sales outcomes to predict user satisfaction. We validate our metrics-based regression model against a transformer-based text-embedding baseline on 5200 simulated phone sales dialogues. Our interpretable model achieves strong performance (R^2=0.87 vs. 0.92 for the baseline) with full transparency, identifying key satisfaction drivers with practical implications for dialogue design: purchase completion, conversation efficiency, and negotiation patterns.

1 Introduction

Customer satisfaction is a key metric in sales conversations, influencing customer retention, brand perception, and long-term revenue. In dialogue systems, it reflects how effective and pleasant the interaction is from the user's perspective, combining both task success and user experience (Siro et al., 2022). It influences whether customers continue using the system and remain loyal (Cheng and Jiang, 2020), and can be used to predict their future engagement (Orden-Mejia et al., 2025).

While predicting user satisfaction in sales dialogues is valuable, understanding what drives positive customer experience requires interpretable models that reveal which sales behaviors matter most. Neural approaches based on transformer models (Vaswani et al., 2017) achieve high predictive accuracy (Sun et al., 2021) but function as black boxes, unable to explain why certain conversations satisfy customers. This limitation is particularly problematic in commercial settings, where actionable guidance on offer timing, cross-sell strategies, and negotiation patterns is essential. Unlike human salespeople who develop intuitions through experience, current conversational systems must be explicitly programmed or trained with interpretable rules to systematically improve sales strategies.

To enable interpretable satisfaction modeling in sales dialogues, we introduce a sales action annotation framework, inspired by dialogue act annotation (Stolcke et al., 2000), that captures the key elements of commercial interactions: offers and cross-sell attempts by the sales bot, and customer decisions. From these annotations, we derive interpretable sales-specific metrics across various categories: conversion rates, timing patterns, negotiation dynamics, sales strategies, conversation flow, and cart outcomes. We combine these sales-specific metrics with basic dialogue features (e.g. number of turns) to build a regression model for satisfaction prediction, which we validate against a transformer-based text-embedding baseline.

We apply our framework to simulated phone sales dialogues and show that our interpretable metrics-based model achieves competitive performance with the text-embedding approach. Using SHAP analysis (Lundberg and Lee, 2017), we identify key satisfaction drivers, providing actionable insights for dialogue system design. While some metrics are specific to this setting, the annotation framework and modeling approach are designed to generalize to other sales domains with appropriate metric adaptation.

Our main contributions are:

(i) An automatic, Large Language Model (LLM)-based sales action annotation framework and a set of derived interpretable sales metrics for analyzing sales dialogues.

(ii) An interpretable metrics-based regression

Proceedings of the 27th Annual Meeting of the Special Interest Group on Discourse and Dialogue, pages 565–584
August 2–5, 2026.

model for satisfaction prediction that achieves competitive performance with a transformer-based approach while enabling explainability.

(iii) SHAP-based identification and analysis of key satisfaction drivers, including conversion efficiency, negotiation success, and cart outcomes, providing actionable guidance for sales dialogue design.

The paper is organized as follows: Section 2 reviews related work on dialogue evaluation and user satisfaction. Section 3 describes our dataset and satisfaction annotation method. Section 4 introduces our sales action annotation framework and derived sales metrics. Section 5 describes our modeling approaches, with detailed satisfaction driver analysis in Section 6. Finally, Section 7 concludes and Section 8 discusses future work.

2 Related work

From Task Success to User Satisfaction Traditional approaches to evaluating task-oriented dialogue (ToD) systems focus on binary task success metrics (Braggaar et al., 2023), typically defined as an exact match between system response and user goal (Peng et al., 2021). Modern evaluation methods have refined this approach: WB-Score rates answer quality on a 1-10 scale based on task-specific criteria, while WB-Reward uses pairwise comparisons (Lin et al., 2025), and LLM-based evaluation with models like GPT-4 as judges has gained popularity (Kazi et al., 2024).

However, task completion alone does not capture dialogue quality. In sales contexts, customers may complete purchases while feeling dissatisfied, or conversely, have positive experiences without buying. Research shows that maximizing task success can lead to efficient but unnatural dialogues (Davidson et al., 2023), suggesting that how goals are achieved matters alongside whether they are achieved efficiently.

This limitation has led to growing focus on user satisfaction as a more comprehensive measure of dialogue system performance. User satisfaction represents how effective and pleasant the interaction is from the user's perspective (Siro et al., 2022), capturing not just task completion, but overall interaction quality (Feng et al., 2023). It is influenced by multiple factors: utilitarian gratifications (task effectiveness), hedonic gratifications (interaction pleasantness), and social gratifications (perceived social presence) (Cheng and Jiang, 2020). Importantly, satisfaction does not always align with objective task success: users might be satisfied even when tasks are incomplete, and vice versa (Liu et al., 2023). Moreover, satisfaction varies across users and contexts, with factors like response time and empathy influencing satisfaction differently across cultures (Liu et al., 2023).

Modeling User Satisfaction The PARADISE framework (Walker et al., 1997) provides a foundational approach to modeling user satisfaction in dialogue systems, combining Task Success Rate and Dialogue Cost (efficiency measures like turn count and response time). Originally evaluated on information-seeking dialogues (train timetable information and email management), PARADISE demonstrated that satisfaction can be modeled from objective metrics via linear regression, reducing the need for constant user feedback.

Recent work has explored neural models for satisfaction prediction. Sun et al. (Sun et al., 2021) find that distributed representations outperform feature-based methods, with hierarchical GRU models (Cho et al., 2014) achieving the best in-domain performance, and BERT-based models (Devlin et al., 2019) showing better cross-domain generalization. However, such neural approaches function as black boxes, providing little insight into which dialogue behaviors drive satisfaction. This lack of interpretability is particularly limiting in commercial settings, where actionable guidance on sales strategies is essential.

Our work builds on both modeling approaches, i.e., metrics-based inspired by PARADISE and transformer-based text-embeddings, by focusing specifically on sales dialogues, where satisfaction can depend not only on task completion, but also on sales strategies, negotiation dynamics, and purchase outcomes. To enable the metrics-based approach, we propose a sales action annotation framework and derive interpretable metrics for satisfaction prediction.

3 Dialogue Dataset

Our dataset consists of 5200 English conversations between an experimental sales agent selling mobile phones and LLM-based user simulators representing our customers. This section describes the dialogue generation process and user satisfaction annotation method.

3.1 Dialogue Generation

The sales agent uses OpenAI GPT-4o-mini[1], the same model deployed in our company's production system. To ensure diversity in sales strategies and interaction dynamics, we tested 13 agent configurations varying in sales strategy (phone recommendation assistance, cross-sell, up-sell), timing of proactive sales tactics, and whether these tactics were applied systematically or only to susceptible customers.

The user simulator is based on Google Gemini-1.5-Flash[2], selected for its different model family from the sales agent (avoiding model-specific biases), cost-effectiveness, and compliance with company approval requirements. The simulator is persona-driven, with validated adherence to assigned personas, goals, and scenarios, as described in our prior work (Gromada et al., 2025). Each persona represents a realistic customer archetype for our organization, defined by demographics (age, marital status, children, occupation), hobbies, budget constraints, product preferences, currently used telecom services, and tech proficiency. For each persona, two conversation goals and four scenario variants were defined, varying in levels of assertiveness and precision of needs. Personas, goals, and conversation scenarios were generated using Anthropic Claude 3.5 Sonnet v2[3] and are publicly available on our repository[4] (Gromada et al., 2025).

Each simulated customer adopts one such persona, goal, and scenario during the conversation, yielding 400 distinct customer configurations (100 personas × 1 of 2 possible goals × 4 scenarios). Combined with 13 agent configurations, we generated 5200 dialogues in total (400 per agent configuration). Conversations were terminated either after 70 turns (never reached in practice) or upon an explicit end signal from the simulator, triggered when the customer considered their goal reached or unachievable. Dialogue statistics are summarized in Table 1.

3.2 User Satisfaction Annotation

We annotated user satisfaction using an LLM-as-a-judge methodology. In our prior work (Gromada et al., 2025), we conducted a validation study comparing multiple commercial and open-source language models against human judgments on 104 simulated dialogues rated by three independent annotators. Among the evaluated models, Qwen-14B-1M[5] achieved the strongest alignment with the mean of human ratings among open-source models (Spearman ρ=0.78) and was therefore deployed to annotate the entire dataset (5200 dialogues).

Agent configs	13
Simulator configs	400 (100P × 1G × 4S)
Total dialogues	5200
Turns/dialogue	12.25 ± 2.78 (5–31)
Chars/turn	U: 226 ± 65, A: 707 ± 210

Table 1: Dialogue dataset. P: Personas, G: Goals, S: Scenarios, U: simulated User, A: Agent

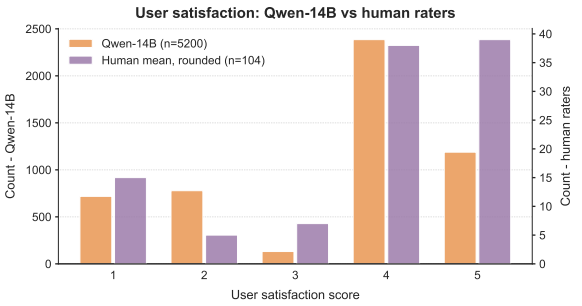

Figure 1: Distribution of satisfaction scores for Qwen annotations (n=5200) and human evaluations (n=104).

Satisfaction was rated on a 5-point scale from 1 (very dissatisfied) to 5 (very satisfied) by observers (not the simulated user) evaluating their perception of the customer's overall experience with the sales interaction, independently of purchase decisions. Figure 1 presents the distribution of satisfaction scores for both Qwen annotations (full dataset) and human evaluations (validation subset). The Qwen-annotated scores show a mean of 3.49 (SD=1.35) and median of 4, while human ratings average 3.71 (SD=1.36) with a median of 4, both indicating generally positive customer experiences with considerable variation. The distributions exhibit similar overall shapes with identical medians and comparable means, although human annotators assign the highest score at a higher rate than Qwen (37.50% vs. 22.85%), suggesting that the LLM is more conservative in assigning maximum ratings.

[1] https://developers.openai.com/api/docs/models/gpt-4o-mini

[2] https://deepmind.google/technologies/gemini/flash/

[3] https://www.anthropic.com/news/claude-3-5-sonnet

[4] https://huggingface.co/datasets/Orange/PersonasForSalesbot

[5] https://huggingface.co/Qwen/Qwen2.5-14B-Instruct-1M

4 Sales Actions Annotation Framework

We annotated each conversation with structured tags capturing sales agent actions (phone offers, cross-sells) and customer responses (acceptance, rejection, cancellation). This structured representation allows us to measure sales metrics such as acceptance or conversion rates, and to analyze the impact of dialogue strategies on purchase decisions.

4.1 Annotation Method

We evaluated multiple LLMs, i.e. commercial models from OpenAI, Google, and Anthropic, and open-source models from Qwen, Meta, and DeepSeek, with various prompts on 20 test dialogues. Three human annotators manually assessed outputs for accuracy and consistency. Based on annotation quality, cost, and response time, we selected Claude 3.5 Sonnet v2 with a decision-tree prompt (see Appendix A).

Sales tags were applied automatically to all 5200 dialogues. For 87 dialogues with formatting errors, we used Claude 4.5 Sonnet[6]. Tags were assigned at the turn level for explicit sales actions only, with an additional dialogue-level tag summarizing the final cart. Total annotation cost was $87.

We validated annotation consistency through automated logical checks across all dialogues: (1) acceptance counts cannot exceed corresponding offer counts, (2) category-level counts sum correctly to totals, (3) actions follow valid sequences (e.g., acceptances or cancellations require prior offers), and (4) final cart contents are consistent with accepted and cancelled items. Most dialogues passed these consistency checks; only 1.17% violated at least one constraint.

4.2 Sales Tags Taxonomy

Table 2 presents the complete sales tags structure, organized by actor and action type, with a practical application shown in Figure 2.

The sales agent taxonomy captures two offer types: phone-only and bundle (phone+plan). For cross-selling, we distinguish between bot-initiated (agent proactively suggests accessories, insurance, etc.) and user-initiated (customer asks about additional products) attempts, enabling analysis of proactive versus reactive sales strategies.

Customer sales actions include acceptance (explicit purchase commitment), rejection (declining an offer), and cancellation (withdrawing previous acceptance).

Sales Action	Tag Pattern
Sales Agent (turn-level)	
Offer	`[OFFER:PHONE:{...}]`
	`[OFFER:BUNDLE:{...}]`
Cross-sell	`[XS:B_INIT:{...}]`
	`[XS:U_INIT:{...}]`
Customer (turn-level)	
Accept	`[ACCEPT_OFFER:PHONE:{...}]`
	`[ACCEPT_OFFER:BUNDLE:{...}]`
	`[ACCEPT_XS:B_INIT:{...}]`
	`[ACCEPT_XS:U_INIT:{...}]`
Reject	`[REJECT_OFFER:PHONE:{...}]`
	`[REJECT_OFFER:BUNDLE:{...}]`
	`[REJECT_XS:B_INIT:{...}]`
	`[REJECT_XS:U_INIT:{...}]`
Cancel	`[CANCEL_OFFER_ACC:PHONE:{...}]`
	`[CANCEL_OFFER_ACC:BUNDLE:{...}]`
	`[CANCEL_XS_ACC:B_INIT:{...}]`
	`[CANCEL_XS_ACC:U_INIT:{...}]`
Dialogue-level summary	
Final Cart	`[FINAL_CART:{...}]`
	`[FINAL_CART:EMPTY]`

Table 2: Sales tags taxonomy. XS = CROSS_SELL, ACC = ACCEPTANCE, B_INIT = BOT_INITIATED, U_INIT = USER_INITIATED. Placeholders {...} for specific product names (phones, plans, accessories).

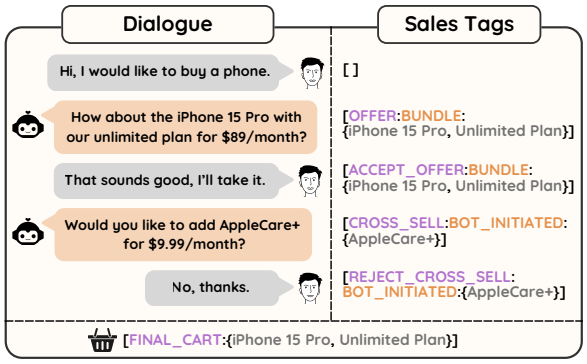

Figure 2: Sales tag annotation example.

4.3 Sales Tags Statistics

Table 3 presents the sales tags distribution across our dataset. Sales agents make offers in nearly all dialogues (99.12%) with an average of 3.86 offers per conversation, while cross-selling attempts occur in approximately half of conversations (51.10%). Customer acceptance rates are higher for primary offers (68.94%) than for cross-sells (30.96%), and rejection rates are significant (89.56% for offers), indicating iterative negotiation processes. Cancellations are rare (<5% of dialogues), suggesting strong commitment once

[6] https://www.anthropic.com/news/claude-sonnet-4-5

customers accept products. Extreme maximum values (e.g., 18 offers) represent outliers where agents presented multiple options within single turns.

Tag Category	% Dialogues	Mean±SD	Max
OFFER	99.12	3.86±1.93	18
ACCEPT_OFFER	68.94	0.79±0.62	4
REJECT_OFFER	89.56	2.11±1.46	14
CANCEL_OFFER	4.25	0.04±0.21	2
CROSS_SELL	51.10	1.02±1.32	10
ACCEPT_XS	30.96	0.42±0.73	6
REJECT_XS	16.96	0.24±0.60	5
CANCEL_XS	0.56	0.01±0.09	2

Table 3: Sales tag distribution across 5200 dialogues (per dialogue statistics). OFFER includes phone and bundle offers. CROSS_SELL (XS) includes bot-initiated and user-initiated cross-sells.

4.4 Sales Metrics

Based on the sales action annotations, we computed interpretable sales metrics across six categories (Table 4), enabling both predictive modeling of satisfaction and interpretable analysis of how sales behaviors impact customer satisfaction.

Category	Examples
Basic	Counts of offers, acceptances, rejections, cancellations
Cart	Cart completion rate, final cart size, abandonment rate
Conversion	Offer acceptance rates, cross-sell success rates, cancellation patterns
Flow	Negotiation patterns, sales density, customer journey progression
Strategy	Product mix ratios, initiative rates, recovery success
Timing	Turns to first offer, negotiation length, sales cycle duration

Table 4: Sales metrics categories with representative examples. Complete definitions in Appendix B.

5 Modeling User Satisfaction

We develop an interpretable metrics-based approach to predict customer satisfaction from sales dialogues and validate its effectiveness against a text-based transformer baseline.

5.1 Experimental Setup

5.1.1 Data split

We split our dataset of 5200 dialogues into training (3566, 68.6%), validation (765, 14.7%), and test (869, 16.7%) sets. All 104 human-evaluated dialogues were placed in the test set, to enable dual evaluation against both Qwen-14B and human satisfaction scores. Models were trained using Qwen-14B annotations as targets, with the validation set used for hyperparameter tuning and early stopping. Satisfaction score distributions are similar across splits, ensuring consistent evaluation conditions.

5.1.2 Metrics-Based Model

Approach Our primary approach uses interpretable features derived from sales action annotations (Section 4) enriched with dialogue structure features. This approach enables satisfaction prediction while providing interpretable insights into which sales behaviors influence customer experience, making it directly applicable to sales strategy optimization.

Model architecture and training We built an XGBoost regression model using 120 features: 91 sales-specific metrics derived from annotated tags and 29 dialogue structure features (e.g., turn counts, turn length statistics).

Given high feature dimensionality, we applied SHAP-based feature selection. A preliminary model estimated feature importance, and features were retained by thresholding these values to maximize validation performance, resulting in 91 selected features. The final model was trained on this subset with regularization and early stopping based on validation Mean Squared Error (MSE). XGBoost was configured with a maximum tree depth of 4, learning rate 0.05, row and feature subsampling ratios of 0.8, minimum child weight of 5, L1 regularization ($\alpha = 0.1$), and L2 regularization ($\lambda = 1.0$).

5.1.3 Text-Based Model

Approach We fine-tuned a transformer model to predict satisfaction scores directly from conversation text, without requiring annotation or feature engineering. This serves as our baseline.

Model architecture and training We fine-tuned ModernBERT[7] with a regression head to predict satisfaction scores from dialogue text. Each dialogue was formatted as a sequence of turns with speaker labels:

```
user:> Hi, I would like to buy a phone.
assistant:> How about the iPhone 15 Pro with our
     unlimited plan for $89/month?
user:> That sounds good, I'll take it.
```

[7] https://huggingface.co/answerdotai/ModernBERT-base

	Metrics-based Model		Text-based Model	
Metric	**Qwen-14B**	**Human**	**Qwen-14B**	**Human**
MSE ↓	0.24	0.34	0.14	0.29
MAE ↓	0.40	0.42	0.28	0.41
R^2	0.87	0.78	0.92	0.81
ρ	0.82	0.75	0.89	0.75

Table 5: Regression results by model and test set (Qwen-14B: n=869, Human: n=104). Metrics-based: XGBoost (91 features), Text-based: ModernBERT. Models trained with early stopping on validation set (n=765).

The model was trained with a learning rate of 2e-5 for up to 10 epochs with early stopping based on validation MSE. To handle varying dialogue lengths, we set maximum input length to 8192 tokens with dynamic padding (padding each batch to its longest sequence), which was sufficient to avoid truncating any dialogues in our dataset.

5.2 Results

Quantitative Results We evaluated both approaches using MSE, Mean Absolute Error (MAE), coefficient of determination (R^2), and Spearman's rank correlation coefficient (ρ). Evaluation was performed on the Qwen-14B annotated test set (n=869) and the human-annotated subset (n=104). Table 5 presents the results.

The metrics-based XGBoost model achieves strong performance on Qwen-14B predictions (R^2=0.87, ρ=0.82), while being more computationally efficient and providing interpretable insights applicable to sales strategy optimization. The text-based ModernBERT model achieves even stronger performance (R^2=0.92, ρ=0.89), demonstrating that transformer-based models can learn satisfaction patterns directly from text without requiring additional annotations or feature engineering.

Performance on the human test set remains high for both approaches, with identical Spearman correlation (ρ=0.75) indicating similar ranking ability compared to human judges, and comparable R^2 scores of 0.78 and 0.81 for the metrics-based and text-based models, respectively.

Notably, the performance gap between Qwen-14B and human evaluations is smaller for the metrics-based approach (ΔR^2=0.09) compared to the text-based approach (ΔR^2=0.11), suggesting competitive performance despite the text-based model's higher overall accuracy.

Error Analysis Error analysis further confirms the overall reliability of both models: significant mispredictions (prediction error exceeding 1.5 points, a substantial deviation on the 1–5 scale) occurred in only 4 and 3 of 869 test cases (<1%) for the metrics-based XGBoost and text-based ModernBERT models, respectively.

For metrics-based model, 3 out of 4 mispredictions were overestimations, where customers initially accepted an offer but later abandoned the interaction because the assistant ignored their constraints. The single underestimation occurred in a dialogue without a purchase, where the assistant could not find a suitable product within the customer's budget but still provided helpful guidance.

For text-based model, all 3 mispredictions were overestimations, each involving an assistant error that the model failed to capture, such as recommending the wrong product or suggesting an incompatible accessory.

6 Understanding User Satisfaction

To understand which behaviors drive customer satisfaction, we analyzed feature importance using SHAP values from the metrics-based model. SHAP values explain how each feature contributes to individual predictions: positive SHAP values indicate increased predicted satisfaction, while negative values indicate decreased satisfaction.

We first identify the most important features, then examine dependence patterns to understand how specific feature values and their interactions affect satisfaction.

6.1 Feature Importance Overview

Figure 3 shows the top 20 most important features based on SHAP values. Each point represents one dialogue from the test set, with color indicating feature value (red = high, blue = low) and position on the x-axis showing the SHAP value (impact on satisfaction). Table 6 provides definitions and formulas for these features, ordered by importance.

The color-position patterns reveal each feature's directional impact on satisfaction. For net conversion efficiency rate (rank 1), red points (high values) cluster toward the right, indicating positive contribution to satisfaction, while blue points (low values) cluster left. In contrast, offer rejection rate (rank 9) or mean user turn length (rank 11) shows the opposite pattern: red points shift left, reflecting its negative impact on satisfaction.

The analysis reveals that user satisfaction is primarily driven by outcome-oriented features. The

Rank	Feature	Definition	Category
1	Net conversion efficiency rate	$(A - C)$ per dialogue turn: $(A - C)/T$	Flow
2	Negotiation success rate	Proportion of offer–response cycles ending in acceptance	Flow
3	Num final cart items	Count of items in the final cart (F)	Cart-specific
4	Nbr turns	Total number of dialogue turns (T)	Dialogue structure
5	Customer decisiveness score	$A/(A + R + C)$	Flow
6	Post-rejection offer rate	Proportion of rejections (R) followed by a new offer	Strategy
7	Net overall conversion rate	Net acceptances $(A - C)$ relative to all sales actions $(O + CS)$	Conversion
8	Adaptation score	Proportion of rejections or cancellations $(R + C)$ followed by a new offer	Strategy
9	Offer rejection rate	Proportion of offers (O) explicitly rejected (R_O): R_O/O	Conversion
10	Phone-to-bundle ratio	P/B	Strategy
11	Mean user turn length	Average number of words per user turn	Dialogue structure
12	Stdev user turn length	Standard deviation of user turn lengths (words)	Dialogue structure
13	Rejection recovery success rate	Proportion of rejections (R) eventually followed by an acceptance	Strategy
14	Stdev turn length	Standard deviation of turn lengths across all speakers (words)	Dialogue structure
15	Has final purchase	$\mathbf{1}[F > 0]$: 1 if the final cart is non-empty	Cart-specific
16	Avg offer spacing	Average number of turns between consecutive offers	Timing
17	Conversion efficiency rate	Total acceptances (A) per dialogue turn: A/T	Flow
18	Num reject offer total	$R_O = R_P + R_B$: total count of offer rejections	Basic
19	Purchase path complexity score	$(C + R)/A$	Flow
20	Dialogue duration	Total number of dialogue turns (T)	Dialogue structure

Table 6: Top 20 features ranked by mean absolute SHAP value on Qwen-14B test set. A = total acceptances, C = total cancellations, R = total rejections, O = total offers, CS = total cross-sells, T = total dialogue turns, F = final cart items, P = phone-only offers, B = bundle offers. See Appendix B for complete definitions.

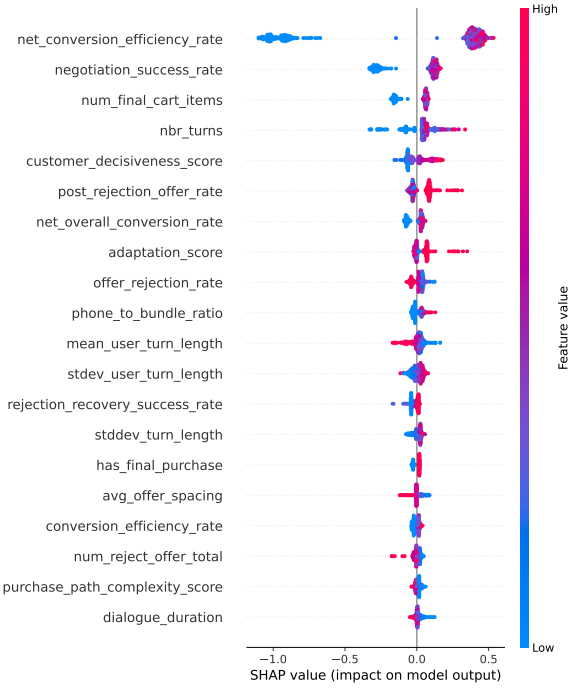

Figure 3: Top 20 features ranked by mean absolute SHAP value on the Qwen-14B test set.

most important predictor is net conversion efficiency rate (rank 1), followed by negotiation success rate (rank 2) and number of final cart items (rank 3), all representing concrete sales outcomes.

Beyond these outcome measures, dialogue structure features play an important secondary role, particularly the number of turns (rank 4). This is followed by flow metrics like customer decisiveness score (rank 5) and strategy metrics such as post-rejection offer rate (rank 6) and adaptation score (rank 8). These patterns suggest that while concrete outcomes matter most, both conversation structure and sales strategy contribute meaningfully to user satisfaction. Notably, most features in the lower half of the ranking show minimal individual impact on satisfaction.

The importance of dialogue structure and strategy features is further confirmed by their representation in the top 20 predictors (Table 6), which comprise 5 flow metrics, 5 dialogue structure features, 4 strategy metrics, 2 conversion and 2 cart-specific features, and 1 timing and 1 basic sales metric. This distribution indicates that while outcomes are key drivers, the conversation process and sales strategies also play a significant role in determining satisfaction.

6.2 Feature Dependence analysis

Figure 4 presents SHAP dependence plots for the top 4 features, revealing the relationship between each feature's value (x-axis) and its contribution to predicted satisfaction (SHAP value, y-axis). Point color (red = high, blue = low) encodes a secondary feature (an interaction variable) that modulates this relationship. Interaction variables are automatically selected by SHAP based on interaction strength: adaptation score for net conversion efficiency rate (rank 1), post-rejection offer rate for negotiation success rate (rank 2), average offers per rejection for number of final cart items (rank 3), and has final purchase for number of turns (rank 4).

The top 3 features show a threshold pattern dis-

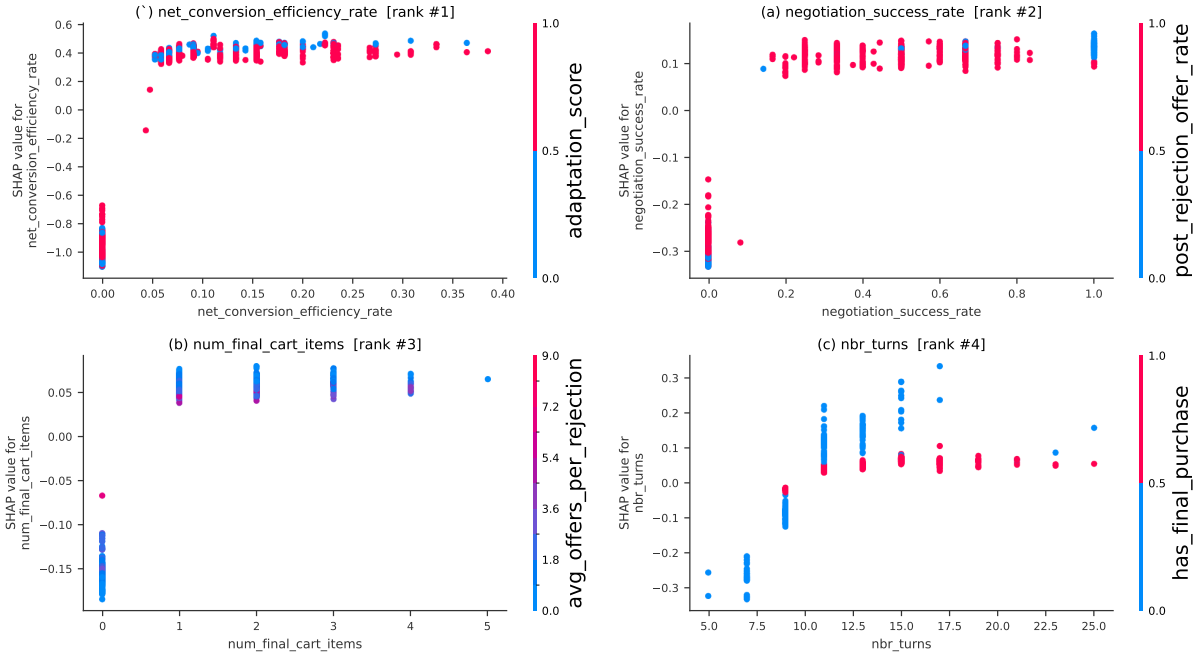

Figure 4: SHAP dependence plots for the top 4 features on Qwen-14B test set. Color indicates interaction variable values: red = high, blue = low.

tinguishing zero from non-zero values. Zero values produce negative SHAP contributions, while any positive value leads to much higher SHAP values.

This effect is best illustrated by net conversion efficiency rate (rank 1), which reflects whether any offered product was purchased. When the rate is zero, SHAP drops to around -1.0, indicating a strong negative contribution to the predicted satisfaction. Once any conversion appears, the effect peaks at +0.4 and remains stable regardless of any further increase. The interaction with adaptation score shows no clear systematic tendency, as both high (red) and low (blue) adaptation scores appear across all conversion rate levels.

The same threshold pattern holds for negotiation success rate (rank 2) and number of final cart items (rank 3), though with weaker effects. For negotiation success rate, a zero rate produces SHAP values of -0.25 to -0.35, while any non-zero success rate raises SHAP to around +0.10. The interaction variable (post-rejection offer rate) shows that dialogues where at least one negotiation succeeded also had the system continue to make offers after rejection, suggesting that persistence in negotiation may jointly contribute to satisfaction. For number of final cart items, zero items produce SHAP values between -0.20 and -0.10, while even one item leads to an increase to +0.05, with no further gains from additional items. The interaction variable (average offers per rejection) shows no clear systematic pattern.

In contrast, the pattern for number of turns (rank 4) is more complex. The interaction variable (final purchase) reveals that the same turn count can have a different effect depending on whether a purchase occurred. At low turn counts (5-9), conversations without a final purchase (blue points) show strongly negative SHAP values down to -0.33, representing short failed conversations. At higher turn counts (>11), these failed conversations (blue points) shift towards neutral to positive SHAP values, suggesting that longer dialogues, despite not leading to a purchase, may still reflect customer engagement. Conversations with a final purchase (red points) contribute positively across almost all turn counts, confirming that the effect of conversation length on satisfaction depends on whether it ended with a purchase.

6.3 Insights for Sales Strategy

The findings from feature importance and dependence analyses suggest several practical insights for sales strategy.

First, satisfaction is primarily driven by customers being able to purchase a suitable product. Sales agents should prioritize offer quality and variety over quantity, since any successful conversion strongly improves satisfaction, while zero conver-

sion is the strongest negative signal observed.

Second, persistence in negotiation matters. Continuing to make offers after rejection is associated with higher negotiation success rates and higher satisfaction, suggesting that agents should not abandon conversations after an initial refusal.

Finally, conversation length effects depend on context. Short failed conversations score lowest, but longer conversations without a purchase can still maintain neutral to positive satisfaction, indicating that engagement itself has value even without a purchase.

These findings confirm that while concrete outcomes are the most critical drivers of satisfaction, the quality of the conversation process and sales strategies employed also play a significant role in customer experience.

7 Conclusions

In this work, we presented an interpretable approach to modeling user satisfaction in sales dialogues, grounded in a sales action annotation framework and 120 derived interpretable sales metrics across six categories: conversion rates, timing patterns, negotiation dynamics, sales strategies, conversation flow, and cart outcomes.

We evaluated our approach on 5200 simulated phone sales dialogues, showing that our metrics-based XGBoost model achieves competitive performance with a transformer-based text-embedding baseline (ModernBERT) while remaining fully interpretable.

SHAP-based analysis revealed that satisfaction depends primarily on outcome-oriented features, with net conversion efficiency rate, negotiation success rate, and number of final cart items being the most influential, while sales strategies and conversation process also contribute meaningfully.

These findings provide actionable insights for sales strategy and dialogue system design, demonstrating that interpretable objective metrics can both predict satisfaction effectively and reveal the behavioral drivers behind it.

8 Future Work

Future work should validate this framework on real customer conversations to assess whether the observed patterns and feature importance rankings generalize beyond simulated dialogues to actual sales interactions.

Additionally, our dataset of LLM-based sales action annotations could serve as training data for a supervised annotation model, enabling automatic annotation without LLM inference while requiring systematic human evaluation to ensure quality.

Limitations

Simulated dialogues Our framework was developed on simulated dialogues, which enable controlled experimentation but may not fully capture the complexity of real customer interactions.

LLM-based satisfaction annotation Satisfaction scores were generated using an LLM-as-a-judge approach (Qwen-14B), selected for its high correlation with human raters (Spearman ρ=0.78), though a performance gap remains (ΔR^2=0.09–0.11 vs. human test sets). Additionally, only 104 of 5200 dialogues (2%) were human-annotated, limiting the reliability of the human-based evaluation. Qwen also shows a conservative tendency in assigning maximum satisfaction ratings compared to human annotators (22.85% vs. 37.50%), which may introduce a slight bias in models trained on these annotations, potentially leading them to underestimate satisfaction in the most positive interactions.

Observer-based satisfaction assessment Both human annotators and Qwen-14B assessed satisfaction from dialogue transcripts rather than from the simulated customer's perspective, missing post-conversation effects such as delayed negative reactions to persuasive tactics.

Sales action annotation Sales action annotation framework relies on Claude 3.5 Sonnet v2, selected based on preliminary tests on 20 dialogues validated by 3 human annotators, but without systematic evaluation or inter-annotator agreement. LLM-based annotations may contain logical inconsistencies, identified in 1.17% of dialogues. While relatively low, such errors could impact the reliability of the derived metrics. For future annotation efforts, we recommend conducting a systematic evaluation of different LLM and prompt combinations to identify the most reliable configuration.

Model selection This work explores only two model families: XGBoost for the metrics-based approach and ModernBERT for the text-based approach. Other interpretable methods and transformer architectures were not explored, and alternative models may yield better performance.

References

Anouck Braggaar, Christine Liebrecht, Emiel van Miltenburg, and Emiel Krahmer. 2023. Evaluating task-oriented dialogue systems: A systematic review of measures, constructs and their operationalisations. *arXiv preprint arXiv:2312.13871*.

Yang Cheng and Hua Jiang. 2020. How do ai-driven chatbots impact user experience? examining gratifications, perceived privacy risk, satisfaction, loyalty, and continued use. *Journal of Broadcasting & Electronic Media*, 64(4):592–614.

Kyunghyun Cho, Bart Van Merriënboer, Çağlar Gulçehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using rnn encoder–decoder for statistical machine translation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1724–1734.

Sam Davidson, Salvatore Romeo, Raphael Shu, James Gung, Arshit Gupta, Saab Mansour, and Yi Zhang. 2023. User simulation with large language models for evaluating task-oriented dialogue. *arXiv preprint arXiv:2309.13233*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pages 4171–4186.

Yue Feng, Yunlong Jiao, Animesh Prasad, Nikolaos Aletras, Emine Yilmaz, and Gabriella Kazai. 2023. Schema-guided user satisfaction modeling for task-oriented dialogues. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2079–2091.

Justyna Gromada, Alicja Kasicka, Ewa Komkowska, Lukasz Krajewski, Natalia Krawczyk, Morgan Veyret, Bartosz Przybył, Lina M. Rojas-Barahona, and Michał K. Szczerbak. 2025. Evaluating conversational agents with persona-driven user simulations based on large language models: A sales bot case study. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 230–245, Suzhou (China). Association for Computational Linguistics.

Taaha Kazi, Ruiliang Lyu, Sizhe Zhou, Dilek Hakkani-Tür, and Gokhan Tur. 2024. Large language models as user-agents for evaluating task-oriented-dialogue systems. In *2024 IEEE Spoken Language Technology Workshop (SLT)*, pages 913–920. IEEE.

Bill Yuchen Lin, Yuntian Deng, Khyathi Chandu, Abhilasha Ravichander, Valentina Pyatkin, Nouha Dziri, Ronan Le Bras, and Yejin Choi. 2025. Wildbench: Benchmarking llms with challenging tasks from real users in the wild. 2025:47852–47870.

Yu-li Liu, Bo Hu, Wenjia Yan, and Zhi Lin. 2023. Can chatbots satisfy me? a mixed-method comparative study of satisfaction with task-oriented chatbots in mainland china and hong kong. *Computers in Human Behavior*, 143:107716.

Scott M. Lundberg and Su-In Lee. 2017. A unified approach to interpreting model predictions. In *Neural Information Processing Systems*.

Miguel Orden-Mejia, Mauricio Carvache-Franco, Assumpcio Huertas, Orly Carvache-Franco, and Wilmer Carvache-Franco. 2025. The role of ai-based destination chatbots in satisfaction, continued usage intention, and visit intention: A study from quito, ecuador. *International Journal of Human–Computer Interaction*, 41(15):9384–9399.

Baolin Peng, Chunyuan Li, Zhu Zhang, Chenguang Zhu, Jinchao Li, and Jianfeng Gao. 2021. Raddle: An evaluation benchmark and analysis platform for robust task-oriented dialog systems. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4418–4429.

Clemencia Siro, Mohammad Aliannejadi, and Maarten de Rijke. 2022. Understanding user satisfaction with task-oriented dialogue systems. In *Proceedings of the 45th International ACM SIGIR conference on research and development in information retrieval*, pages 2018–2023.

Andreas Stolcke, Klaus Ries, Noah Coccaro, Elizabeth Shriberg, Rebecca A. Bates, Dan Jurafsky, Paul A. Taylor, Rachel Martin, Carol Van Ess-Dykema, and Marie W. Meteer. 2000. Dialogue act modeling for automatic tagging and recognition of conversational speech. *Computational Linguistics*, 26:339–373.

Weiwei Sun, Shuo Zhang, Krisztian Balog, Zhaochun Ren, Pengjie Ren, Zhumin Chen, and Maarten De Rijke. 2021. Simulating user satisfaction for the evaluation of task-oriented dialogue systems. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2499–2506.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Marilyn Walker, Diane Litman, Candace A Kamm, and Alicia Abella. 1997. Paradise: A framework for evaluating spoken dialogue agents. pages 271–280.

A Prompt used to annotate dialogues with sales tags

```
# Dialogue Annotation Task - Decision Tree Approach
You are an expert dialogue analyst.
You will evaluate dialogues between our customer and bot selling phones with plans, accessories, and
    additional services.
**Your task:** Annotate each utterance by following a decision tree to determine the appropriate tags
    .
---
## Core Concepts
### **Sales Strategies**
- **Cross-selling:** Suggesting complementary products (accessories, services, insurance) that go
    well with the main purchase
  - Example: Customer buys phone -> Bot recommends case, screen protector, or insurance

### **Key Distinctions**
- **OFFER** = Phone or bundle presentation that matches customer needs (always bot-initiated)
- **CROSS_SELL** = Accessories, services, or additional products (can be user-initiated or bot-
    initiated)
- **USER_INITIATED** = Bot responds to user's explicit request for that product category
- **BOT_INITIATED** = Bot proactively suggests without user asking

**Important:** Once a product category is initiated, it stays the same type throughout the
    conversation, even if user asks for variants or alternatives within that category.

### **Product Formatting**
- Copy product names exactly: `**Product Name**`
- Bundles use `#`: `**Phone**#**Plan**`
- Multiple products = multiple tags

### **Bundle Formatting Rules**
**When bot offers multiple products:**
- **Alternative options (OR)** -> Separate tags for each
  - "Samsung A56 with Plan M OR Motorola edge 50 with Plan S"
  - Tags: `[OFFER:BUNDLE:**Samsung A56**#**Plan M**]` + `[OFFER:BUNDLE:**Motorola edge 50**#**Plan S
    **]`
- **Complementary products** -> Separate tags with appropriate types
  - "Samsung A56 with Plan M, plus memory card for 99 PLN"
  - Tags: `[OFFER:BUNDLE:**Samsung A56**#**Plan M**]` + `[CROSS_SELL:BOT_INITIATED:**Memory Card**]`
- **Bundled together (AND)** -> One tag with `#` separators
  - "Samsung A56 with Plan M and Plan XS together"
  - Tag: `[OFFER:BUNDLE:**Samsung A56**#**Plan M**#**Plan XS**]`
---
## Decision Tree
### **Step 1: Who is speaking?**
**BOT** -> Go to Step 2
**USER** -> Go to Step 5
---
### **Step 2: Does bot present a product?**
**YES** -> Go to Step 3
**NO** -> No tag, move to next utterance
---
### **Step 3: Is this a phone/bundle or accessory/service?**
**PHONE or BUNDLE** -> Go to Step 4A
**ACCESSORY or SERVICE** -> Go to Step 4B
---
### **Step 4A: Tag the OFFER**
**Phone only:** `[OFFER:PHONE:**phone name**]`
**Phone + plan(s):** `[OFFER:BUNDLE:**phone**#**plan1**#**plan2**...]`

**Multiple alternatives?** Create separate tags for each.
**Examples of accessories/services (NOT offers):**
- Cases, chargers, headphones, screen protectors
- Insurance, warranties, cloud storage

**Done.** Move to next utterance.
---
### **Step 4B: Tag the CROSS_SELL**
**Format:** `[CROSS_SELL:???:**product**]`
**Question:** Who initiated this product category?
```

```
**Important rules:**
1. **Initiation happens when category is FIRST mentioned** - if user asked about "smartwatches" in
    Turn 6, and bot presents specific smartwatch models in Turn 9, it's USER_INITIATED
2. **Look back through conversation** - user might have mentioned the category several turns earlier
3. **Once initiated, stays that type** - all products in that category maintain the same initiation
    type

**Determine initiation type:**
- **USER_INITIATED:** User explicitly asked for this category (in current or ANY earlier turn) -> `[
    CROSS_SELL:USER_INITIATED:**product**]`
- **BOT_INITIATED:** Bot suggested without user asking -> `[CROSS_SELL:BOT_INITIATED:**product**]`

**Examples:**
- Turn 5 User: "Could I add a smartwatch?"
- Turn 6 Bot: [discusses other things]
- Turn 8 User: "What smartwatches do you have?"
- Turn 9 Bot: "We have **Galaxy Watch5** and **Galaxy Watch4**"
  - Tags: `[CROSS_SELL:USER_INITIATED:**Galaxy Watch5**]` + `[CROSS_SELL:USER_INITIATED:**Galaxy
    Watch4**]`
  - Reason: User initiated the smartwatch category in Turn 5

**Multiple products?** Create separate tags for each.
**Done.** Move to next utterance.
---
### **Step 5: Does user accept, reject, or cancel?**

**Just asking questions/discussing** -> No tag, move to next utterance
**ACCEPT** -> Go to Step 6
**REJECT** -> Go to Step 7
**CANCEL previous acceptance** -> Go to Step 8
---
### **Step 6: Tag the ACCEPTANCE**
**What is user accepting?**

#### **Accepting OFFER:**
- Phone only? `[ACCEPT_OFFER:PHONE:**phone**]`
- Bundle? `[ACCEPT_OFFER:BUNDLE:**phone**#**plan**]`

#### **Accepting CROSS_SELL:**
- Format: `[ACCEPT_CROSS_SELL:???:**product**]`
- Match initiation type from original cross-sell (USER_INITIATED or BOT_INITIATED)

**Explicit acceptance phrases:**
- "I'll take it/X"
- "I want X"
- "Yes" (in direct response to "would you like X?")
- "Let's go with X"
- "I choose X"
- "Perfect, I'll buy it"
- Proceeding to purchase steps (delivery, payment, confirmation)

**NOT acceptance (just interest/preference):**
- "I like X"
- "X sounds good/interesting/better"
- "X looks nice"
- "Tell me more about X"
- "Okay" or "Hmm" (without commitment)
- Continuing to ask questions or negotiate after expressing interest

**Key rule:** Only tag acceptance when user **commits to purchase**, not when they express interest,
    preference, or continue deliberating.

**Examples:**
- "The S25 sounds much better. Let me think about the plans." -> NO TAG (still deliberating)
- "The S25 sounds much better. I'll take it." -> ACCEPT (clear commitment)
- "I like the Samsung. What plans do you have?" -> NO TAG (asking questions)
- "I like the Samsung. I want to buy it." -> ACCEPT (commitment)

**Done.** Move to next utterance.
```

```
---
### **Step 7: Tag the REJECTION**
**Important:** Only tag EXPLICIT rejections ("no", "I don't want X", "not interested").
Do NOT tag when user simply chooses one option from alternatives.

**What is user rejecting?**
#### **Rejecting OFFER:**
- Phone only? `[REJECT_OFFER:PHONE:**phone**]`
- Bundle? `[REJECT_OFFER:BUNDLE:**phone**#**plan**]`

**Partial bundle rejection:** If user likes phone but not plan (or vice versa), reject the ENTIRE
    bundle.

#### **Rejecting CROSS_SELL:**
- Format: `[REJECT_CROSS_SELL:???:**product**]`
- Match initiation type from original cross-sell

**Multiple rejections in one utterance:**
- Tag EACH rejection separately, even with general statements like "all too expensive"
- Example: User says "both are too expensive" about 2 bundles -> create 2 rejection tags

**Done.** Move to next utterance.
---
### **Step 8: Tag the CANCELLATION**
**When to tag:**
- User explicitly cancels: "I don't want X anymore"
- User switches products: "I'll take Y instead of X"
- User modifies choice: "Can I get Plan L instead of Plan M?"
**Important:**
- ANY product can be cancelled
- Don't assume mutual exclusivity - user might buy multiple items
- Only tag when user explicitly cancels or switches

**What type of acceptance is being cancelled?**
#### **Cancelling OFFER acceptance:**
- Phone only? `[CANCEL_OFFER_ACCEPTANCE:PHONE:**phone**]`
- Bundle? `[CANCEL_OFFER_ACCEPTANCE:BUNDLE:**phone**#**plan**]`

#### **Cancelling CROSS_SELL acceptance:**
- Format: `[CANCEL_CROSS_SELL_ACCEPTANCE:???:**product**]`
- Match initiation type from original acceptance

**Note:** Cancellation and new acceptance can occur in same utterance - tag both.
**Done.** Move to next utterance.
---
## Output Format
For each utterance in the dialogue, provide:
**Turn X - [Speaker]:** [utterance text]
[TAG_1]
[TAG_2]
...
If no tags: **Turn X - [Speaker]:** [utterance text]
**After all utterances, add a final cart summary:**
`[FINAL_CART:**product1**|**product2**|**product3**...]` or `[FINAL_CART:EMPTY]`
---
## Final Cart Summary
After annotating all utterances, add a final cart summary on a new line:
**Format:** `[FINAL_CART:**product1**|**product2**|**product3**...]`

**Rules:**
- Use `|` to separate products
- Include bundles as `**phone**#**plan**`
- List phones/bundles first, then accessories/services
- Only include products the user **explicitly accepted** (tagged with ACCEPT_OFFER or
    ACCEPT_CROSS_SELL)
- Do NOT include products that were accepted then cancelled
- If user doesn't purchase anything: `[FINAL_CART:EMPTY]`

**Examples:**
```
```
```

[FINAL_CART:**Samsung Galaxy S24**#**Plan M**|**Mobile Insurance**|**Phone Case**]
```
```
[FINAL_CART:**iPhone 15 256GB**#**Plan L**]
```
```
[FINAL_CART:**Motorola edge 50**#**Plan S**|**Motorola edge 50**#**Plan XS**|**Memory Card 128GB**]
```
```
[FINAL_CART:EMPTY]
```
---
## Examples
### Example 1: Simple Offer & Accept
**Turn 0 - User:** I need a Samsung phone
**Turn 1 - Bot:** I recommend **Samsung Galaxy A56** with **Plan M** for 2149 PLN
[OFFER:BUNDLE:**Samsung Galaxy A56**#**Plan M**]
**Turn 2 - User:** I'll take it
[ACCEPT_OFFER:BUNDLE:**Samsung Galaxy A56**#**Plan M**]
[FINAL_CART:**Samsung Galaxy A56**#**Plan M**]
---
### Example 2: Multiple Offers & Cross-sell
**Turn 0 - User:** I need a phone around 2000 PLN
**Turn 1 - Bot:** I have three options:
- **Samsung A56** with **Plan M** for 1999 PLN
- **Motorola Edge 50** with **Plan L** for 2299 PLN
- **iPhone 15** with **Plan S** for 2499 PLN
[OFFER:BUNDLE:**Samsung A56**#**Plan M**]
[OFFER:BUNDLE:**Motorola Edge 50**#**Plan L**]
[OFFER:BUNDLE:**iPhone 15**#**Plan S**]
**Turn 2 - User:** The iPhone is too expensive
[REJECT_OFFER:BUNDLE:**iPhone 15**#**Plan S**]
**Turn 3 - Bot:** Would you prefer the **Samsung A56** or **Motorola Edge 50**?
**Turn 4 - User:** I'll take the Motorola
[ACCEPT_OFFER:BUNDLE:**Motorola Edge 50**#**Plan L**]
**Turn 5 - Bot:** Excellent choice! Would you like **Mobile Insurance** for 25 PLN/month?
[CROSS_SELL:BOT_INITIATED:**Mobile Insurance**]
**Turn 6 - User:** Yes, please
[ACCEPT_CROSS_SELL:BOT_INITIATED:**Mobile Insurance**]
[FINAL_CART:**Motorola Edge 50**#**Plan L**|**Mobile Insurance**]
---
### Example 3: Negotiation Rounds
**Turn 0 - User:** I need a budget phone
**Turn 1 - Bot:** I recommend **Samsung A56** for 1999 PLN
[OFFER:PHONE:**Samsung A56**]
**Turn 2 - User:** Too expensive
[REJECT_OFFER:PHONE:**Samsung A56**]
**Turn 3 - Bot:** How about **Motorola Edge 50** for 1499 PLN?
[OFFER:PHONE:**Motorola Edge 50**]
**Turn 4 - User:** Perfect, I'll take it
[ACCEPT_OFFER:PHONE:**Motorola Edge 50**]
[FINAL_CART:**Motorola Edge 50**]
---
### Example 4: Mind Change
**Turn 0 - User:** I'll take **iPhone 15 128GB** with **Plan M**
[ACCEPT_OFFER:BUNDLE:**iPhone 15 128GB**#**Plan M**]
**Turn 1 - Bot:** Great choice!
**Turn 2 - User:** Actually, do you have the 256GB version?
**Turn 3 - Bot:** Yes! **iPhone 15 256GB** with **Plan M** for 4299 PLN
[OFFER:BUNDLE:**iPhone 15 256GB**#**Plan M**]
**Turn 4 - User:** I'll take that instead
[CANCEL_OFFER_ACCEPTANCE:BUNDLE:**iPhone 15 128GB**#**Plan M**]
[ACCEPT_OFFER:BUNDLE:**iPhone 15 256GB**#**Plan M**]
[FINAL_CART:**iPhone 15 256GB**#**Plan M**]
---
### Example 5: Multiple Rejections
**Turn 0 - User:** What phones do you have?
**Turn 1 - Bot:** I have **Samsung A56** with **Plan M** for 2149 PLN or **Motorola edge 50** with **
    Plan S** for 1580 PLN
[OFFER:BUNDLE:**Samsung A56**#**Plan M**]
```

```
[OFFER:BUNDLE:**Motorola edge 50**#**Plan S**]
**Turn 2 - User:** Both are too expensive
[REJECT_OFFER:BUNDLE:**Samsung A56**#**Plan M**]
[REJECT_OFFER:BUNDLE:**Motorola edge 50**#**Plan S**]
**Turn 3 - Bot:** I understand. Let me check other options for you.
[FINAL_CART:EMPTY]
---
### Example 6: User-Initiated Cross-sell
**Turn 0 - User:** I'll take the **Samsung A56**
[ACCEPT_OFFER:PHONE:**Samsung A56**]
**Turn 1 - Bot:** Great choice!
**Turn 2 - User:** Do you have phone cases?
**Turn 3 - Bot:** Yes! **Silicone Case** for 79 PLN
[CROSS_SELL:USER_INITIATED:**Silicone Case**]
**Turn 4 - User:** Do you have leather?
**Turn 5 - Bot:** Yes! **Leather Case** for 129 PLN
[CROSS_SELL:USER_INITIATED:**Leather Case**]
**Turn 6 - User:** I'll take the leather one
[ACCEPT_CROSS_SELL:USER_INITIATED:**Leather Case**]
[FINAL_CART:**Samsung A56**|**Leather Case**]
---
### Example 7: Choosing Between Alternatives (No Rejection Tag)
**Turn 0 - User:** I need a new phone
**Turn 1 - Bot:** I have **Phone A** for 1500 PLN or **Phone B** for 2000 PLN
[OFFER:PHONE:**Phone A**]
[OFFER:PHONE:**Phone B**]
**Turn 2 - User:** I'll take Phone A
[ACCEPT_OFFER:PHONE:**Phone A**]
**Note:** Choosing Phone A does NOT mean rejecting Phone B. Only tag the acceptance.
[FINAL_CART:**Phone A**]
---
### Example 8: Buying Multiple Products
**Turn 0 - User:** I need two phones - one for me and one for my daughter
**Turn 1 - Bot:** Great! What are you looking for?
**Turn 2 - User:** I want a Samsung with a good plan, and something cheaper for my daughter
**Turn 3 - Bot:** For you, I recommend **Samsung Galaxy S24** with **Plan L** for 2499 PLN. For your
    daughter, **Motorola edge 50** with **Plan S** for 1580 PLN
[OFFER:BUNDLE:**Samsung Galaxy S24**#**Plan L**]
[OFFER:BUNDLE:**Motorola edge 50**#**Plan S**]
**Turn 4 - User:** Perfect, I'll take both
[ACCEPT_OFFER:BUNDLE:**Samsung Galaxy S24**#**Plan L**]
[ACCEPT_OFFER:BUNDLE:**Motorola edge 50**#**Plan S**]
**Turn 5 - Bot:** Excellent! Would you like to add insurance to both phones?
[CROSS_SELL:BOT_INITIATED:**Mobile Insurance**]
[CROSS_SELL:BOT_INITIATED:**Mobile Insurance**]
**Turn 6 - User:** Yes, add insurance for both
[ACCEPT_CROSS_SELL:BOT_INITIATED:**Mobile Insurance**]
[ACCEPT_CROSS_SELL:BOT_INITIATED:**Mobile Insurance**]
**Note:** User bought multiple products without cancelling - tag all acceptances.
[FINAL_CART:**Samsung Galaxy S24**#**Plan L**|**Motorola edge 50**#**Plan S**|**Mobile Insurance**|**
    Mobile Insurance**]
---
### Example 9: Plan Modification
**Turn 0 - User:** I want the **Samsung Galaxy S24** with a good plan
**Turn 1 - Bot:** **Samsung Galaxy S24** with **Plan L** for 2499 PLN
[OFFER:BUNDLE:**Samsung Galaxy S24**#**Plan L**]
**Turn 2 - User:** I like the phone but **Plan L** is too expensive. What other plans do you have?
[REJECT_OFFER:BUNDLE:**Samsung Galaxy S24**#**Plan L**]
**Turn 3 - Bot:** I can offer **Samsung Galaxy S24** with **Plan M** for 2299 PLN or with **Plan S**
    for 2099 PLN
[OFFER:BUNDLE:**Samsung Galaxy S24**#**Plan M**]
[OFFER:BUNDLE:**Samsung Galaxy S24**#**Plan S**]
**Turn 4 - User:** I'll take it with **Plan M**
[ACCEPT_OFFER:BUNDLE:**Samsung Galaxy S24**#**Plan M**]
**Note:** Rejecting part of bundle = rejecting entire bundle. New plan = new bundle offer.
[FINAL_CART:**Samsung Galaxy S24**#**Plan M**]
---
### Example 10: Phone-Only Purchase
**Turn 0 - User:** I just need a phone, no plan
**Turn 1 - Bot:** We have **Samsung Galaxy A56** for 1999 PLN
```

```
[OFFER:PHONE:**Samsung Galaxy A56**]
**Turn 2 - User:** I'll take it
[ACCEPT_OFFER:PHONE:**Samsung Galaxy A56**]
**Turn 3 - Bot:** Great! Would you like to add a **Phone Case** for 79 PLN?
[CROSS_SELL:BOT_INITIATED:**Phone Case**]
**Turn 4 - User:** Yes, I'll take the case
[ACCEPT_CROSS_SELL:BOT_INITIATED:**Phone Case**]
**Note:** Phone-only purchase (no bundle) with successful cross-sell.
[FINAL_CART:**Samsung Galaxy A56**|**Phone Case**]
---
### Example 11: Interest vs. Acceptance
**Turn 0 - User:** I need a new Samsung phone
**Turn 1 - Bot:** I recommend **Samsung Galaxy A56** for 1999 PLN
[OFFER:PHONE:**Samsung Galaxy A56**]
**Turn 2 - User:** Hmm, that's interesting. Do you have something more powerful?
**Turn 3 - Bot:** Yes! **Samsung Galaxy S24** for 2999 PLN
[OFFER:PHONE:**Samsung Galaxy S24**]
**Turn 4 - User:** The S24 sounds much better. That's quite expensive though. What plans do you have?
**Turn 5 - Bot:** We have **Plan M** for 75 PLN and **Plan L** for 95 PLN
[OFFER:BUNDLE:**Samsung Galaxy S24**#**Plan M**]
[OFFER:BUNDLE:**Samsung Galaxy S24**#**Plan L**]
**Turn 6 - User:** Okay, I'll take the S24 with Plan M
[ACCEPT_OFFER:BUNDLE:**Samsung Galaxy S24**#**Plan M**]
**Note:**
- Turn 2: "That's interesting" = NO TAG (just interest)
- Turn 4: "Sounds much better" + continues asking questions = NO TAG (interest, not commitment)
- Turn 6: "I'll take it" = ACCEPT (clear commitment)
[FINAL_CART:**Samsung Galaxy S24**#**Plan M**]
---
## Key Reminders
1. **Follow the decision tree step-by-step** for each utterance
2. **Copy product names exactly** as they appear (with `**bold**`)
3. **All phones and bundles are OFFERs** - regardless of price or features
4. **Only accessories and services are CROSS_SELLs**
5. **OFFERs are always bot-initiated** (no initiation tag needed)
6. **CROSS_SELLs require initiation type** (USER_INITIATED or BOT_INITIATED)
7. **Look back for initiation** - user might have mentioned product category several turns earlier;
   check entire conversation history
8. **Once initiated, always that type** - product category initiation type persists throughout
   conversation
9. **Match initiation types** - if original cross-sell was `:BOT_INITIATED`, acceptance/rejection/
   cancellation must also be `:BOT_INITIATED`
10. **Only tag explicit acceptances** - "I'll take it", not just "I like it"
11. **Interest != Acceptance** - "sounds good/better/interesting" without commitment = NO TAG
12. **Still negotiating = Not accepted** - if user continues asking questions after expressing
    interest, don't tag acceptance yet
13. **Only tag explicit rejections** - not when user simply chooses one alternative
14. **Multiple rejections** - tag each rejection separately, even with general statements
15. **Bundles are atomic** - rejecting part of a bundle = rejecting the whole bundle
16. **Alternative options (OR)** - create separate tags for each
17. **Bundled together (AND)** - use one tag with `#` separators
18. **Multiple tags per utterance** are allowed - tag all relevant actions
19. **Cancellations** - tag whenever user explicitly cancels or switches products
20. **Multiple products** - don't assume mutual exclusivity; user can buy multiple items
21. **Final cart summary** - after all utterances, add `[FINAL_CART:...]` listing all accepted
    products, or `[FINAL_CART:EMPTY]` if nothing purchased
---
**Now annotate the dialogue following this decision tree.**

## The dialogue for evaluation:
{conv_history}
```

B List of metrics

Feature	Definition	Category
# Sales Tags	Count of all sales-specific tags in the dialogue	Basic
# Phone-Only Offers	Count of phone-only offers made by the bot (P)	Basic (Offer)
# Bundle Offers	Count of bundle (phone + plan) offers made by the bot (B)	Basic (Offer)
Total # Offers	$O = P + B$	Basic (Offer)
# Bot-Initiated Cross-Sells	Count of cross-sells proactively suggested by the bot (CS_{bot})	Basic (Cross-sell)
# User-Initiated Cross-Sells	Count of cross-sells in response to user requests (CS_{usr})	Basic (Cross-sell)
Total # Cross-Sells	$CS = CS_{\text{bot}} + CS_{\text{usr}}$	Basic (Cross-sell)
# Phone Offer Acceptances	Count of user acceptances of phone-only offers (A_P)	Basic (Acceptance)
# Bundle Offer Acceptances	Count of user acceptances of bundle offers (A_B)	Basic (Acceptance)
Total # Offer Acceptances	$A_O = A_P + A_B$	Basic (Acceptance)
# Bot-Initiated Cross-Sell Acceptances	Count of acceptances of bot-initiated cross-sells ($A_{CS_{\text{bot}}}$)	Basic (Acceptance)
# User-Initiated Cross-Sell Acceptances	Count of acceptances of user-initiated cross-sells ($A_{CS_{\text{usr}}}$)	Basic (Acceptance)
Total # Cross-Sell Acceptances	$A_{CS} = A_{CS_{\text{bot}}} + A_{CS_{\text{usr}}}$	Basic (Acceptance)
Total # Acceptances	$A = A_O + A_{CS}$	Basic (Acceptance)
# Phone Offer Rejections	Count of user rejections of phone-only offers (R_P)	Basic (Rejection)
# Bundle Offer Rejections	Count of user rejections of bundle offers (R_B)	Basic (Rejection)
Total # Offer Rejections	$R_O = R_P + R_B$	Basic (Rejection)
# Bot-Initiated Cross-Sell Rejections	Count of rejections of bot-initiated cross-sells ($R_{CS_{\text{bot}}}$)	Basic (Rejection)
# User-Initiated Cross-Sell Rejections	Count of rejections of user-initiated cross-sells ($R_{CS_{\text{usr}}}$)	Basic (Rejection)
Total # Rejections	$R = R_O + R_{CS_{\text{bot}}} + R_{CS_{\text{usr}}}$	Basic (Rejection)
# Phone Offer Acceptance Cancellations	Count of canceled phone-only offer acceptances (C_P)	Basic (Cancellation)
# Bundle Offer Acceptance Cancellations	Count of canceled bundle offer acceptances (C_B)	Basic (Cancellation)
Total # Offer Acceptance Cancellations	$C_O = C_P + C_B$	Basic (Cancellation)
# Bot-Initiated Cross-Sell Acceptance Cancellations	Count of canceled bot-initiated cross-sell acceptances ($C_{CS_{\text{bot}}}$)	Basic (Cancellation)
# User-Initiated Cross-Sell Acceptance Cancellations	Count of canceled user-initiated cross-sell acceptances ($C_{CS_{\text{usr}}}$)	Basic (Cancellation)
Total # Cancellations	$C = C_O + C_{CS_{\text{bot}}} + C_{CS_{\text{usr}}}$	Basic (Cancellation)
Overall Conversion Rate	% of all offers and cross-sells that were accepted ($\frac{A}{O+CS}$)	Conversion
Overall Offer Conversion Rate	% of all offers that were accepted ($\frac{A_O}{O}$)	Conversion (Offer)
Phone Offer Conversion Rate	% of phone-only offers that were accepted ($\frac{A_P}{P}$)	Conversion (Offer)
Bundle Offer Conversion Rate	% of bundle offers that were accepted ($\frac{A_B}{B}$)	Conversion (Offer)
Overall Cross-Sell Conversion Rate	% of all cross-sells that were accepted ($\frac{A_{CS}}{CS}$)	Conversion (Cross-sell)
Bot-Initiated Cross-Sell Conversion Rate	% of bot-initiated cross-sells that were accepted ($\frac{A_{CS_{\text{bot}}}}{CS_{\text{bot}}}$)	Conversion (Cross-sell)

Table – continued from previous page

Feature	Definition	Category
User-Initiated Cross-Sell Conversion Rate	% of user-initiated cross-sells that were accepted ($\frac{A_{CS_{\text{usr}}}}{CS_{\text{usr}}}$)	Conversion (Cross-sell)
Overall Offer Rejection Rate	% of all offers that were explicitly rejected ($\frac{R_O}{O}$)	Conversion (Rejection)
Phone Offer Rejection Rate	% of phone-only offers that were rejected ($\frac{R_P}{P}$)	Conversion (Rejection)
Bundle Offer Rejection Rate	% of bundle offers that were rejected ($\frac{R_B}{B}$)	Conversion (Rejection)
Overall Cross-Sell Rejection Rate	% of all cross-sells that were rejected ($\frac{R_{CS_{\text{bot}}} + R_{CS_{\text{usr}}}}{CS}$)	Conversion (Rejection)
Bot-Initiated Cross-Sell Rejection Rate	% of bot-initiated cross-sells that were rejected ($\frac{R_{CS_{\text{bot}}}}{CS_{\text{bot}}}$)	Conversion (Rejection)
User-Initiated Cross-Sell Rejection Rate	% of user-initiated cross-sells that were rejected ($\frac{R_{CS_{\text{usr}}}}{CS_{\text{usr}}}$)	Conversion (Rejection)
Offer Cancellation Rate	% of accepted offers that were later canceled ($\frac{C_O}{A_O}$)	Conversion (Cancellation)
Cross-Sell Cancellation Rate	% of accepted cross-sells that were later canceled ($\frac{C_{CS_{\text{bot}}} + C_{CS_{\text{usr}}}}{A_{CS}}$)	Conversion (Cancellation)
Overall Cancellation Rate	% of all acceptances that were later canceled ($\frac{C}{A}$)	Conversion (Cancellation)
Net Offer Conversion Rate	Offer conversion rate after accounting for cancellations ($\frac{A_O - C_O}{O}$)	Conversion (Net)
Net Cross-Sell Conversion Rate	Cross-sell conversion rate after accounting for cancellations ($\frac{A_{CS} - C_{CS}}{CS}$)	Conversion (Net)
Net Overall Conversion Rate	Overall conversion rate after accounting for cancellations ($\frac{(A_O - C_O) + (A_{CS} - C_{CS})}{O + CS}$)	Conversion (Net)
# Turns to First Offer	t(1st offer), where $t(\cdot)$ = turn index of an event	Timing (Time to first action)
# Turns to First Cross-Sell	t(1st CS)	Timing (Time to first action)
# Turns to First Acceptance	t(1st acceptance)	Timing (Time to first action)
# Turns to First Rejection	t(1st rejection)	Timing (Time to first action)
# Turns from First Offer to First Acceptance	t(1st acceptance after 1st offer) $-$ t(1st offer)	Timing (Offer-to-Acceptance)
Avg. Turns Between Offers and Their Acceptances	$\frac{\sum_i \left[t(\text{acceptance}_i) - t(\text{most recent offer}_i)\right]}{\text{offer acceptances preceded by an offer}}$	Timing (Offer-to-Acceptance)
# Turns from First Acceptance to First Cross-Sell	t(1st CS after 1st acceptance) $-$ t(1st acceptance)	Timing (Cross-Sell)
Avg. Turns from Acceptance to Cross-Sell	$\frac{\sum_i \left[t(\text{next CS}_i) - t(\text{acceptance}_i)\right]}{\text{acceptances followed by a CS}}$	Timing (Cross-Sell)
Avg. Turns Between Cross-Sells and Their Acceptances	$\frac{\sum_i \left[t(\text{CS acceptance}_i) - t(\text{most recent preceding CS}_i)\right]}{\text{CS acceptances preceded by a CS}}$	Timing (Cross-Sell)
# Turns After Last Acceptance	$T - t$(last acceptance) $- 1$, where T = total number of dialogue turns	Timing (Post-Action)

Table – continued from previous page

Feature	Definition	Category
# Turns After Last Cancellation	$T - t(\text{last cancellation}) - 1$	Timing (Post-Action)
Avg. Offer Spacing	$\frac{\sum_i \left[t(\text{offer}_{i+1}) - t(\text{offer}_i)\right]}{\text{consecutive offer pairs}}$	Timing (Spacing)
Avg. Cross-Sell Spacing	$\frac{\sum_i \left[t(\text{CS}_{i+1}) - t(\text{CS}_i)\right]}{\text{consecutive CS pairs}}$	Timing (Spacing)
Avg. Recovery Turns After Rejection	$\frac{\sum_i \left[t(\text{next offer}_i) - t(\text{rejection}_i)\right]}{\text{rejections followed by an offer}}$	Timing (Recovery)
Avg. Recovery Turns After Cancellation	$\frac{\sum_i \left[t(\text{next offer}_i) - t(\text{cancellation}_i)\right]}{\text{cancellations followed by an offer}}$	Timing (Recovery)
Offer-to-Cross-Sell Ratio	Ratio of direct offers to cross-sell attempts ($\frac{O}{CS}$)	Strategy (Product Mix)
Phone-to-Bundle Ratio	Ratio of phone-only offers to bundle offers ($\frac{P}{B}$)	Strategy (Product Mix)
Bot-to-User Initiated Cross-Sell Ratio	Ratio of bot-initiated to user-initiated cross-sells ($\frac{CS_{\text{bot}}}{CS_{\text{usr}}}$)	Strategy (Product Mix)
Bot Initiative Rate	% of cross-sells that are bot-initiated ($\frac{CS_{\text{bot}}}{CS}$)	Strategy (Initiative)
User Initiative Rate	% of cross-sells that are user-initiated ($\frac{CS_{\text{usr}}}{CS}$)	Strategy (Initiative)
Bot Initiative Success Rate	Conversion rate of bot-initiated cross-sells ($\frac{A_{CS_{\text{bot}}}}{CS_{\text{bot}}}$)	Strategy (Initiative)
User Initiative Success Rate	Conversion rate of user-initiated cross-sells ($\frac{A_{CS_{\text{usr}}}}{CS_{\text{usr}}}$)	Strategy (Initiative)
Post-Acceptance Cross-Sell Rate	Rate of cross-sells following acceptances ($\frac{CS}{A_O}$)	Strategy (Post-Action)
Post-Rejection Offer Rate	$\frac{\text{rejections followed by} \geq 1 \text{ new offer}}{R}$	Strategy (Post-Action)
Post-Cancellation Offer Rate	$\frac{\text{cancellations followed by} \geq 1 \text{ new offer}}{C}$	Strategy (Post-Action)
Rejection Recovery Success Rate	$\frac{\text{rejections eventually followed by an acceptance}}{R}$	Strategy (Recovery)
Cancellation Recovery Success Rate	$\frac{\text{cancellations eventually followed by an acceptance}}{C}$	Strategy (Recovery)
Avg. Offers Per Rejection	$\frac{\text{total offers made after all rejections}}{R}$	Strategy (Recovery)
Avg. Offers Per Cancellation	$\frac{\text{total offers made after all cancellations}}{C}$	Strategy (Recovery)
Persistence Score	Measure of sales persistence after rejection ($\frac{O-1}{R_O}$)	Strategy (Persistence)
Adaptation Score	$\frac{\text{rejections or cancellations followed by a new offer}}{R+C}$	Strategy (Persistence)
Total Negotiation Rounds	Total count of offer–response cycles (accepted or rejected)	Flow (Negotiation)
Rejection Negotiations	Count of offer–rejection cycles	Flow (Negotiation)
Acceptance Negotiations	Count of offer–acceptance cycles	Flow (Negotiation)
Negotiation Success Rate	$\frac{\text{acceptance negotiations}}{\text{total negotiations}}$	Flow (Negotiation)
Avg. Negotiation Length	Average turns between an offer and its response	Flow (Negotiation)
Avg. Successful Negotiation Length	Average turns between an offer and its acceptance	Flow (Negotiation)
Avg. Unsuccessful Negotiation Length	Average turns between an offer and its rejection	Flow (Negotiation)

Table – continued from previous page

Feature	Definition	Category
# Modification Rounds	Count of acceptance–cancellation–new acceptance cycles	Flow (Negotiation)
Sales Tags Density	$\frac{\text{Sales Tags}}{T}$	Flow (Density)
Offer Tags Density	Proportion of turns containing offers $(\frac{O}{T})$	Flow (Density)
Cross-Sell Tags Density	Proportion of turns containing cross-sells $(\frac{CS}{T})$	Flow (Density)
Conversion Efficiency Rate	Acceptances per dialogue turn $(\frac{A}{T})$	Flow (Efficiency)
Net Conversion Efficiency Rate	Net acceptances after cancellations per dialogue turn $(\frac{A-C}{T})$	Flow (Efficiency)
Cross-Sell Efficiency Rate	Cross-sell acceptances per total cross-sells $(\frac{A_{CS}}{CS})$	Flow (Efficiency)
Sales Cycle Length	t(last acceptance) $- t$(1st offer)	Flow (Efficiency)
First Offer to First Acceptance	t(1st acceptance after 1st offer) $- t$(1st offer)	Flow (Efficiency)
Offers to Acceptance Ratio	Average # of offers needed to achieve each acceptance $(\frac{O}{A})$	Flow (Efficiency)
# Accepted Products	Total count of accepted offers and cross-sells $(A_O + A_{CS})$	Flow (Product)
# Net Accepted Products	Total count after accounting for cancellations $(A - C)$	Flow (Product)
Customer Decisiveness Score	Ratio of acceptances to total interactions $(\frac{A}{A+R+C})$	Flow (Customer Journey)
Customer Engagement Score	Ratio of user-initiated cross-sells to total cross-sells $(\frac{CS_{\text{usr}}}{CS})$	Flow (Customer Journey)
Purchase Path Complexity Score	# of cancellations plus rejections relative to acceptances $(\frac{C+R}{A})$	Flow (Customer Journey)
# Final Cart Items	Count of items in the final cart (from `FINAL_CART` tag) (F)	Cart-Specific
Has Final Purchase	Binary indicator: $\mathbf{1}[F > 0]$	Cart-Specific
Cart Completion Rate	% of accepted items that made it to final cart $(\frac{F}{A})$	Cart-Specific
Cart Abandonment Rate	% of accepted items NOT in final cart $(\frac{A-F}{A})$	Cart-Specific
Has Bundle in Cart	Binary indicator: $\mathbf{1}$[at least one bundle product in final cart]	Cart-Specific