

Generative Brand Choice

Kevin Lee

University of Chicago, Booth School of Business

October 3, 2024

Fundamental Brand Issues



New brand



Brand extension



Brand crossover

- Positioning a new brand, brand extension, or brand crossover
- Pricing
- Predict who'll like a new brand

Modeling Demand

Choices are driven by latent preferences over product attributes and price.

[Lancaster (1966), McFadden (1974), BLP (Berry et al. (1995)), ...]



Calories, caffeine



Calories, sodium,
fiber



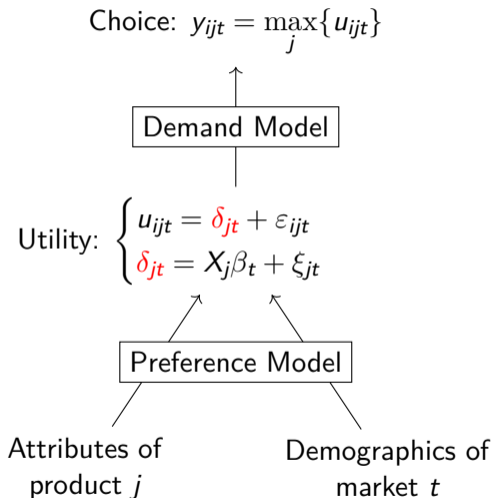
Size, HP, MPG

Often, the observed product attributes don't explain choices well.

For **existing** products, fixed effects can capture prefs over unobserved attributes.

What to do for products from **new** brands?

Standard model of demand (and its limitations)



Let δ_{jt} be systematic part of utility.

Problem: observed info is inadequate for predicting δ_{jt} for new brands

Solution: Collect more information!

Past work: incorporating additional data in demand estimation

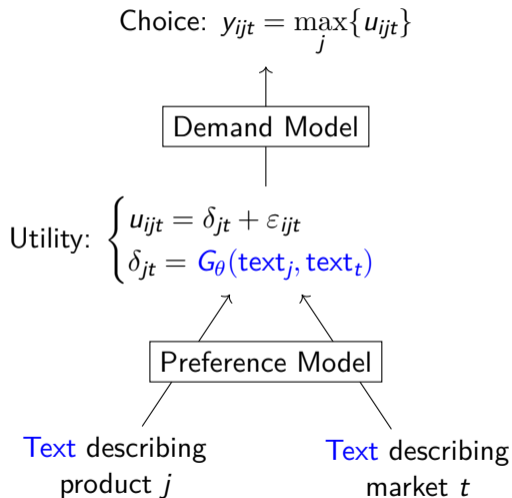
Measuring perceptual attributes, e.g. “mushyness” of cereal (Nevo (2001)), can help.

- Judgment call: requires analyst choose the right attributes to measure
- Cost: requires survey, which can be expensive
- Information loss: reducing a brand to attributes is lossy

Other approaches:

- Berry et al. (2004): judgment call with fixed effects of existing products
- Sisodia et al. (2024): automated attribute extraction from images, worth trying on text.

Proposal: Explain brand preferences with text data



Premise: text is informative

Suppose you have text that describes product j , market t .

I show how to incorporate raw text **directly** using an LLM.

- I propose a method that augments a structural demand model with
 - (i) Text data
 - (ii) A large language model (LLM)
 - (iii) A novel method for adapting the LLM to the structural model
- Importantly, (i) or (ii) on their own fail

My method: Augment structural model with fine-tuned LLM

1. Estimate δ_{jt} for existing brands and markets with standard methods.
2. Collect text descriptions of existing brands and markets.
3. **Fine-tune** LLM to predict the δ_{jt} 's from Step 1 given text from Step 2.
4. Use tuned LLM to predict out of sample on **new** brands or markets.

1. Estimate δ_{jt} with standard methods

Estimate δ_{jt} with standard methods (Berry (1994), BLP (1995), Nevo (2001)).

Berry (1994) logit: $\hat{\delta}_{jt} = \log(s_{jt}/s_{0t})$.

Arrange estimates into a matrix. Goal: extrapolate to **new** brands and markets.

	Brand 1	Brand 2	...	Brand J	New Brand
Market 1	10	12	...	9	?
Market 2	3	16	...	7	
⋮	⋮	⋮	⋮	⋮	
Market T	8	3	...	12	
New Market	?				

2. Collect text descriptions of brands and markets

ChatGPT query: In 80 words or less, describe the positioning of the {brand} energy drink brand.

Example response: Red Bull is positioned as a premium energy drink that enhances mental and physical performance. Targeted primarily at young adults, it's marketed as a vital tool for energetic lifestyles, enhancing endurance, concentration, reaction speed, and mood. Famous for its slogan "Red Bull gives you wings," the brand heavily invests in marketing through extreme sports, music festivals, and youth culture events, creating a strong, adventurous brand image that resonates globally with those seeking excitement and a performance edge.

2. Collect text descriptions of brands and markets

ChatGPT query: In 80 words or less, describe the zip code {zip_code}. Talk about the types of people that live there.

Example response: This suburban area is home to many families and professionals. Residents often enjoy spacious homes, good schools, and community amenities. The population includes a mix of young families, established professionals, and retirees, reflecting a range of cultural backgrounds. The area is known for its safety and family-friendly environment, with numerous parks and recreational options.

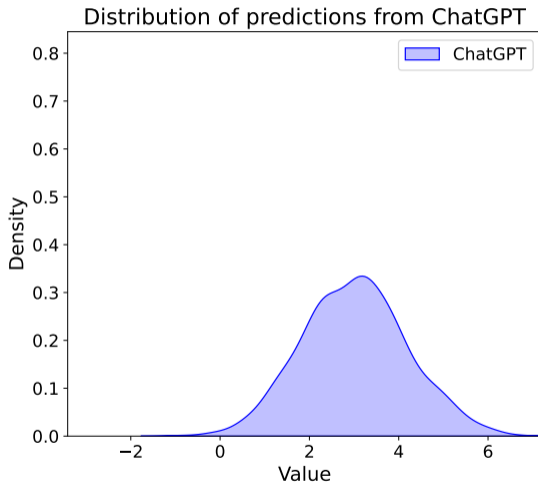
3. Fine-tune LLM with the estimated δ_{jt} 's

LLM-generated predictions

User: Predict the average utility for {brand_j} in {market_t}.
Give a numerical response only.

ChatGPT: 2.5

$\times 100$



3. Fine-tune LLM with the estimated δ_{jt} 's

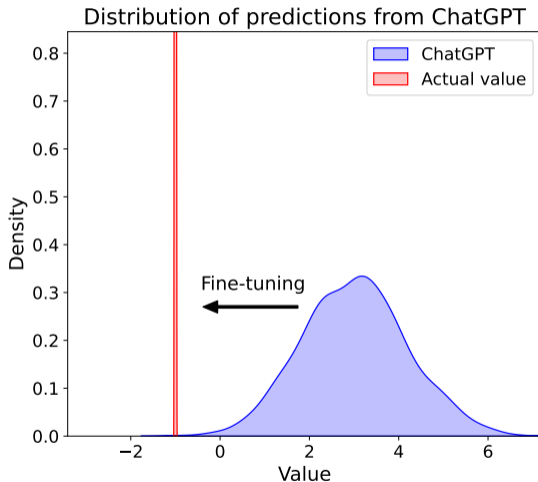
LLM-generated predictions

User: Predict the average utility for {brand_j} in {market_t}.
Give a numerical response only.

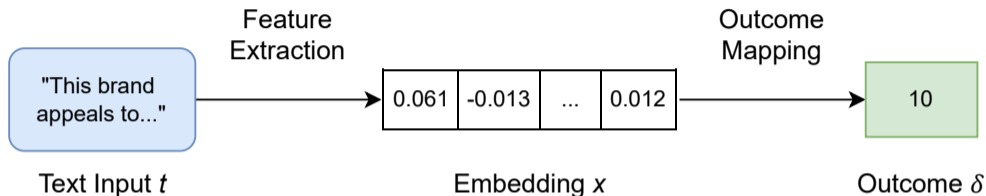
ChatGPT: 2.5

Increase likelihood of “correct” δ_{jt} 's:

$$\max_{\theta} E[\log G_{\theta}(\delta_{jt} | \text{text}_j, \text{text}_t)]$$



Background: two key steps in text regression

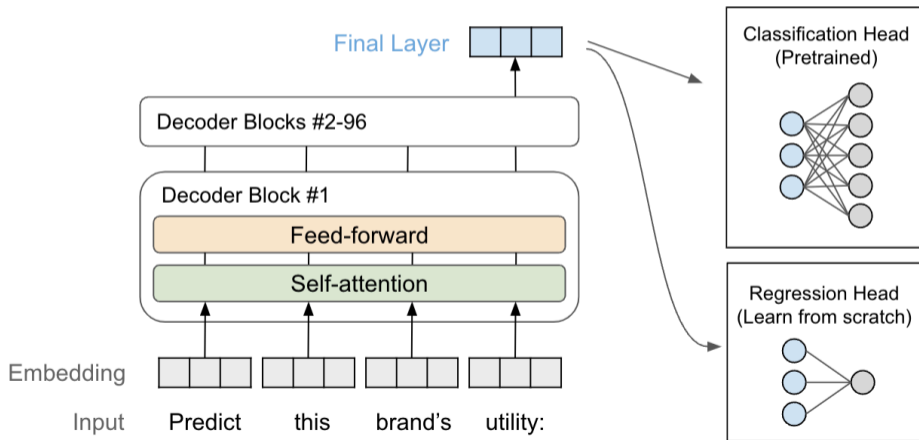


- Raw text is turned into numerical features then mapped to the outcome
- Feature extractor: take as given from pretrained model or adapt to data
- Outcome map: learn from scratch or adapt from pretrained model

Feature extraction and outcome mapping in an LLM

Feature Extraction

Outcome Mapping



Intuition: embeddings will struggle to extrapolate

- Learning the *mapping* from 3,000 dimension embedding to outcome will be hard
- Learning this from scratch requires an enormous amount of data – “knowledge” and “logic” emerge in 10B parameter LLMs trained on 150B words¹
- Thus leverage the abilities of a pretrained LLM and adapt it to my setting

¹Emergent Abilities of LLMs (Wei et al. (2022)),
Training Compute-optimal LLMs (Hoffmann et al. (2022))

Fine-tuning details

- LLM defines a conditional probability distribution $G_{\theta}(\cdot | \text{text}_j, \text{text}_t)$
- Fine-tuning changes θ to increase $G_{\theta}(\delta_{jt} | \text{text}_j, \text{text}_k)$
- To prevent overfitting, only allow parameters to change in a small neighborhood, which works well if the *initialization* of the parameters is good
- Notation: fine-tuning updates θ_0 to $\theta = \theta_0 + \Delta\theta$

Once fine-tuning is done, use LLM to predict δ_{jt} for new brands.

Application 1: Predicting Unseen Brands

- Q: Can market shares for new brands be predicted from brand positioning statements?
- Need ground truth to evaluate so hold out existing brands while taking care to avoid data leakage.

Evaluation strategy

I evaluate how well various methods might do on new data by holding out existing data.

Arrange δ_{jt} estimates from structural model into a matrix:

	Brand 1	Brand 2	...	Brand J-1	Brand J
Market 1	10	12	...	?	?
Market 2	3	?	...	?	?
⋮	⋮	⋮	⋮	⋮	⋮
Market T	?	3	...	?	?

- All data for two “new” brands (green data) are held out to form the **test** set
- Remaining data is randomly split 80%-20% into a **training** and **validation** set

Choice Data

Energy drink purchase data from Numerator: 465k individuals, 3.2m trips, 6.2m items.

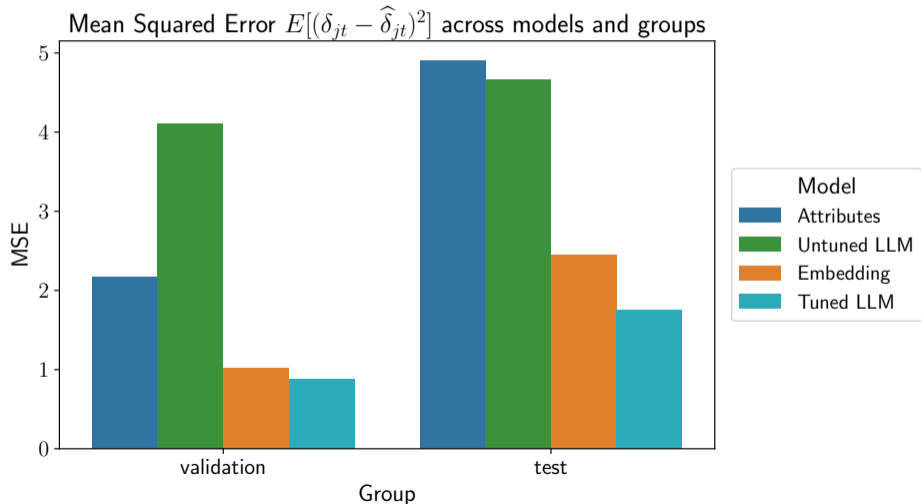


I compare the following methods on predicting held-out brands:

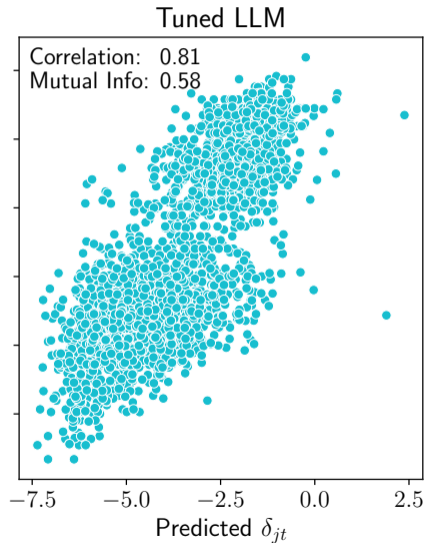
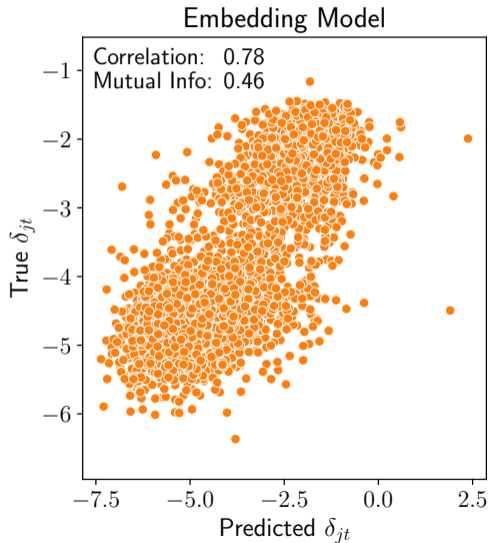
1. Logit model using measurable product attributes
2. Untuned LLM (asking ChatGPT to guess)
3. Machine learning methods based on text embeddings
4. LLM fine-tuned with my procedure

Results: text data improves predictions of δ_{jk}

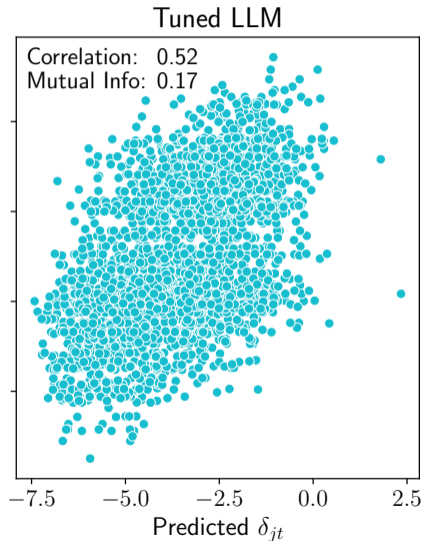
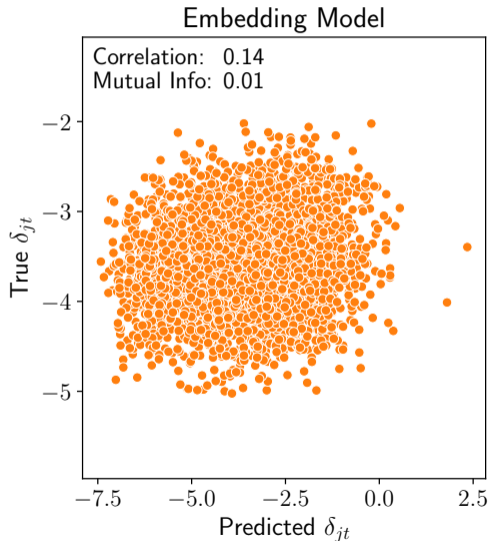
Embeddings are good in validation set, tuned LLM better in test set (“new” brands).



Both embeddings and tuned LLM predict well on validation set

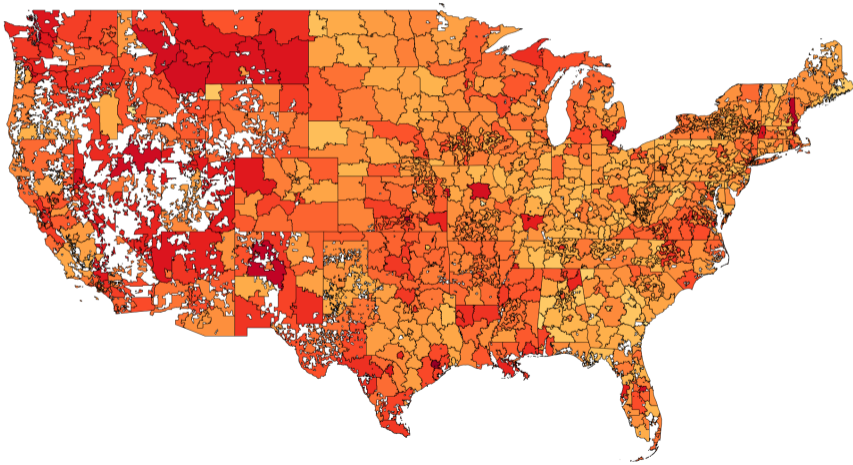


Tuned LLM much more *informative* on held out brands (test set)



Identifying target markets for Celsius

Can predict utilities even for areas where Numerator doesn't have coverage.



Solving for optimal prices

- Define the non-price part of utility:

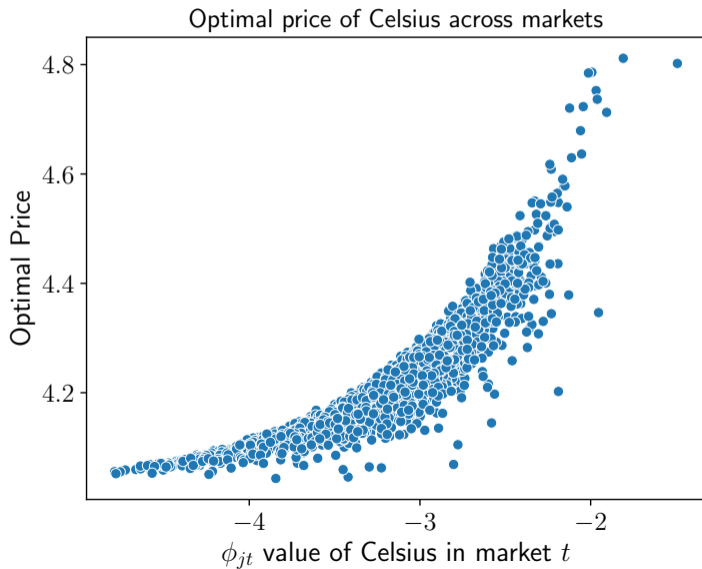
$$\phi_{jt} := \delta_{jt} + \alpha p_{jt}$$

δ_{jt} is estimated as before, α is causal IV estimate, p_{jt} is observed price

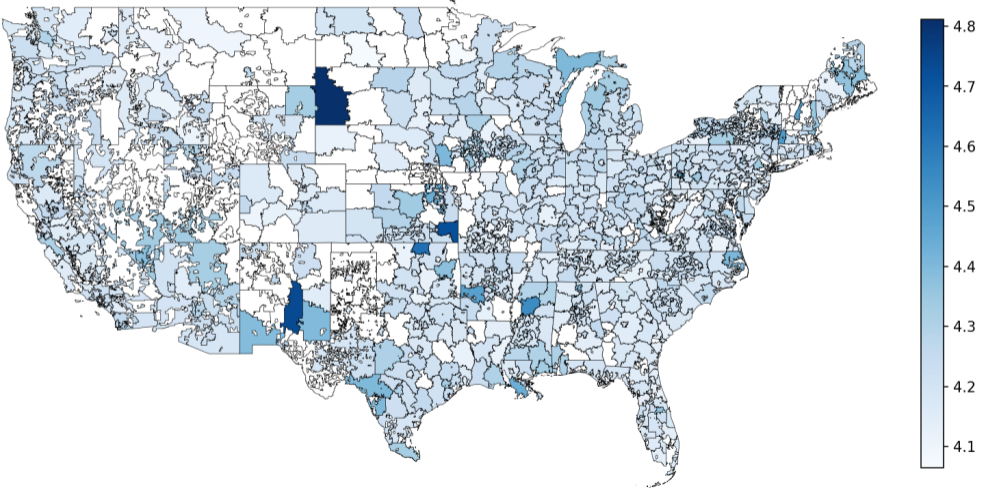
- Train a model to predict ϕ_{jt} from text
- Numerically solve first order condition for profit with respect to price:

$$\begin{aligned}\pi'(p) &= p - c - \frac{1}{\alpha(1 - s_{jt})} \\ &= p - c - \frac{1}{\alpha} \left(1 - \frac{\exp(\phi_{jt} - \alpha p)}{1 + \sum_{j'} \exp(\delta_{j't})} \right)^{-1}\end{aligned}$$

Optimal price varies due to preference heterogeneity and competition



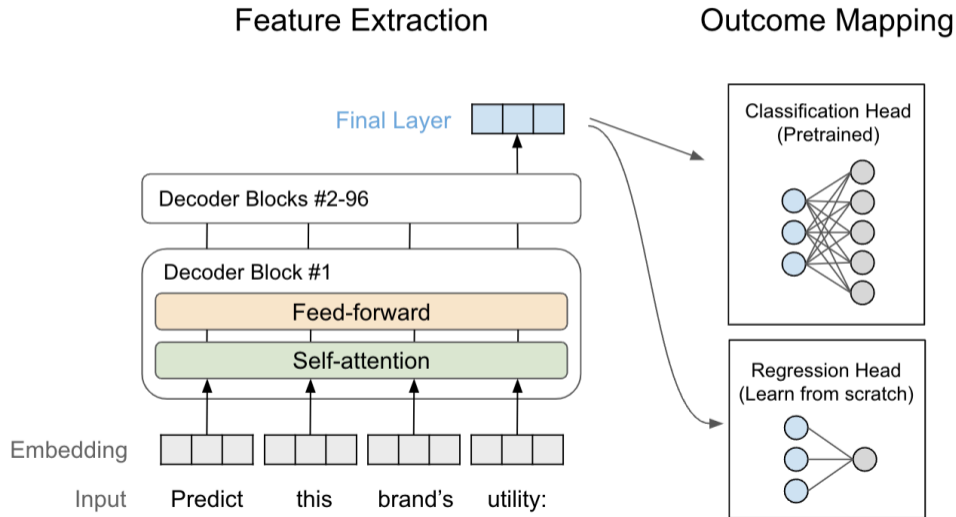
Geographical distribution of optimal prices for Celsius



Interlude: Factors affecting performance

- To get some sense of why the tuned LLM works, I evaluate how changes to specific components affect prediction accuracy.
- I evaluate performance for 4 cases
 - ▶ Outcome map is {learned from scratch, adapted from pretrained LLM}
 - ▶ Embeddings are {taken as given from pretrained LLM, adapted to the data}

Recall: Feature extraction and outcome mapping in an LLM



Empirical Results: MSE on held out brands using Mistral-7B

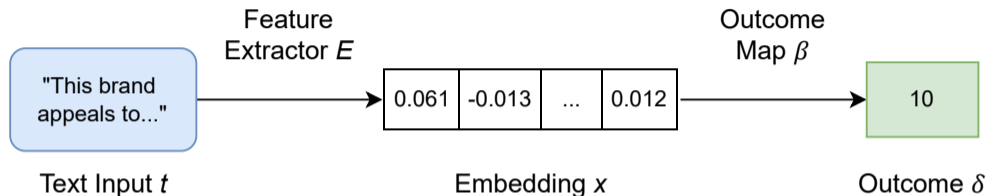
		Outcome Mapping	
		From Scratch	From LLM
Embedding	Frozen	3.136	2.718
	Adapted*	2.75	1.896

- Adapting the embedding and the LLM's prior on the outcome map are each helpful
- Combination is even better
- Detail: how you adapt the embeddings matters - normal gradient descent fails

Key detail: naive gradient descent doesn't work

		Outcome Mapping	
		From Scratch	From LLM
Embedding	Frozen	3.136	2.718
	Adapted	3.22	3.329
	Adapted*	2.75	1.896

Theoretical Analysis



- Suppose $\delta = \beta^T E t$
- Feature extractor: embedding matrix E , can be frozen or adapted
- Outcome map: vector β , can be learned from scratch or adapted

Initial parameter values and learning algorithm are important!

Theorem: $\beta\beta^\top - EE^\top$ is preserved under gradient descent.

[Algorithmic Regularization in Learning Deep Homogeneous Models, Du et al. (2018)]

Suppose initial values (β_0, E_0) , optimal values (β_*, E_*) . Then:

- If β_0 is far from β_* , updating β will **distort** E
- Freezing E might be too restrictive, but unrestricted adaptation too flexible
- Restricting how E can update might strike the right balance if E_0 is **close** to E_*

Restricting the change in embeddings

- Fine-tuning updates initial E_0 to $E = E_0 + \Delta E$
- **Claim:** “Less distortion” of embedding achieved by restricting rank of ΔE
- Change in embedding = $\|(E_0 + \Delta E)t - E_0 t\| = \|\Delta E t\|$. Then by SVD:

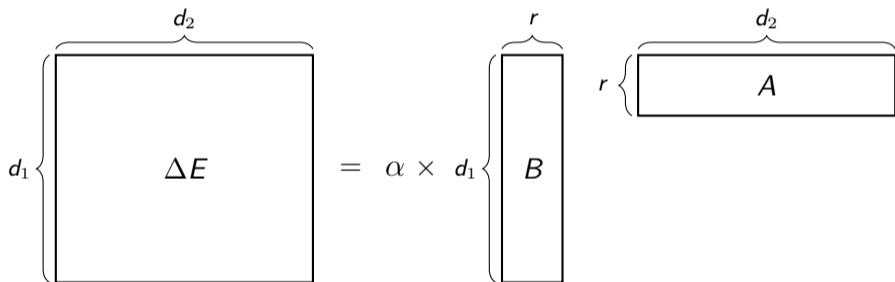
$$\|\Delta E t\| = \|U\Sigma V^T t\| = \sqrt{\sum_{i=1}^r \sigma_i^2 (v_i^T t)^2},$$

where r is rank, σ_i are singular values.

- Implication: restrict rank of ΔE to restrict $\|\Delta E t\|$.

Low-rank adaptation (LoRA), Hu et al. (2021)

- Suppose ΔE is a $d_1 \times d_2$ matrix (typically 4k-12k dimensions in LLMs)
- Impose low-rank structure where $r \ll \min(d_1, d_2)$:



- Past result: $r = 4$ doesn't harm **validation** performance. Cost-effective fine-tuning!
- Result here: LoRA helps **test** set performance

Application 2: designing a new brand

Have shown model is good at [prediction](#). Now I show how to use it for [optimization](#).

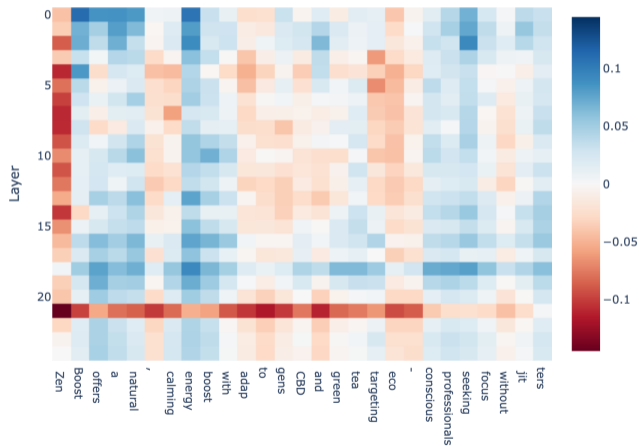
I made up a brand that is distinctly positioned from incumbents.

ZenBoost offers a natural, calming energy boost with adaptogens, CBD, and green tea, targeting eco-conscious professionals seeking focus without jitters.

Optimization in this high-dimensional space seems incredibly complex. Is there hope?

Open the black-box: which parts of the text affect predicted utility?

Change in predicted δ_{jk} from ablating internal LLM activations at each word and layer:



Layer 21 seems to be doing something important.

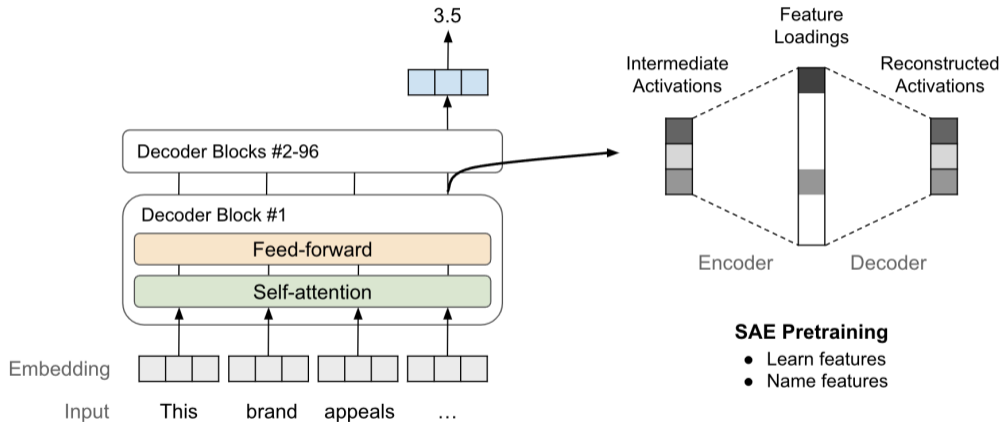
Interpreting internal layers

- Let \mathbf{h} be intermediate values of LLM at a layer
- \mathbf{h} is not interpretable, so project it into a feature space:

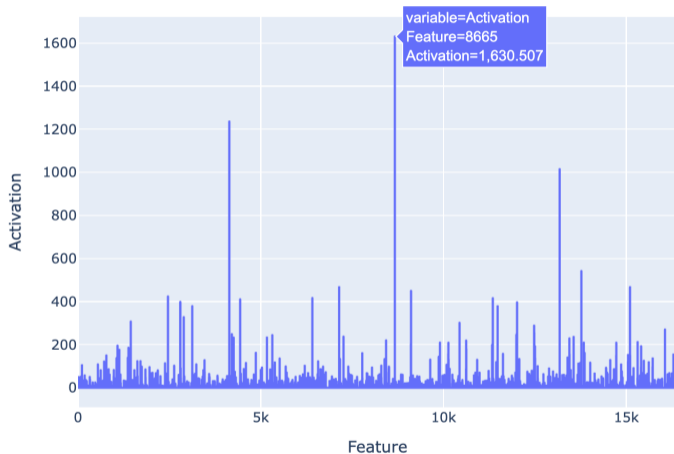
$$\hat{\mathbf{h}} = \sum_{k=1}^K c_k(\mathbf{h}) \mathbf{f}_k$$

- **Result:** imposing **sparsity** on coefficients $c_k(\mathbf{h})$ leads to **interpretable** features \mathbf{f}_k .
[Sparse autoencoders find highly interpretable features in LLMs (Cunningham et al. (2023))]
- In math: $\min_{\mathbf{f}, \mathbf{c}(\cdot)} \|\mathbf{h} - \hat{\mathbf{h}}\|_2^2 + \lambda \|\mathbf{c}(\mathbf{h})\|_1$ leads to interpretable \mathbf{f}_k

Interpreting internal layers with a (pretrained) sparse autoencoder

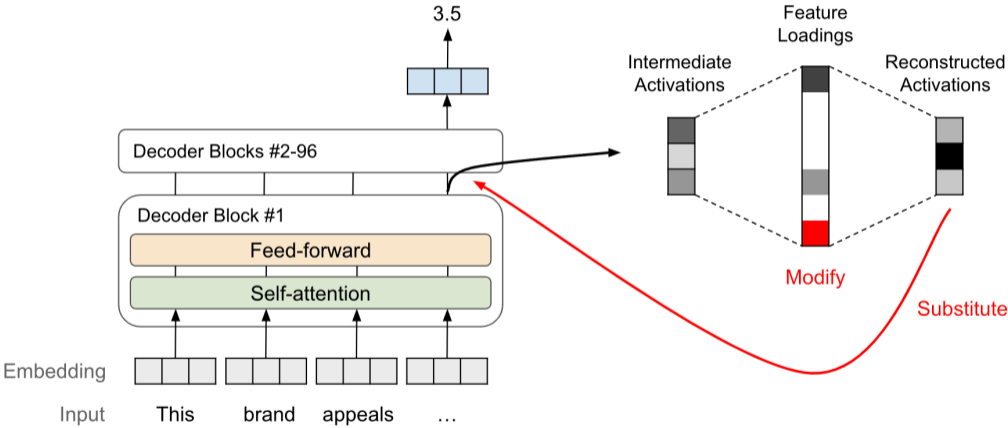


Interpreting layer 21 activations with sparse autoencoder (Gemma-Scope)



Measure feature importance by the change in predicted δ_{jt} from setting loading to 0.

Amplifying features: substitute in reconstructed activations

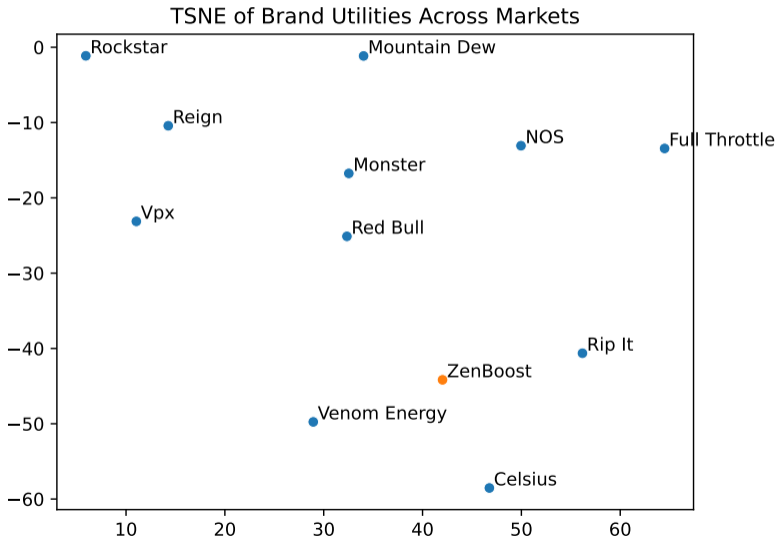


Amplifying the “plant-based” feature

Interpretable optimization of brand position: amplify feature that positively affects δ_{jt} .

	Generated text	Predicted δ_{jt}
Original	ZenBoost offers a natural, calming energy boost with adaptogens, CBD, and green tea, targeting eco-conscious professionals seeking focus without jitters.	-4.21
Amplified	ZenBoost delivers a serene, plant-powered energy lift, blending adaptogens, CBD, and green tea to cater to eco-conscious professionals. Achieve heightened focus without the jitters, all while supporting a sustainable lifestyle.	-3.22

Brand Map: visualizing similarity



Discussion of results

- For new brands, my method gives more informative predictions (correlation \uparrow 3.7x, mutual info \uparrow 17x) compared to traditional embeddings
- For existing brands, text data improves predictions of δ compared to using observable attributes only (MSE \downarrow 54%) and matches performance of fixed effects
- Enabled pricing counterfactuals by combining IV estimates
- Derived interpretable directions to optimize the brand positioning text from internal layers of the LLM

Conclusion

- I develop a scalable method for modeling brand preferences from text that **generalizes** to unseen brands and markets
- Keys to success: adapting embeddings and outcome map that were initialized from a pretrained LLM, imposing low-rank structure on parameter updates
- Application to pricing, entry strategy, and more principled optimization of product positioning

Future: integrate unstructured data into choice models

- Text + images of advertisements to measure effects of ad content
- Transcripts from customer interviews to measure need satisfaction
- Enriched models of mergers

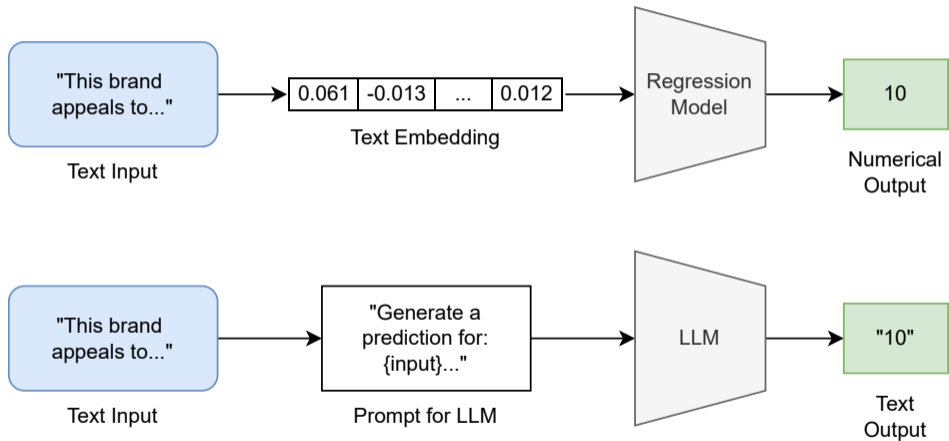
Thank you!

Happy to discuss further: kevin.lee@chicagobooth.edu

Research Agenda

1. Inferring effects of content on consumer behavior/perceptions
 - Causal Alignment: Augmenting Language Models with A/B Tests
[with Panagiotis Angelopoulos and Sanjog Misra]
 - Ad content elasticities
2. Optimization with generative models
 - JMP, Causal Alignment, Performance-safety tradeoffs
3. Human-AI complementarity
 - Causal Alignment
 - Improving Imperfect Decision Makers via State Imputation (startup)
[with Jack Light]

Old vs. new method for quantifying text



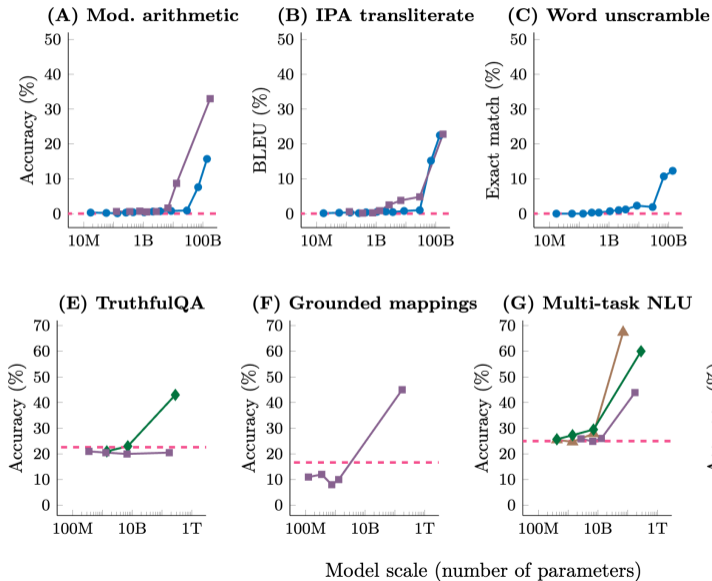
Issue with embeddings: can't *extrapolate*

Suppose we have an “ideal test market”:

- 1 market, 1 product available at a time, observe the proportion that buys
- Fit logit on text embedding: $Pr(y_j = 1) = \frac{1}{1 + \exp(-\delta(\text{text}_j))}$
- Out of sample predictions are similar for **opposite** meanings!

		Predicted Purchase Probability	
		Embedding	ChatGPT
In-sample	care about quality	79.2%	80%
	care about convenience	31.5%	30%
Out-of-sample	care about value	70.5%	50%
	do not care about value	64.9%	20%

Emergence (figure from Wei et al. (2022))



References I

- Berry, S., J. Levinsohn, and A. Pakes (1995). Automobile prices in market equilibrium. *Econometrica* 63(4), 841–890.
- Berry, S., J. Levinsohn, and A. Pakes (2004). Differentiated products demand systems from a combination of micro and macro data: The new car market. *Journal of political Economy* 112(1), 68–105.
- Berry, S. T. (1994). Estimating discrete-choice models of product differentiation. *The RAND Journal of Economics* 25(2), 242–262.
- Cunningham, H., A. Ewart, L. Riggs, R. Huben, and L. Sharkey (2023). Sparse autoencoders find highly interpretable features in language models. *arXiv preprint arXiv:2309.08600*.
- Du, S. S., W. Hu, and J. D. Lee (2018). Algorithmic regularization in learning deep homogeneous models: Layers are automatically balanced. *Advances in neural information processing systems* 31.

References II

- Hoffmann, J., S. Borgeaud, A. Mensch, E. Buchatskaya, T. Cai, E. Rutherford, D. d. L. Casas, L. A. Hendricks, J. Welbl, A. Clark, et al. (2022). Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*.
- Hu, E. J., Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen (2021). Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.
- Lancaster, K. J. (1966). A new approach to consumer theory. *Journal of political economy* 74(2), 132–157.
- McFadden, D. (1974). Conditional logit analysis of qualitative choice behavior. In P. Zarembka (Ed.), *Frontiers in Econometrics*, pp. 105–142. Academic Press.
- Nevo, A. (2001). Measuring market power in the ready-to-eat cereal industry. *Econometrica* 69(2), 307–342.

References III

- Sisodia, A., A. Burnap, and V. Kumar (2024). Express: Generative interpretable visual design: Using disentanglement for visual conjoint analysis. *Journal of Marketing Research*, 00222437241276736.
- Wei, J., Y. Tay, R. Bommasani, C. Raffel, B. Zoph, S. Borgeaud, D. Yogatama, M. Bosma, D. Zhou, D. Metzler, E. Chi, T. Hashimoto, O. Vinyals, P. Liang, J. Dean, and W. Fedus (2022). Emergent abilities of large language models. *Transactions on Machine Learning Research*.