

# Exploring Out-of-distribution Generalization for Text-Matching Recommender Systems

Parikshit Bansal, Yashoteja Prabhu, Emre Kiciman, Amit Sharma

# Text-Matching Recommenders

- Text Matching Recommenders :
  - Both input queries and labels are textual
  - Goal to learn function  $f$  which maps text to vector space
  - Recommendations based on ANNS in vector space : Cosine similarity
  - E.g. : NGAME-M1 module
- E.g, product-product recommendation
  - Short text matching
  - We will focus on LF-AmazonTitles-131K and LF-AmazonTitles-1.3M
  - Conveniently shortened as Amzn131K and Amzn1.3M hereon

# Text-Matching Recommenders

- 
- $Q$  : Input Queries to the system,  $L$  : Set of all labels
  - $P(Q)$  : distribution of input queries
  - $P(L)$  : marginal distribution of labels
  - $P(L|Q)$  : distribution of labels given an input query
  - $P(Q, L)$  : data distribution
- $f_{\theta}(\cdot) : x \rightarrow R^N$  denotes the BERT model, with parameters  $\theta$  mapping sentences to unit norm  $N$  dimensional space
- Similarity between sentences  $x_1, x_2$  is defined as the dot product/cosine similarity between their representations i.e.  $\langle f_{\theta}(x_1), f_{\theta}(x_2) \rangle$  or  $f_{\theta}(x_1)^T f_{\theta}(x_2)$
- Trained using Triplet loss
- We use P@1 as the metric for reporting all the numbers

# OOD Generalization for Text-Matching Recommenders

- Dealing with text-based recommenders :
  - New labels (e.g., new category of products) can be added in a zero-shot manner : Leads to change in  $P(L)$
  - Or new queries can be added by users/inventory: Leads to change in  $P(Q)$
  - $P(L|Q)$  defines user preferences and is assumed to be constant
- Test distribution  $P^*(Q, L)$  is **different** from train distribution  $P(Q, L)$ 
  - OOD (out-of-distribution) test set
- Test distributions **same** as train distribution
  - ID (in-domain) test set
- **Goal** : An OOD generalizable model
  - Does well on both the ID and OOD test set

# Motivating Examples

- Consider training NGAME-M1 on Amzn131K
  - 1000 epochs, 42.70 P@1 on Amzn131K Test set (ID Test)
- Sample queries from Amzn1.3M
  - Find top predicted NGAME-M1 131K label
  - Change in  $P(Q)$
- On unseen keywords like “Reflective Heater” or unseen brands like “Rowe”, the model might predict using spurious signals which don’t hold in OOD

Query	NGAME-M1 Top Predicted Label	Appropriate Label (from Base Model Prediction)
Soleus Air Oscillating <b>Reflective</b> <b>Heater</b>	3M Scotchlite <b>Reflective</b> Tape, Silver, 1-Inch by 36-Inch	Roadpro 12V <b>Heater</b> and Fan with Swing-out Handle
<b>Rowe</b> USA Spoke <b>Wrench</b> - Bagged 09-0001	<b>Rowe</b> nta ZD100 Non-Toxic Soleplate Cleaner Kit	<b>Wrench</b> Set, Open End Metric 4mm-6mm - SCR-913.00

# Bing Shopping Example

- Searching for a query like “*asics belt*” overfits on brand to give irrelevant products as predictions
- Asics shoes get ranked higher than the belts

Microsoft Bing asics belt InPrivate English

ALL IMAGES VIDEOS MAPS NEWS SHOPPING

Bing Shopping > asics belt

Clear all filters

Sort by: Featured About

**BRAND**






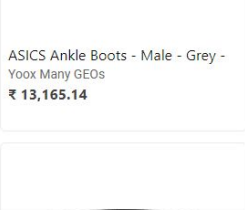
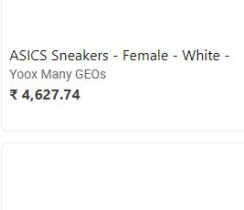
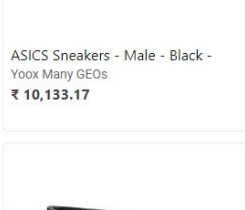
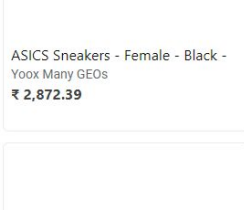
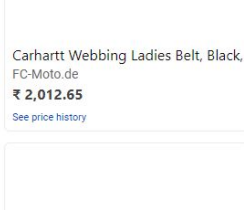
- ☐ Polo Ralph Lauren
- ☐ Apsis
- ☐ Shot
- ☐ Held
- ☐ Spidi
- [+3 more](#)

**PRICE**

- ☐ Up to ₹ 750
- ☐ ₹ 750 - ₹ 1,500
- ☐ ₹ 1,500 - ₹ 2,400
- ☐ ₹ 2,400 - ₹ 4,700
- ☐ Over ₹ 4,700

**SELLER**

- ☐ Myntra.com
- ☐ FC-Moto.de
- ☐ The Collective India
- ☐ Tata CLiQ
- ☐ Ajlo
- [+2 more](#)

 <p>ASICS Ankle Boots - Male - Grey - Yoox Many GEOs ₹ 13,165.14</p>	 <p>ASICS Sneakers - Female - White - Yoox Many GEOs ₹ 4,627.74</p>	 <p>ASICS Sneakers - Male - Black - Yoox Many GEOs ₹ 10,133.17</p>	 <p>ASICS Sneakers - Female - Black - Yoox Many GEOs ₹ 2,872.39</p>	 <p>Carhartt Webbing Ladies Belt, Black, FC-Moto.de ₹ 2,012.65 <a href="#">See price history</a></p>
				

# Related Work

- $P(Q, L) = P(L|Q) P(Q)$
- How can we solve the issues like “Reflective Heater” coming into test set?
- In Vision many of issues are alleviated by augmentations
  - Changing  $P(Q)$  to simulate OOD input distribution
  - Augmentations aim to make predictor invariant of spurious feature
  - rotating, cropping, are some popular *universal augmentations*
- No such *universal, controllable augmentations* in NLP :
  - Word deletion, Swap etc don’t work well
- Augmentations can be done in latent space also : e.g. SimCSE [Gao et al., 2020]
- Identification of spurious features is not clear :
  - Every word in a query might be causal to some extent
  - Brand name, product code, item description etc are all important
- Can we leverage NLP models for augmentations?

# Outline

- Empirical Setup and Analysis of NGAME-M1
- Causal Formulation of the Relevance Function
- Possible Solutions : Intervention and Output-based Regularizers
- Experiments
- Open questions



# NGAME Analysis

- Analysis of Transformer models : Not straightforward
  - Final layer activations : They aren't interpretable
  - Attention Analysis : Attentions are not explanations
  - Saliency : Gradient based, Propagation based, Occlusion based
- Why use attention as explanation when we have saliency?
- Occlusion based Saliency the *most interpretable and simple* choice
  - For classification/regression tasks  $f_{\theta}(x) - f_{\theta}(x'_{-j})$ , where token  $j$  is masked
- For similarity we define :
  - Token-wise Importance Score for token  $j$  is
  - $s_j = (1 - f_{\theta}(x)^T f_{\theta}(x'_{-j}))$

# Base Model vs Finetuned Model Analysis

- Model definitions :
  - BASE Model : **MS-MARCO-DistillBERT-v3**
  - Finetuned Model : **NGAME-M1** finetuned on **Amzn131K** using **Triplet Loss** for **1000 epochs**
- Consider token wise importance for the examples discussed before

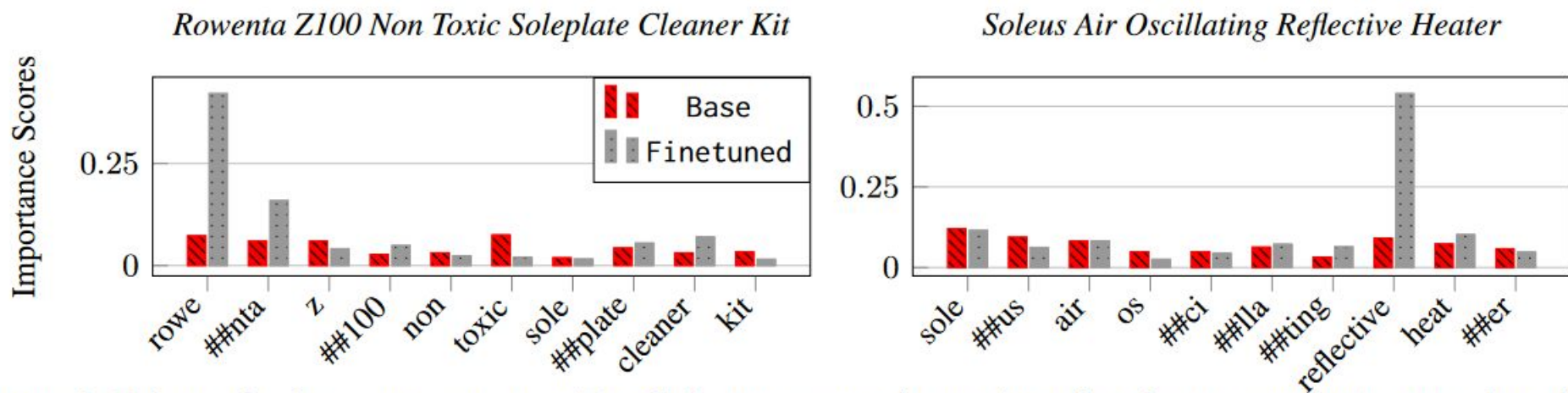


Figure 2: Token-wise importance scores (Eqn 2) for two example queries using the Base and Finetuned models. Base gives approximately equal importance scores to all tokens for both sentences whereas Finetuned model gives disproportionately high weights to the tokens "rowe" and "reflective" in first and second sentence respectively.

# Base Model vs Finetuned Model Analysis

- We also look at their performance on a constructed benchmark
- We use categorical information on Amzn131K to remove certain category of products
- The removed Queries and Labels serve as OOD test set
- While Finetuned model does well on ID setting, Base model is better on OOD, even though Finetuned model was trained on top of Base model

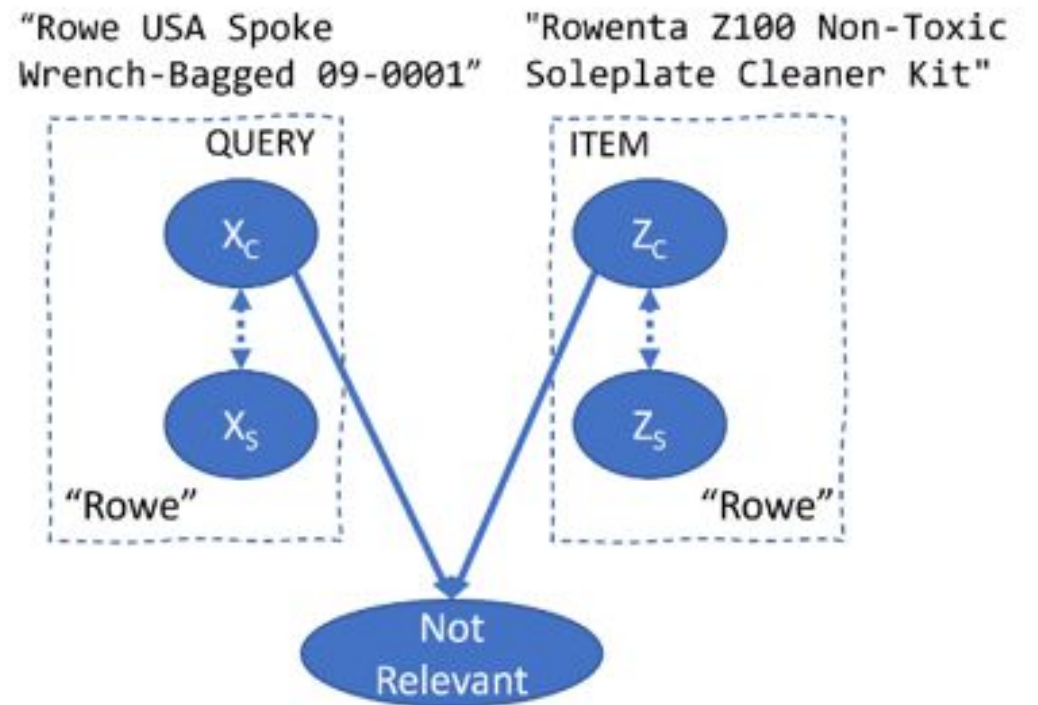
Method	In-Domain	Out-of-Distribution
Base Model	20.01	<b>30.61</b>
Finetuned Model	<b>38.74</b>	28.31

Finetuning *overfits to training distribution* such that some tokens are disproportionately weighted in the representaton.

*Model similarity reduces to **token-matching**.*

# Explaining results through a causal formulation of the relevance function

- $Q$  : Query ( $X$  in fig),  $L$  : Item ( $Z$  in fig)
- Textual representations can be broken down as :
  - $X_c, Z_c$  : Causal part of query/item
  - $X_s, Z_s$  : Spurious part of query/item
- OOD Model should focus on ONLY  $X_c$  and  $Z_c$
- ID model can explore  $X_s$  and  $Z_s$
- Ideal solution in practice requires combination of both with appropriate use cases



*Solid arrow means causation, Dotted arrows mean correlation .*

# Solution 1: Output Regularization (OutReg)

- The goal is to learn representation which corresponds more to  $X_c$
- BASE Model is not trained on the same dataset i.e. doesn't have the same correlations
- BASE Model can hence be leveraged to break these correlations and learn more stable and causal features  $X_c$
- Simplest way to achieve this by making learnt embeddings similar to base embeddings i.e.  $(f_\theta(x) - f_{\theta_0}(x))$ 
  - where  $\theta$  are learnt parameters,  $\theta_0$  are Base model parameters
- If  $L_{ERM}$  is the NGAME loss, the new loss  $L_{Total}$  can be defined as
- $L_{Total} = L_{ERM} + \lambda(f_\theta(x) - f_{\theta_0}(x))^2$ , where  $\lambda$  is a hyper-parameter

# Solution 2: Interventional Regularizer (ITVReg)

- OutReg over-relies on the BASE model i.e.
  - If Base model is bad, the OutReg will move representations away from good solutions
- Since finetuning distorts the occlusion importance of tokens, why not regularize to the base model's importance score?
  - Allows learning from data, without disproportionately amplifying any single token
- We can assume importance scores of tokens as sufficient statistics extracted from the BASE Model
- Instead of regularise the complete embeddings vector, regularise the importance of tokens to the base model

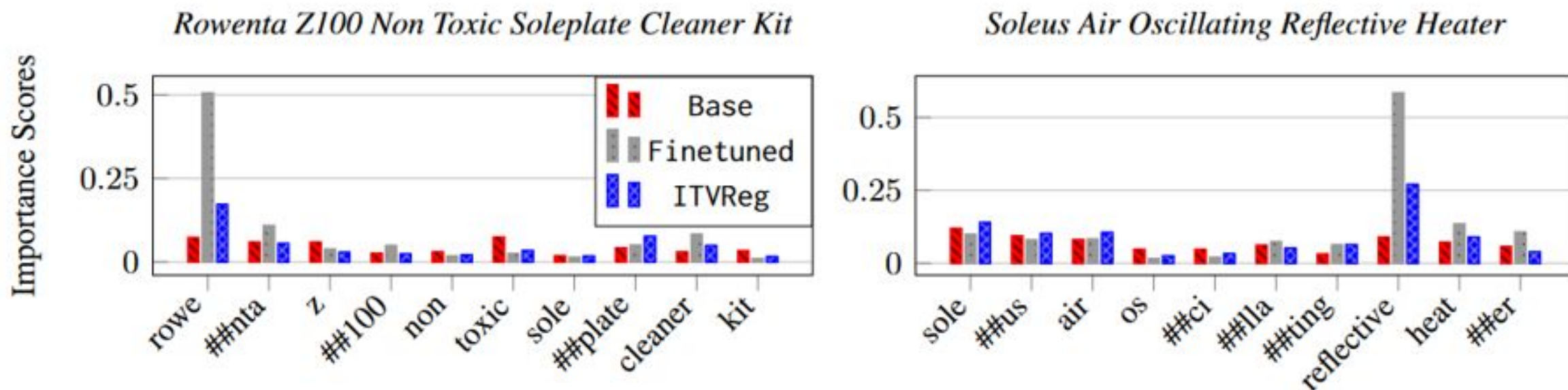
$$\begin{aligned} & [(1 - f_{\theta}(X)^{\top} f_{\theta}(X')) - (1 - f_{\theta_0}(X)^{\top} f_{\theta_0}(X'))]^2 \\ &= (f_{\theta}(X)^{\top} f_{\theta}(X') - f_{\theta_0}(X)^{\top} f_{\theta_0}(X'))^2 \end{aligned}$$

- $L_{Total} = L_{ERM} + \lambda(f_{\theta}(x)^T f_{\theta}(x') - f_{\theta_0}(x)^T f_{\theta_0}(x'))^2$ , where  $x'$  has some of the tokens randomly masked



# Motivating examples with ITVReg

- For the examples presented before, we compare token importance scores to base and finetuned model below





# Empirical evaluation: Does ITVReg lead to better OOD generalization?

## Baselines

No explicit methods for OOD generalisation in text-matching systems

- We construct the following baselines :
  - BASE : The base model
  - Finetuned : Standard Finetuning of NGAME
  - MaskReg : Similar to ITVReg, but regularises to 0 importance score. Loss :  $(f(x)^T f(x') - 1)^2$
  - SimCSE : Forward passes twice with different dropout mask. Loss :  $(f(x, s)^T f(x, s') - 1)^2$

## Setup

Construct two realistic OOD evaluation benchmarks.

- \* **Category Shift:** New categories of queries and labels are added
- \* **Temporal Shift:** New queries and labels are added over time

# Empirical Setup – Categorical Shift

- Construct an OOD setup on top of Amazon131K
  - Filter all labels not having categorical information : 99K remaining after filter
  - Selected 5 categories to remove. 13K labels belong to these, 86K remaining
    - *Automobiles, Kitchen and Dining, Health and Personal Care, Electronics, Tools and Home Improvement*
- Train on the 86K labels and their corresponding train queries
  - Test in-domain uses the test queries of 86K in-domain labels and 99K labels
  - Test OOD uses the test queries of 13K OOD labels and 99K labels
- While OutReg has the best performance on OOD ITVReg improves performance on OOD without compromising ID numbers. ITVReg might serve as a better compromise.

Finetuning Method	In-Domain Test	Out-of-Distribution Test
Base	20.01	30.61
Finetuned	<b>38.74</b>	28.31
MaskReg	37.92	29.09
SimCSE	38.05	28.52
OutReg	37.66	<b>31.21</b>
ITVReg	<b>38.77</b>	29.53

# Experimental Setup – Temporal Shift

- We train the model on Amzn131K collected in 2013
- Setting 1 : Evaluate on Amzn1.3M dataset collected in 2014
  - Since 1.3M is a different task,  $P(Q|L)$  might also be shifting but assumed to be constant

Fine-tuning Method	Temporal Shift	
	Amazon131K (IID)	Amazon1.3M (OOD)
Base	22.50	25.71
Finetuned	<b>39.71 <math>\pm</math> 0.14</b>	26.02 $\pm$ 0.08
MaskReg	39.56 $\pm$ 0.01	26.65 $\pm$ 0.02
SimCSE	39.47 $\pm$ 0.11	26.05 $\pm$ 0.02
OutReg	38.03 $\pm$ 0.53	<b>27.60 <math>\pm</math> 0.03</b>
ITVReg	<b>39.72 <math>\pm</math> 0.10</b>	27.08 $\pm$ 0.01

# Experimental Setup – Simulating temporal evolution

- Start with 131K test set and progressively add items from 1.3M (and their corresponding queries) to test set
- Generally these models are retrained periodically to account for distribution shifts (addition of labels and queries in this case)
- This plot shows how these models will do without retraining if they were deployed with x axis being temporal dimension
- Y-axis is difference between performance of difference methods with Finetuned
- ITVReg is always better than finetuned
- OutReg hurts performance in ID setting (left extreme) but is better than ITVReg in OOD (right extreme)

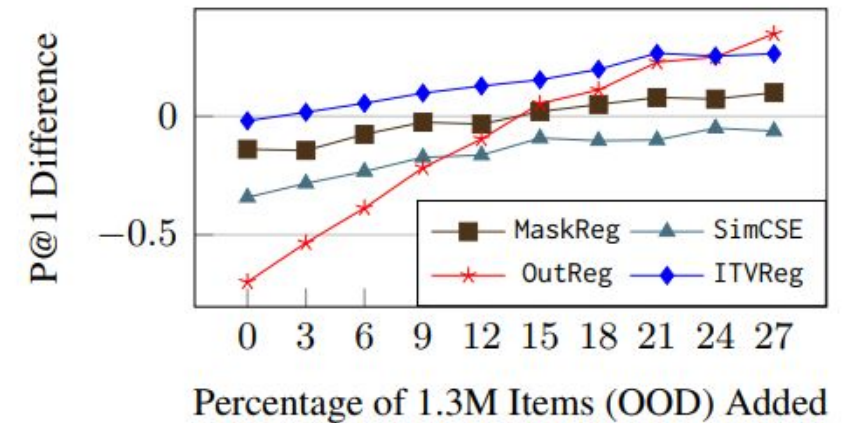


Figure 3: To simulate temporal evolution, we start with *Amazon131K* and add 39K new items from *Amazon1.3M* dataset at each tick. ITVReg is the only method that is consistently better than Finetuned on P@1.

# Qualitative Predictions

**Query:** *Rowe USA Spoke Wrench – Bagged 09-0001*

Method	Top 5 Predicted Items
Base	Ridgid 31105 24-Inch Aluminum Pipe <b>Wrench</b>
	Ridgid 31115 48-Inch Aluminum Pipe <b>Wrench</b>
	Ridgid 31110 36-Inch Aluminum Pipe <b>Wrench</b>
	Ridgid 31100 18-Inch Aluminum Pipe <b>Wrench</b>
	Craftsman 9-41796 Ratcheting Ready Bit Screwdriver
Finetuned	Rowenta ZD100 Non-Toxic Soleplate Cleaner Kit
	Rowenta DR5015 800 Watt Ultra Steam Brush with Travel Pouch
	Rowenta(R) Stainless Steel Soleplate Cleaning Kit ZD-110
	Rowenta DR6015 Ultrasteam Hand-Held Steam Brush with Travel Pouch, 800-watt
	Rowenta DR6050 Ultrasteam Hand-Held Steam Brush Dual-Voltage with Travel Pouch, 800-watt
OutReg	Rowenta DR6015 Ultrasteam Hand-Held Steam Brush with Travel Pouch, 800-watt
	Rowenta DW4060 Auto Steam Iron 1700W with Airglide Stainless Steel Soleplate Auto-off Anti-Scale, Blue
	Rowenta DR5015 800 Watt Ultra Steam Brush with Travel Pouch
	Rowenta VU2531 Turbo Silence 4-Speed Oscillating Desk Fan, 12-Inch, Bronze
	Rowenta(R) Stainless Steel Soleplate Cleaning Kit ZD-110
MaskReg	Rowenta(R) Stainless Steel Soleplate Cleaning Kit ZD-110
	Rowenta ZD100 Non-Toxic Soleplate Cleaner Kit
	Rowenta DG8430 Pro Precision Steam Station with 400 hole Stainless Steel soleplate 1800 Watt, Purple
	Rowenta DR5015 800 Watt Ultra Steam Brush with Travel Pouch
	Rowenta DR6015 Ultrasteam Hand-Held Steam Brush with Travel Pouch, 800-watt
SimCSE	Rowenta RH8559 Delta Force 18V Cordless Bagless Energy Star Rated Stick Vacuum Cleaner ...
	Rowenta ZD100 Non-Toxic Soleplate Cleaner Kit
	Rowenta(R) Stainless Steel Soleplate Cleaning Kit ZD-110
	Rowenta DR6015 Ultrasteam Hand-Held Steam Brush with Travel Pouch, 800-watt
	Rowenta DR6050 Ultrasteam Hand-Held Steam Brush Dual-Voltage with Travel Pouch, 800-watt
ITVReg	<b>Wrench</b> Set, Open End Metric 4mm-6mm - SCR-913.00
	Craftsman 6 pc. Universal <b>Wrench</b> Set - Metric
	Tusk Spoke <b>Wrench</b> Set
	Crescent RD12BK 3/8-Inch Ratcheting Socket <b>Wrench</b>
	Allen <b>Wrench</b> Set, 10 Pc. Heavy Duty, Extra Long 9 T-handle, Metric Sizes

Table 8: Top 5 predicted items for the query *Rowe USA Spoke Wrench - Bagged 09-0001* given by various methods sorted by relevance. Correct items should be about *wrench* and ITVReg and Base model both give the same. Other models rely on spurious feature i.e. *Rowe* for predicting items, which leads to wrong results



# Results on sentence matching benchmarks

- Thakur et al. Proposed OOD datasets for sentence matching.
  - Sentence similarity tasks
  - Question Recommendations specifically e.g. Quora Question Pairs
- Datasets with different losses : MSE loss, Contrastive loss, Triplet loss
- Mixed results observed:
  - OutReg is good for OOD if base model is good on OOD
  - ITVReg helps in ID setting by acting as a regulariser (avoiding too high weights)
  - ITVReg better than MaskReg in OOD if base model is good on OOD
    - But OutReg is better than ITVReg in these cases
  - MaskReg gets better numbers than ITVReg if base model is bad

# Future Work – Limitations

- A major assumption is that ‘importance scores’ is a sufficient statistic to regularise with, which may not be true
- We still rely on the base model. Ideal OOD methods should be able to capture stable signals from dataset itself
  - Usually people take data from different environments (i.e. having different correlations, but same causal features) to learn these stable features.
- Improvement in P@1 are marginal for Amzn131K and mixed results observed for other sentence similarity datasets

# Future Work –semi-synthetic dataset for developing a causal method

- Major issue faced was analysing Amazon is hard
  - Defining valid predictions in case of OOD queries is hard
  - For queries like “Nike Running Shoes” is recommending “Nike watch” bad?
- Ideally would like to have a controllable setup
- Generally in vision people bias an input feature (like background of image) with the output label
  - E.g. you want to a classifier which classifies bird pictures as LandBirds or WaterBirds. WaterBirds generally have a blue background which is hence correlated with label and captured by label. Breaking this correlation in test set serves as OOD test
- Took the EURLex dataset and constructed a synthetic dataset
  - Majority of queries had date in them. We modified the dates of the queries corresponding to a selected label to have a particular month more often (90%) than other months
  - Changing month distribution during test time creates OOD test set



# Future Work -

- Combination of ID and OOD models
  - People generally are fine exploiting spurious features for getting extra ID P@1
  - We would like to develop methods which can on demand (or during inference) switch from exploiting spurious to using only causal
- Propensity scoring methods have been shown to work well previously
  - Propensity weighing (i.e., weighing terms by their propensity while computing loss) doesn't work well with deep models i.e., **doesn't** give unbiased models
  - Propensity based Data Loading (i.e., sampling while drawing indices) though does lead to unbiased models
  - Still exploring why weighing doesn't work and if that can lead to some insights