Sequential Search Transformer: A Deep Structural Econometric Model Online Appendix

Yicheng Song
Carlson School of Management, University of Minnesota, Minneapolis, MN 55455, ycsong@umn.edu

Tianshu Sun
Cheung Kong Graduate School of Business, Beijing, China 100006, tianshusun@ckgsb.edu.cn

Appendix A: Notation Table

Table 1 Table of Notations

	Notation	Format	Description
Data	i	Scalar	Consumer index
	h	Scalar	Session index
	j	Scalar	Product index
	t	Scalar	Search index in the focal session
	$j^{'}$	Scalar	Product index
	H_{ih}	Matrix	Historical Interaction for consumer i before h-th session
	M_{ih}	Matrix	Memory Representation of H_{ih}
	x_j	Vector	Exposed feature of product j on product list page
	π_j	Scalar	Price of product j
	$\widetilde{x}_j$	Vector	Features of product j only show on product page
	S_{it}	Set	Searched items from consumer i before the t-th search
	$\bar{S}_{it}$	Set	Unsearched items from consumer i before the t-th search
Sequential Search	$\bar{p}_i$	Vector	Intrinsic Preference of consumer i
	$p^{'}_{ih}$	Vector	Preference shock for consumer i in the h-th session
	p_{ih}	Vector	Preference for consumer i in the h-th session
	α_i	Scalar	Price sensitivity for consumer i
	γ_i	Scalar	Searching cost sensitivity of consumer i
	c_i	Scalar	Searching cost of consumer i
	u_{ij}	Scalar	Expected utility of consumer i to item j
	v_{ij}	Scalar	Known utility of consumer i to item j before search
	ϵ_{ij}	Scalar	Random error in u_{ij} that follows standard Normal distribution
	$u^{'}_{ij}$	Scalar	Post-search utility of consumer i to item j
	$\epsilon^{'}_{ij}$	Scalar	Resolve the utility uncertainty ϵ_{ij} after the search product
	u^*_{it}	Scalar	Best post-search utility for consumer i before the t-th search
	r_{ij}	Scalar	Reservation utility for consumer i to search item j
Transformer	$Encoder()$	Function	Encoder of the Transformer model
	$Decoder()$	Function	Decoder of the Transformer model
	q	Vector	Query vector
	K_{ih}	Matrix	Key Matrix for Historical Interaction H_{ih}
	v_{ih}	Matrix	Value vector for Historical Interaction H_{ih}
	σ^2	Scalar	Variance of the Normal prior distribution of $p^{'}_{ih}$
	$Softmax()$	Function	Softmax function
	$Linear()$	Function	A linear layer of neural network

Appendix B: Monte Carlo Study

We employ Monte Carlo simulations to demonstrate the feasibility of parameter identification. For each consumer i, the intrinsic preference $\bar{p}_i$, price sensitivity α_i, and the consumer-specific search cost parameter γ_i are drawn from a standard normal distribution. Using $(\bar{p}_i, \alpha_i, c_i)$, we simulate a consumer search session based on the standard sequential search; this session serves as the historical interaction H_{ih}. Building on $\bar{p}_i$ and H_{ih}, we update the preference p_{ih} and follow the mechanism in Section 3 to simulate the interactions in the incoming session. In this session, exploration set $\bar{S}_{i0}$ is sampled from a product warehouse. Each product $j \in \bar{S}_{i0}$ is characterized by two types of features: x_j, the exposed product features visible on the product list, and $\widetilde{x}_j$, the product features available only on the product page. The simulation process is detailed in Algorithm 1. The dataset consists of 300 sessions from 10 consumers, involving 5000 different products.

Algorithm 1 Generate a Simulated Search Session for Monte Carlo Simulation

Input: Unsearched Product set $\bar{S}_{i0}$, Intrinsic preference $\bar{p}_i$, searching cost $c_i = exp(\gamma_i)$, and Historical interaction H_{ig}
Initialization: Initialization: $t = 0$, Searched Product set $S_{it} = 0$ (outside option)
Initialization: Highest utility: $u^*_{it} = 0$
Initialization: Compute reservation utility r_{ij} for each product $j \in \bar{S}_i$ based on `RU_Net`
Compute $p^{'}_{ih}$ by feeding H_{ih} into an RNN with randomly initialized parameters
Update p_{ih} based on Equation 1
while $\max_{j \in \bar{S}_{it}} r_{ij} \geq u^*_{it}$ **do**
    Select the product j to search based on Equation 10
    Compute $e^{'}_{ij} = Linear(\widetilde{x}_j)$
    Update $u^{'}_{ij}$ based on Equation 15
    Update $t = t + 1$
    Update $S_{it} = S_{i(t-1)} \cup \{j\}$, $\bar{S}_{it} = \bar{S}_{i(t-1)} \setminus \{j\}$
    Update u^*_{it} based on Equation 7
end while
Purchase product $j \in S_{it}$ base on Equation 13

We train the SST model on the simulated data to estimate the parameters. The proposed SST model includes four sets of parameters: 1. Consumer intrinsic preference $(\bar{p}_i)$ and price sensitivity α_i, 2. Preference shock $(p^{'}_{ih})$, 3. Consumer cost coefficient (γ_i), and 4. Utility uncertainty evaluation $(\epsilon^{'}_{ij})$. Since each consumer has a distinct set of parameters, listing the ground truth and corresponding estimated parameters in a table is infeasible due to the large number of parameters. Instead, we compare the differences between the ground truth and estimated parameters using Mean Absolute Error (MAE) and Mean Absolute Percentage Error

Table 2 Parameter Recovery Performance on the Simulation Data

	MAE	MAPE
Intrinsic preference ($\bar{p}_i, \alpha_i$)	0.0064	0.0305
Preference shock ($p^{'}_{ih}$)	0.0040	0.0351
Cost sensitivity (γ_i)	0.0031	0.0172
Utility uncertainty ($\epsilon^{'}_{ij}$)	0.0048	0.0946

(MAPE) [1] to evaluate the performance of parameter recovery. As shown in Table 2, our model achieves excel parameter recovery performance, with MAE and MAPE values being extremely small. This demonstrates the model's capability in recovering and identifying ground-truth parameter sets. The code for generating the simulation data and implementing the model estimation procedure is publicly available. [2]

The simulation results demonstrate that the proposed SST model can effectively disentangle and recover the parameters for intrinsic preference, preference shock, search cost sensitivity, and utility uncertainties. This indicates that the model is well-suited for application in empirical contexts to estimate these structural parameters within the sequential searching framework. We also acknowledge mismatches between the simulation procedure and the modeling assumptions. Specifically, the simulation does not enforce that preference shocks or realized uncertainty originate from normal distributions.

[1] To avoid distortions in percentage calculations due to extremely small denominators, we evaluate MAPE only when the ground-truth value (denominator) is greater than 0.01.

[2] `https://github.com/DeepCPT/SST_Model`

Appendix C: Product Features on Product List and Product Page

Table 3 Exposed Features on Product List

Cate	Exposed Features	Cate	Exposed Features	Cate	Exposed Features
Computes	Screen Size	Cell Phone	Model Family	Refrigerator	Refrigerator Style
	Screen Resolution		Built-in Storage		Capacity
	CPU		Locked or Unlocked		# of Doors
	Ram		Screen Type		Ice Maker
	GPU		Camera		Water Filter
	Hard Disk Capacity		Color		LED Screen
	Color	TV	Screen Size		Color
Tablets	Screen Size		Resolution	Microwave	Microwave Style
	Chip		Display Type		Capacity
	Display Surface		Smart Capable		Sensor Cooking
	Storage Capacity		Direct/Edge Lit		Control Type
	Pen	Dryer	Capacity		Color
	Keyboard		Steam Function	Projector	Resolution
	Color		Electric/Gas		Brightness
Printer	Wireless or Cable		Moisture Sensor		Light Source
	Printer Type		Stackable		Portability
	Laser Print or not		Color		Wireless
	Display Screen	Washer	Capacity	Common Features	Brand
	Print Speed		High-Efficiency		Price
	Ink Duration		Stackable		# of Review
	Scanner Resolution		Steam Function		Avg Rating
			Color		Pick Up Ready

Table 4 Sample of Product Features only shown on Product Page (PG)

Cate	Features on PG	Cate	Features on PG	Cate	Features on PG
Computer	Operating System	Cell Phone	Display Type	Washer	Load Type
	Wireless Connection		Refresh Rate		App Compatible
	Type of Memory		Brightness		Antimicrobial
	Storage Type		Front-Facing Camera		Vibration Reduction
	Display Connector		Rear-Facing Camera		Spin Speed
	Touch Screen		Optical Zoom		......
	Refresh Rate		Digital Zoom	Refrigerator	Depth
	......		......		Lighting Type
Tablet	Screen Resolution	TV	HDR		Refrigerator Capacity
	Wireless Connection		Refresh Rate		Freezer Capacity
	Expandable Memory		Streaming Services		Fingerprint Resistant
	Display Connector		TV Tuner Type		......
	# of port		Steam Function	Microwave	Wireless
	Battery Capacity		# of HDMI Inputs		Fingerprint Resistant
	Security Features		Network Connector		Turntable
	......		......		Cooktop Lighting
Printer	Output Capacity	Dryer	Time-Remaining Display		Display Type
	Auto Doc Feeder		Reversible Door Hinge		......
	Scanner Type		Exhaust Vent Location	Projector	Product Weight
	Integrated Fax		Compact Design		Color Brightness
	Cartridges		Drum Material		White Brightness
	Memory Card		Drying Rack		Contrast Ratio
	Auto Two-Sided		Drying Cycles		Connector
	......		......		......

Appendix D: Additional Data Summary

Figure 1 Products Distribution Across Different Categories

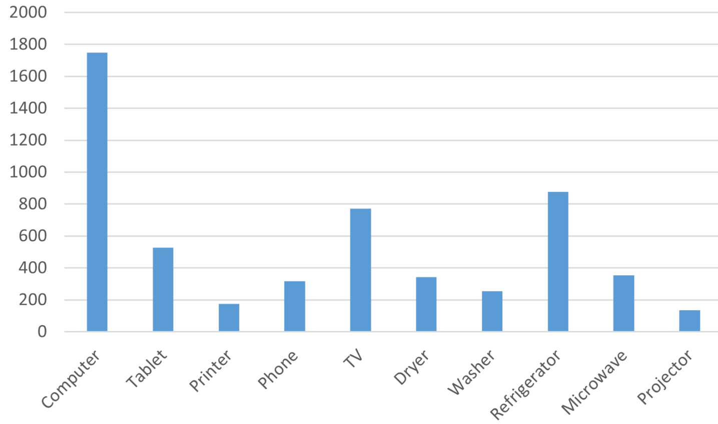


Figure 2 Search Distribution Across Different Categories

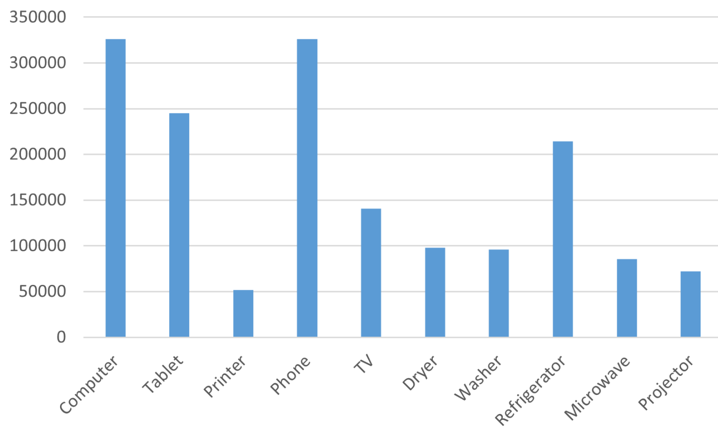
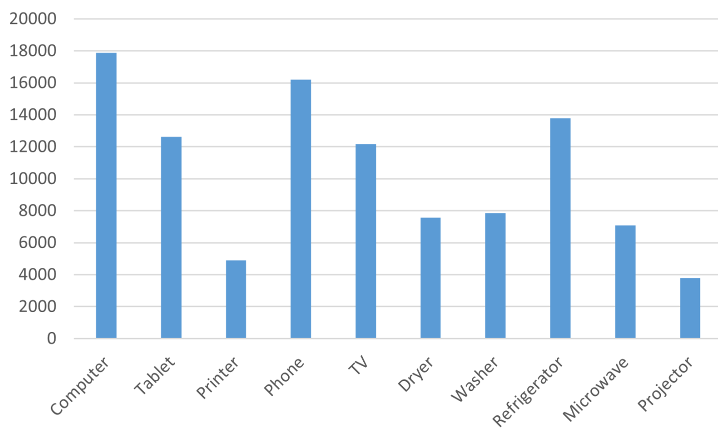


Figure 3 Purchase Distribution Across Different Categories



Appendix E: Sequential Search or Simultaneous Search

Consumer search models generally classify search behavior into two primary schemes: sequential search and simultaneous search. In this study, we adopt a sequential search framework, where consumers evaluate product prices and other features one at a time and decide at each step whether to continue searching or to stop. This process is modeled as an optimal stopping problem. In contrast, simultaneous search assumes that consumers commit to a predefined search set, retrieve all relevant product information at once, and then select the alternative that best matches their preferences or offers the lowest price. Honka and Chintagunta (2017) study auto insurance searches and propose a method to distinguish between these two search strategies. They argue that if consumers engage in simultaneous search, the mean price of products in their consideration set should be equal to the mean price across the entire population. However, under sequential search, the mean price of searched products may deviate from the population mean, as consumers adaptively decide whether to continue or stop searching based on observed attributes. Applying this framework to our study, we analyze three types of product features that consumers might use in their search process: 1. Price (π_j), 2. Exposed features on the product list page (x_j), which are visible before a consumer clicks on a product and 3. Features available only on the product page ($\widetilde{x}_j$). For each feature type, we compute the mean value across all products in the focal category (representing the population mean) and compare it to the mean value of the searched product set. [3] To assess whether consumers engage in sequential search, we perform statistical tests to determine whether the mean of the searched product set differs significantly from the population mean. The resulting average p-values for the three feature types are: (π_j): $p = 0.0263$, (x_j): $p = 0.0152$, ($\widetilde{x}_j$): $p = 0.0117$. All p-values indicate significant differences between the means of the searched product set and the overall population, suggesting that consumers do not predefine a fixed search set but rather adaptively update their decisions based on observed product features. This evidence supports the notion that consumers in our study engage in sequential search, reinforcing the suitability of the empirical context for applying the sequential search model.

Appendix F: Model Performance in the Thanksgiving Week

Our dataset contains detailed clickstream records between June 2018 and May 2019, together with the recommendation lists displayed at each interaction, we are able to observe price changes for recommended products over time. The largest price promotions occur during the Thanksgiving week, and the magnitude of these discounts varies across products. Additionally, 6.2% of the new products were introduced in that week, making the market environment particularly dynamic and uncertain. Thus, we therefore use Thanksgiving week (November 19–25, 2018) as a natural testing period. We train all models on pre–Thanksgiving data and evaluate their performance during Thanksgiving week following the same procedure described in Section 5.2. The comparative results, presented in Table 5, show that SST remains the most robust predictor even in this highly dynamic context.

[3] Categorical features are excluded from this statistical test because the mean representation of one-hot encoding is not meaningful for statistical comparisons.

Table 5 Prediction Performance Comparison during the Thanksgiving Week

	Item Click		Item Purchase		Stop Search
	Recall@10	MRR@10	Recall@10	MRR@10	RMSE
GRU4Rec	0.4928 ***	0.1883 ***	0.5628 ***	0.2326 ***	Not Applicable
DMT	0.5156 ***	0.2002 ***	0.6172 ***	0.2603 ***	Not Applicable
SSM_{V1}	0.3812 ***	0.1366 ***	0.6937 ***	0.3058 ***	3.3815 ***
SSM_{V2}	0.4034 ***	0.1487 ***	0.7233 ***	0.3205 ***	2.6149 ***
SST_{V1}	0.4509 ***	0.1605 ***	0.7768 ***	0.3451 ***	2.8216 ***
SST_{V2}	0.5838 ***	0.2271 **	0.8157 ***	0.3481 **	1.8857 ***
SST	0.5972	0.2382	0.8516	0.3607	1.7031

Note: ** $p < .05$, *** $p < .01$ represents the P-value of whether the performance is significantly different with SST.

Table 6 Prediction Performance Comparison

	Item Click		Item Purchase		Stop Search
	Recall@10	MRR@10	Recall@10	MRR@10	RMSE
SST_{V3}	0.6617 ***	0.2567 **	0.9154 **	0.3882 *	1.6921 *
SST	0.6875	0.2664	0.9283	0.3935	1.6835

Note: ** $p < .05$, *** $p < .01$ represents the P-value of whether the performance is significantly different with SST.

Appendix G: Independent Assumption of ϵ_{ij} and Dependencies Among $\epsilon_{ij}^{'}$

To retain the Weitzman-style sequential search logic, we model $\epsilon_{ij} \sim \mathcal{N}(0,1)$ and assume that these shocks are independent prior to search. This implies that before inspecting any product, consumers perceive utility uncertainty to be independent across items. After the search, however, the realized uncertainty ϵ'_{ij} can be modeled as a function of focal product features alone or as a function that also incorporates historical interactions. If ϵ'_{ij} were determined solely by focal product features, there would be no divergence between the i.i.d. assumption and the subsequent inference of realized utility shocks. We implement this specification as SST_{V3}, where ϵ'_{ij} is just a linear functional form based on the exposed product features $\widetilde{x}_j$, and compare its performance with the original version in which ϵ'_{ij} is estimated using both focal product information and past interactions through the attention mechanism. We evaluate both model setups in our empirical setting. As shown in Table 6, the SST structure consistently outperforms SST_{V3} across all evaluation metrics. These empirical results indicate that incorporating historical interactions enhances predictive accuracy.

The attention mechanism in the Transformer decoder dynamically learns ϵ'_{ij} based on both the focal product and the consumer's historical interactions, which leads to improved empirical prediction performance. For this reason, we adopt this flexible specification rather than SST_{V3} that learns ϵ'_{ij} based only on the exposed product features $\widetilde{x}_j$.

References

Honka E, Chintagunta P (2017) Simultaneous or sequential? search strategies in the us auto insurance industry. *Marketing Science* 36(1):21–42.