Joint Query-Product Modeling using Adversarial Transfer Learning

Suchismit Mahapatra[†] **& Amin Mantrach & Suju Rajan**

[†] University at Buffalo

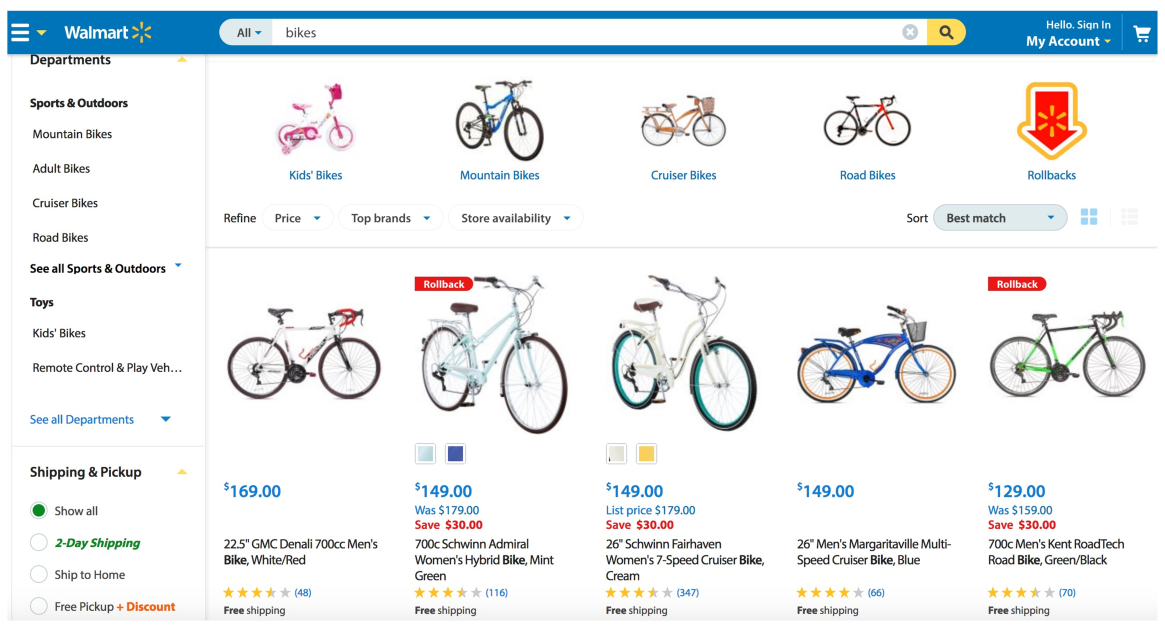
criteo research

Abstract

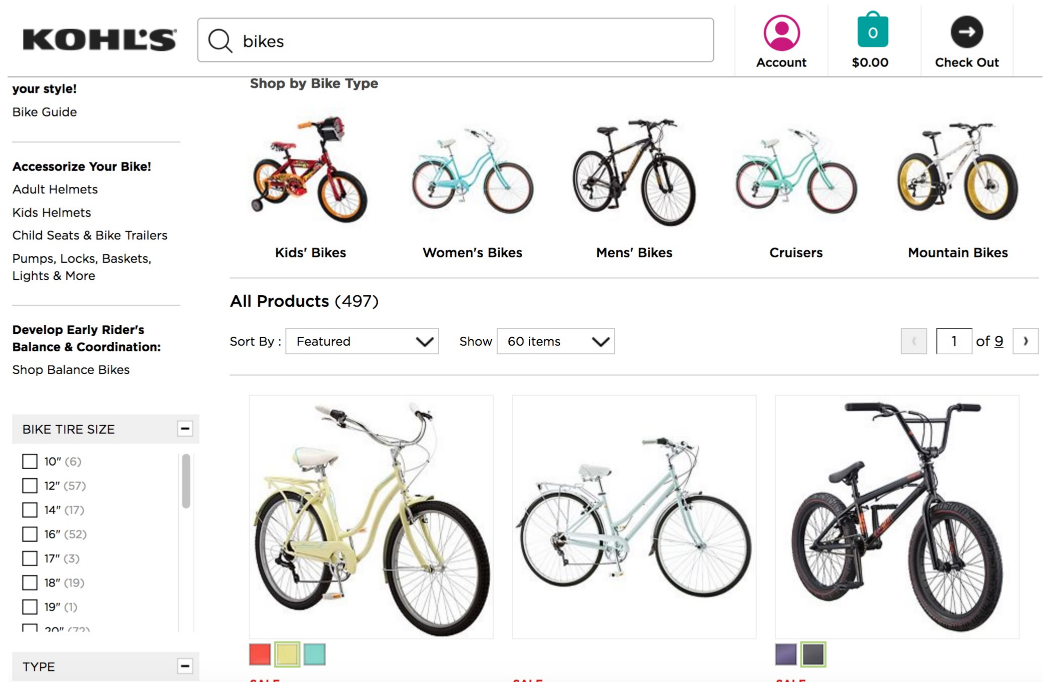
Product search is a very important part of most retailer web sites. While searching for products, the end user expresses his intent via search queries. However, matching relevant products to explicit user intent as formulated by the search query may be challenging as the user may make use of keywords that are not necessarily related to the product description itself. Hence, there is a strong need of understanding the query-product matching beyond the syntactic level.

Context

- Can we improve the *quality* of search results by *transferring* knowledge across different domains ?
- Can user *search behavior* on a specific platform (i.e. Amazon) help to improve the user *experience* on *another* platform ?



(a)



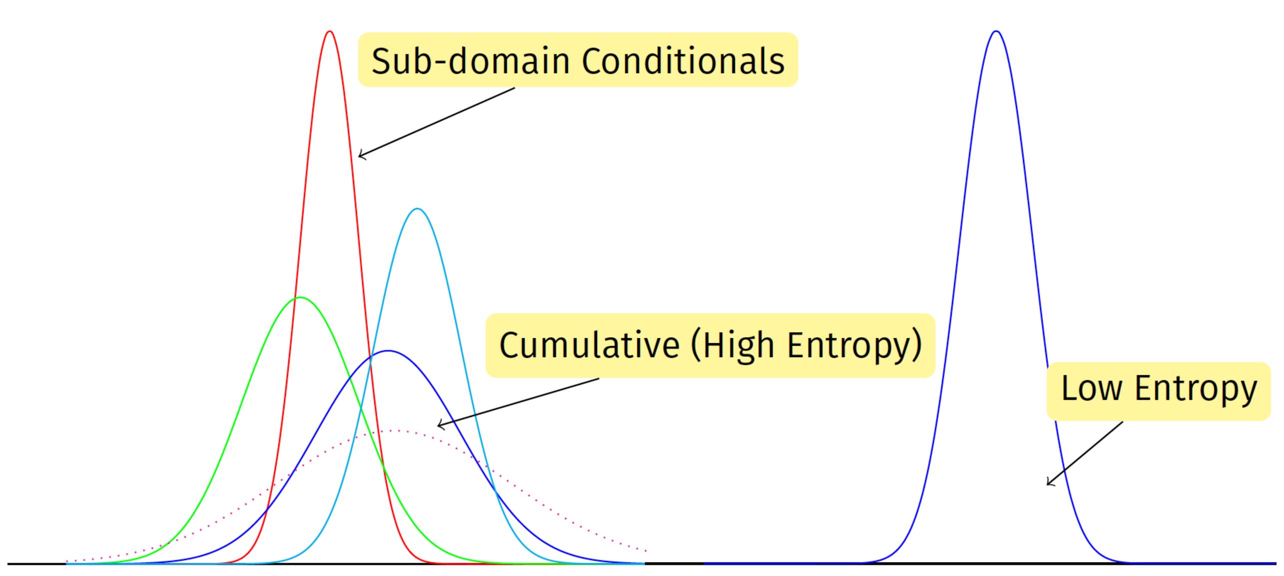
(b)

Can we use the search platform of different retailers to improve overall user experience ?

Objective

- How can we improve/augment the Query-Product model using data from different retailer domains via Adversarial techniques in the context of Transfer Learning ?

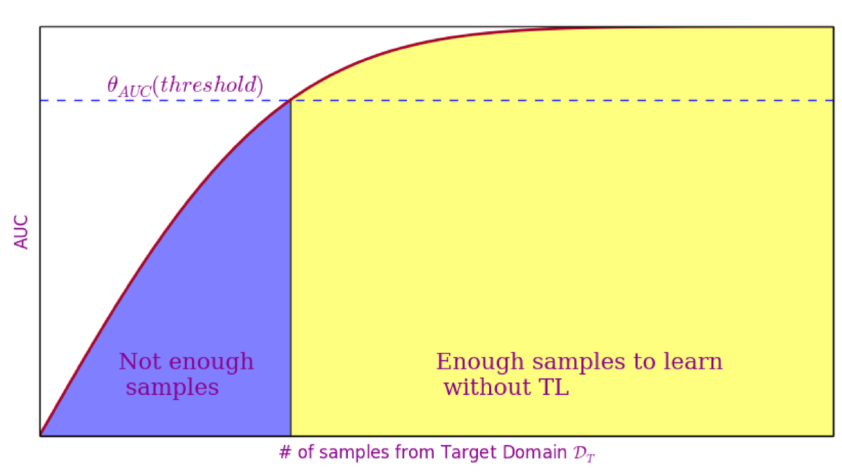
Theory



(c)

Can we transfer knowledge from low-entropy distributions to high-entropy distributions better or vice-versa?

Transfer Learning



(d)

Transfer Learning is generally used when we do not have enough data for $\mathcal{D}_T$ and we want to leverage $\mathcal{D}_S$ to augment our learning in $\mathcal{D}_T$.

Domain Adaptation

The Domain Adaptation learning problem is defined as follows :-

Definition 0.1. Given an input space X and $Y = \{1, 2, \ldots, L\}$, the set of labels as well as $\mathcal{D}_S$ (source domain) and $\mathcal{D}_T$ (target domain) which are distributions over $X \times Y$, the goal of an unsupervised domain adaptation learning algorithm is to build a classifier $\eta : X \to Y$ with a low target risk $R_{\mathcal{D}_T}(\eta) = \underset{(x,y)\sim\mathcal{D}_T}{Pr}(\eta(x) \neq y)$.

The concept of $\mathcal{H}$-divergence (Ben David et al., 2006) can defined as follows :-

Definition 0.2. Given two domain distributions $\mathcal{D}_S$ and $\mathcal{D}_T$ and a hypothesis class $\mathcal{H}$, the $\mathcal{H}$-divergence between $\mathcal{D}_S$ and $\mathcal{D}_T$ is :-

$$d_{\mathcal{H}} \stackrel{def}{=} 2 * \sup_{\eta\in\mathcal{H}} \left| \underset{x^s\sim\mathcal{D}_S}{Pr}[\eta(x^s) = 1] - \underset{x^t\sim\mathcal{D}_T}{Pr}[\eta(x^t) = 1] \right|.$$

- The $\mathcal{H}$-divergence measures the ability of an hypothesis class $\mathcal{H}$ to discriminate between source $\mathcal{D}_S$ and target $\mathcal{D}_T$ distributions.
- In general, it is not possible to measure $\underset{x^s\sim\mathcal{D}_S}{Pr}[\eta(x^s) = 1]$ and $\underset{x^t\sim\mathcal{D}_T}{Pr}[\eta(x^t) = 1]$ directly. However, empirical versions can be approximated by using training examples :-

$$\hat{d}_{\mathcal{H}}(\mathcal{D}_S, \mathcal{D}_T) \stackrel{def}{=} 2\max_{\eta\in\mathcal{H}} \left| \frac{1}{m}\sum_{i=1}^{m} \mathcal{I}[\eta(x_i^s) = 1] - \frac{1}{n}\sum_{j=1}^{n} \mathcal{I}[\eta(x_j^t) = 1] \right|$$

Results

Retailer	# of positive samples	# of products
Toys"R"Us	1729293	31503552
Kohl's	5065947	295128739

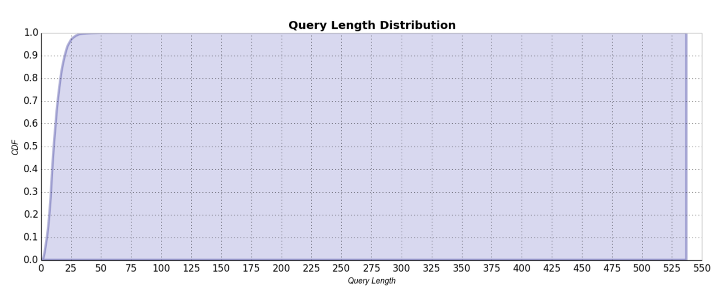
query	product	clicked
britax+marathon	Britax Marathon Car Seat	0
hooks	Wonderart Latch Hook Kit	0
marvel+action+figures	Marvel Legends Series Figure	0
old+skool	Old Skool Nintendo Joystick	0

(e)

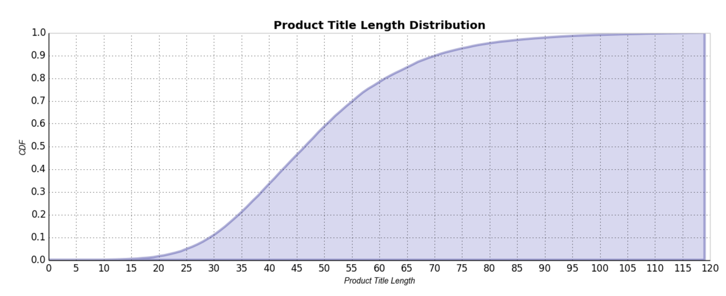
query	product	clicked
mop	Libman White Tornado Twist Mop	0
mens+watches	Armitron Men's Gold Watch	0
power+ranger	Power Rangers Ninja Steel	0
swiffer+wetjet	Swiffer WetJet Starter Kit	0

(f)

Toys"R"Us and Kohl's sample data

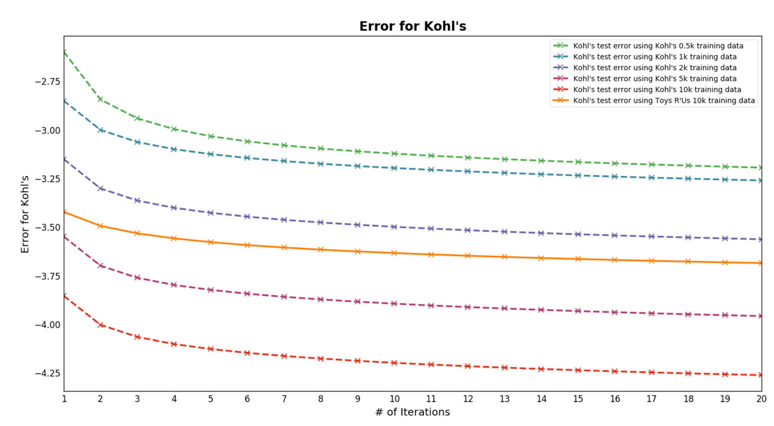


(g)



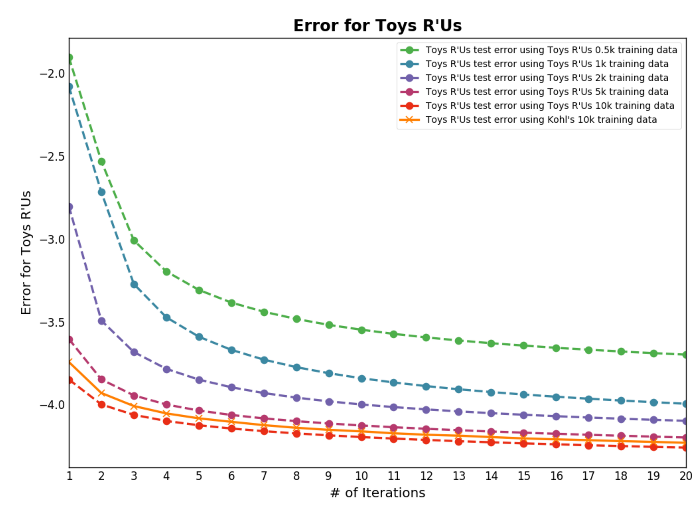
(h)

Query and Product Description length distributions



(i)

Transfer learning from Toys"R"Us to Kohl's



(j)

Transfer learning from Kohl's to Toys"R"Us

Conclusion

It is possible to use transfer learning in the context of user search across domains and there is benefit in transferring knowledge from both low-entropy to high-entropy distributions and vice-versa. Even though the latter worked better above. Can we learn some form of an universal oracle using the above strategy with adversarial learning ?