Seasonal Relevance in E-Commerce Search

Haode Yang
Amazon
Seattle, WA, US
haodyang@amazon.com

Parth Gupta
Amazon
Palo Alto, CA, US
guptpart@amazon.com

Roberto Fernández Galán
Amazon
Seattle, WA, US
galanrob@amazon.com

Dan Bu
Amazon
Seattle, WA, US
budan@amazon.com

Dongmei Jia
Amazon
Seattle, WA, US
djia@amazon.com

ABSTRACT

Seasonality is an important dimension for relevance in e-commerce search. For example, a query `jacket` has a different set of relevant documents in winter than summer. For an optimal user experience, the e-commerce search engines should incorporate seasonality in product search. In this paper, we formally introduce the concept of seasonal relevance, define it and quantify using data from a major e-commerce store. In our analyses, we find 39% queries are highly seasonally relevant to the time of search and would benefit from handling seasonality in ranking. We propose LogSR and VelSR features to capture product seasonality using state-of-the-art neural models based on self-attention. Comprehensive offline and online experiments over large datasets show the efficacy of our methods to model seasonal relevance. The online A/B test on 784 MM queries shows the treatment with seasonal relevance features results in 2.20% higher purchases and better customer experience overall.

CCS CONCEPTS

• **Information systems** → **Electronic commerce**; **Learning to rank**; *Data mining*; • **Computing methodologies** → **Natural language processing**; *Neural networks.*

KEYWORDS

Seasonality; E-commerce search; Learning to rank; Natural language processing; Self-attention mechanism

ACM Reference Format:
Haode Yang, Parth Gupta, Roberto Fernández Galán, Dan Bu, and Dongmei Jia. 2021. Seasonal Relevance in E-Commerce Search. In *Proceedings of the 30th ACM Int'l Conf. on Information and Knowledge Management (CIKM '21), November 1–5, 2021, Virtual Event, Australia.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3459637.3481951

1 INTRODUCTION

Product discovery in e-commerce is mainly attributed to search and recommendation. Product relevance in e-commerce search further depends on various dimensions such as query, user, time and context [25]. While the user, query and context dimensions are well captured in information retrieval research and are incorporated in e-commerce search engines, the time aspect is under addressed, especially from relevance perspective. As a comparison, time is well studied in web search under the area of temporal information retrieval [3]. Several studies are devoted to profile temporal dimensions of queries [14, 18, 23], such as their sensitivity to time. Another track of research incorporates temporal information with web search ranking [5, 7, 8, 12]

Seasonality in e-commerce search plays an important role. A query `jacket` has a different set of relevant documents in winter than in summer. However, with a limited context and an open ended query like `jacket`, the onus is on the search engine to show more seasonally relevant documents, or to the least not get over indexed to the user behavior recorded during the preceding season. Therefore, it is important for the search engines to be season aware and to incorporate such information in ranking. In e-commerce, seasonality has also importance beyond search. For example, logistics can leverage seasonality signal for demand forecasting and inventory management [11].

In this paper, we present a detailed study on seasonality as a dimension of relevance for e-commerce search engines. We present approaches to identify seasonality in queries and products, and define features that capture it. These features can be consumed during standard learning-to-rank (LTR) framework. Finally, we show experiments around search which measure the effectiveness of the features. Comprehensive offline and online experiments reveal importance of handling seasonality in e-commerce search by improved metrics, including 0.62% more clicks, 1.22% more add-to-carts and 2.20% more purchases.

We quantify seasonality of queries and products using query volume and product sales respectively, based on which seasonal relevance is defined. According to the proposed definition, 39% queries are highly seasonally relevant to the time of search and 42% of total purchases in a year are made following those queries. From the perspective of products, they on average are highly seasonally relevant to 35% time in a year, during which they drive up to 48% of total annual sales. We adopt a predictive approach to model seasonal relevance so that applications are not constrained by the cold-start problem and that noise in data is reduced. In our approach, we frame it as a language modeling task of learning seasonal relevance from query text and product titles, respectively. Among all types of product information, we pick product title to model because it

CIKM '21, November 1–5, 2021, Virtual Event, Australia.

ACM ISBN 978-1-4503-8446-9/21/11.
https://doi.org/10.1145/3459637.3481951

has 100% coverage and contains key product attributes that help profile a product's seasonal properties, such as sleeve length, color, fabrics etc. The learning task is performed by using pre-trained text embeddings that capture semantics of text and a neural network architecture that uses the self-attention mechanism [26] to identify words in product titles that are critical to predict seasonal relevance. Our neural network model achieves a 4.5% better performance than the baseline does, and has learned to detect not only products which are frequently purchased in each of the four seasons, but also those intended for special occasions such as Christmas and back-to-school period. We derive ranking features based on predictions of the neural network model and train LTR rankers with those features to make search ranking season aware.

We evaluated the performances of the proposed methods both offline and online. In the first case, we measured how ranking metrics such as NDCG change when seasonal relevance is incorporated. In the second, we ran an A/B test on 784 MM searches with and without our changes. Experimental results suggest that the presented method significantly increases the purchase rate, which underlines the improved customer experience achieved via seasonal relevance.

The rest of the paper is organized as follows. We present the nuances of seasonality from e-commerce search perspective in Section 2. We describe our method to capture seasonality signal in terms of features and corresponding analyses in Section 3. The experiments to discuss the efficacy of proposed methods in search and results from offline and online experiments are presented in Section 4. Finally, in Section 5 we present the related work and we make concluding remarks in Section 6.

2 SEASONALITY AND RELEVANCE

Products in e-commerce may be seasonal (e.g. `rain jacket`) or evergreen (e.g. `jeans`). We carry our study on fashion categories of a major e-commerce store. Sales in fashion categories follow both seasonal and holiday patterns, reflecting the change in customer preference for product types and fashion styles throughout the year. For example, two dresses with different styles had dramatically different patterns in their monthly sales, as shown in Figure 1. E-Commerce search engines should incorporate such user preferences while modeling relevance. Towards this, we define seasonal relevance for queries and products, and describe details of estimating it from data. In Section 3, we present a predictive approach of learning seasonal relevance from query text and product titles.

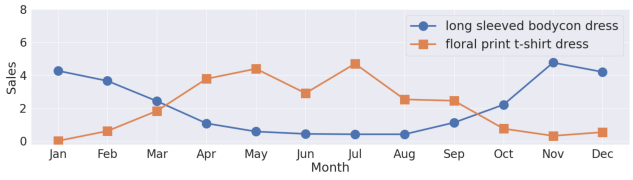


Figure 1: Monthly sales of a `floral print t-shirt dress` (■) and a `long sleeved bodycon dress` (●). The y-axis has been rescaled to omit absolute numbers.

2.1 Definition of Seasonal Relevance

We derive seasonal relevance of a product from its sales, with the intuition that the demand and thus the sales of a product goes up when it becomes in-season, reflecting customers' perception of seasonal relevance. Similar to [23, 24], we take a time unit of a month. Suppose E is a purchase event, A is the product purchased and M is the month of purchase. For any pair of product a and month m where $m \in \{1, ..., 12\}$, we define the seasonal relevance as

$$P_{am} = P(M = m | A = a)$$

and $\mathbf{P_a} = (P_{a1}, ..., P_{a12})$ is essentially a probability distribution. Given a product a, P_{am} can be estimated by the proportion of its annual sales concentrated in month m. Instead of using its raw monthly sales, we normalize the numbers with the overall sales of the month in order to isolate the trends in product sales from those caused merely by a change in overall sales [14]. Formally, we use

$$Q_{am} = \frac{\frac{S_{am}}{S_m}}{\sum_{m'=1}^{12} \frac{S_{am'}}{S_{m'}}}$$

as an estimator of P_{am}, where S_{am} is the sales of product a in month m and S_m is the overall sales of the month. Vector $\mathbf{Q_a} = (Q_{a1}, ..., Q_{a12})$ is named as product monthly sales concentration (MSC). For queries, we follow the same formulation. The seasonal relevance between a query and a month is defined as the probability of seeing the query in the given month conditioned on its occurrence, and can be estimated with query volume. $\mathbf{Q_a}$ then becomes query monthly volume concentration (MVC).

3 APPROACH

In this section, we present a predictive approach to model seasonal relevance. Our approach applies to both products and queries. To avoid repetition, we discuss product seasonal relevance in detail and the same for queries can be derived accordingly.

3.1 Modeling Seasonal Relevance

As discussed in Section 2, seasonal relevance for product a in a month m can be defined as P_{am}. Estimating P_{am} from data through Q_{am} has two potential issues: *(i)* it applies only to products with historical sales; *(ii)* product sales in a specific month as used in Q_{am} calculation can be noisy for various reasons such as discoverability and user behavior. For example, as shown in Table 1, the same down jacket with 2 different sizes had evidently different MSCs. While both were most popular from October to February, one was skewed to the end of the year and the other was skewed towards the beginning of the year. To tackle the aforementioned cold start problem and reduce noise, we take a predictive approach to learn P_{am} from product titles and Q_{am}. Product titles in an e-commerce store typically contain key attributes of each product, such as sleeve length, color, fabrics etc. These attributes help profile a product's seasonal properties, and thus should help in predicting P_{am}.

Suppose we observe a set of products $\mathbb{A}$, along with $\mathbb{Q} = \{\mathbf{Q_a} : a \in \mathbb{A}\}$ and product titles $\mathbb{X} = \{X_a : a \in \mathbb{A}\}$. The seasonal relevance of these products is unknown and is denoted as $\mathbb{P} = \{\mathbf{P_a} : a \in \mathbb{A}\}$. Assume there is a function $f(\theta) : \mathbb{X} \rightarrow \mathbb{P}$ that is parametrized by θ and maps a product title X_a to a seasonal relevance vector $\mathbf{P_a}$. We

Table 1: One pair of product MSCs computed from sales data. High seasonal relevance (> 0.09) is highlighted in bold.

Product	m=1	m=2	m=3	m=4	m=5	m=6	m=7	m=8	m=9	m=10	m=11	m=12
down jacket (S)	**0.319**	**0.179**	0.050	0.007	0.001	0.002	0.002	0.009	0.031	**0.123**	**0.146**	**0.132**
down jacket (XL)	**0.206**	**0.105**	0.044	0.005	0.004	0.003	0.000	0.004	0.046	**0.206**	**0.217**	**0.161**

learn f by minimizing

$$L(\theta) = \frac{1}{|\mathbb{A}|} \sum_{a \in \mathbb{A}} l(\mathbf{Q_a}, f(X_a; \theta))$$

where l is the following cross-entropy loss, since $\mathbf{Q_a}$ and $f(X_a; \theta)$ can be viewed as two probability distributions.

$$l(\mathbf{u}, \mathbf{v}) = -\sum_{m=1}^{12} u_m \log v_m$$

Our learning task requires transforming text into numerical values and we follow the state-of-the-art practice to use dense vector representations of words [19, 20]. Specifically, we use FastText embeddings [1] given that *(i)* they handle out-of-vocabulary and low-frequency words well and offer good results for noisy text such as product titles in e-commerce [10]; and *(ii)* they are lightweight and improve efficiency of the system. We use neural networks to model function f because of their proven track record of semantically modeling text for downstream tasks. The architecture of the model is shown in Figure 2.

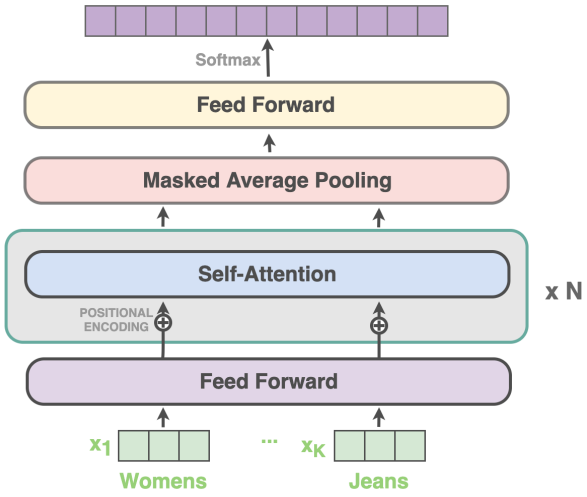


Figure 2: Neural architecture of product monthly sales concentration (MSC) model.

The input is a product title tokenized and represented in FastText embeddings. The subsequent feed-forward and self-attention layers [26] are used to extract the relationship of words in the product title before reducing them into one single embedding for the final prediction layer. We also compared the architecture with three other choices: a feedforward network that directly ingests sentence-level embeddings, a recurrent neural network GRU [6] and BERT [9]. To have a fair comparison, we ran hyper-parameter

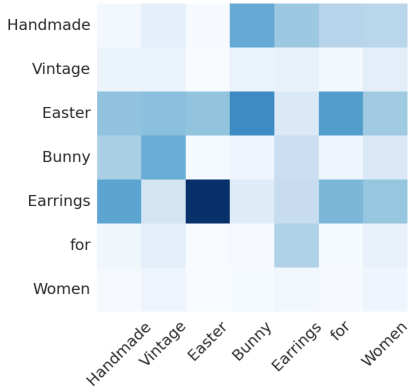


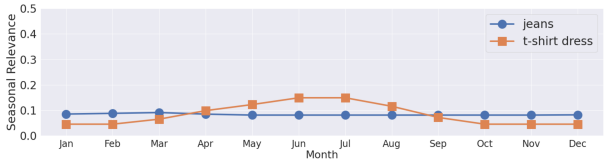
Figure 3: Self-attention weights from the product MSC model for a product titled `Handmade Vintage Easter Bunny Earrings for Women`. A larger weight is denoted by a darker color.

tuning and used the best result for each. The feedforward network underperformed our model by 0.80% in cross-entropy loss on the test dataset, possibly due to the lack of a mechanism to study the relationship between tokens in product titles. GRU, on the other hand, can model sequential relationships. However, it underperformed as well, by 0.55%, likely because product titles in an e-commerce store are less structured than regular language text is. BERT was the best performer among the three but still had a gap of 0.39% with our model. While it can be best utilized when being fine-tuned, that approach was overly heavy-weight for our task and sometimes led to difficulties in convergence. When we took the alternative of freezing its parameters, BERT lost the advantage of generating task-adapted context-aware embeddings. One common property shared by BERT and our model is that both use the self-attention mechanism, which we found to be particularly helpful in capturing information related to holidays such as Easter and Valentine's Day. In Figure 3, we show the self-attention weights generated by our model for a product titled `Handmade Vintage Easter Bunny Earrings for Women`. A darker color means a higher attention weight, and thus a bigger impact to subsequent layers, including the final prediction. As one would expect, `Easter` was given a large weight and treated as a critical token to predict seasonal relevance.

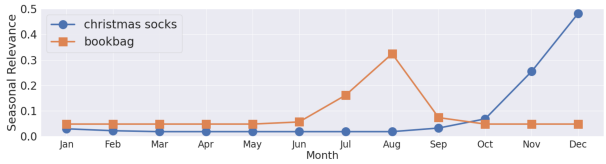
Our final model has 4.4 K trainable parameters with 2 self-attention layers, each with 4 heads. The model was trained with Adam optimizer with a learning rate of 0.001 and a dropout rate of 0.1. To construct the datasets for training and testing, we took the annual sales data for a calendar year and grouped products and months to build the MSC vector $\mathbf{Q_a}$ for each product. Sampling was then performed, giving all products an equal probability.

Table 2: Performance of product MSC model on 4 datasets. Numbers in parentheses are the gains relative to the baseline. A negative gain in cross-entropy loss and a positive gain in cosine similarity mean a better performance.

Dataset	Year	Size	Loss	Cos
Training		0.8M	2.360 (-5.03%)	0.824 (+8.67%)
Validation	2019	0.26M	2.361 (-4.99%)	0.823 (+8.63%)
Test 1		0.25M	2.361 (-4.99%)	0.823 (+8.58%)
Test 2	2020	1.6M	2.372 (-4.54%)	0.795 (+7.80%)



(a) It is predicted that a `t-shirt dress` (■) sells more during summer, whereas a pair of `jeans` (•) should have a flat sales distribution.



(b) It is also predicted that a bookbag (■) will have a sales spike in the back-to-school period near August, and a pair of `christmas socks` (•) starts gaining popularity in November, and reaches the max in December before going completely out-of-season in January.

Figure 4: Predictions of the product MSC model.

We evaluate the neural model with regard to the effectiveness of predictions to the actual MSC vector $\mathbf{Q_a}$. The performance is measured in terms of cross-entropy loss and cosine similarity. Table 2 summarizes the results on 4 datasets. Given our novel definition of the learning task, we did not find previous work to compare our model with, to the best of our knowledge. Therefore, we use the uniform distribution as our baseline, relative to which we report the gains of our model in parentheses. The model was trained on data from 2019. On a test dataset from 2020, we saw a slight drop in model performance but still materially better than the baseline. The drop could be due to a drift in customer behavior and preferences.

Qualitatively, our neural model has learned to distinguish seasonal products from those with a flat sales distribution, e.g. `t-shirt dress` and `jeans` in the top graph of Figure 4. Additionally, it identifies products that are popular for special seasons such as Christmas and back-to-school period, with or without an explicit mention of the intended occasion in the product title, e.g. `christmas socks` and bookbag in the bottom graph of Figure 4.

4 EXPERIMENTS AND RESULTS

In this section, we first present our observations around query seasonality and product seasonality. We give hypotheses on the customer behavior behind. Our analyses are based on the seasonal relevance modeled by our predictive approach instead of that estimated directly through data, because the latter is too noisy to draw reliable conclusions, as discussed previously. We then discuss the impact to e-commerce search by incorporating product seasonal relevance we have modeled into search ranking. We measure the impact through offline evaluation and online A/B testing. Qualitative analysis is included to demonstrate the change in customer experience.

4.1 Query Seasonality

Compared to product MSCs, query MVCs computed from data are less noisy because *(i)* query text is in general shorter than product titles; and *(ii)* customers purchase a large variety of products but issue a smaller set of queries. In Table 3, we show MVCs for 2 pairs of queries computed from data. As one would expect, *sweater* is seasonally relevant to late fall and the entire winter season, and *christmas sweater* has an exceptionally large seasonal relevance to November and December. In the other pair, while *summer dress* is most seasonally relevant to spring and summer, *dress* has a flatter distribution across all 12 months.

To understand how query seasonal relevance relates to query volume and purchases, we segment all query-month pairs into *Low*, *Base* and *High*. The three segments correspond to a seasonal relevance of 0.00-0.075, 0.075-0.09 and 0.09-1.00, respectively. We treat 0.075-0.09 as the baseline since an equally distributed monthly concentration yields a seasonal relevance of 0.083 to any given month. The distribution of query-month pairs over the three segments then tells us the average percentage of time in a year a query is barely, moderately or highly seasonally relevant. It can be noticed from Table 4 that on average, each query is highly seasonally relevant around 31% time in a year, during which they drive close to 39% annual total query volume and as much as 42% purchases. That capability drops as queries move to months to which they are less seasonally relevant. The trend is consistent with our definition of seasonal relevance. We further measure each segment's efficiency in purchase conversion, by calculating the average number of purchases per search. Segment *High* outperforms the *Base* by 23%. Surprisingly, *Base* is also outperformed by *Low*, by 14%. One possible explanation is that customers are more likely to have a clear purchase intent when issuing queries with an either high or low seasonal relevance. For example, customers in general want to buy summer dresses during summer, but they might need summer dresses in winter as well for vacations on the beach. In both cases, they have a strong need of the type of products that they search for. On the other hand, the demand is less pressing and they may tend to browse more when issuing queries with a baseline level of seasonal relevance.

4.2 Product Seasonality

Similar to query seasonality analyses, we segment product-month pairs to study the relationship between seasonal relevance and product sales. As presented in Table 5, products on average are highly seasonally relevant to 35% time in a year, but drive around 48% of total purchases. Noticeably, off-season products (segment *Low*) still contribute a considerable share of purchases (24.5%), and

Table 3: Two pairs of query MVCs computed from query logs. High seasonal relevance (> 0.09) is highlighted in bold.

Query	m=1	m=2	m=3	m=4	m=5	m=6	m=7	m=8	m=9	m=10	m=11	m=12
sweater	0.081	0.045	0.026	0.020	0.018	0.018	0.019	0.027	0.064	**0.150**	**0.266**	**0.268**
christmas sweater	0.010	0.003	0.002	0.002	0.004	0.006	0.008	0.014	0.027	0.055	**0.344**	**0.525**
dress	0.067	0.074	0.088	**0.105**	**0.105**	**0.098**	**0.096**	0.086	0.083	0.084	0.056	0.059
summer dress	0.020	0.042	**0.110**	**0.166**	**0.198**	**0.208**	**0.137**	0.058	0.021	0.014	0.013	0.015

Table 4: Distribution of query volume and associated purchases over three segments of query seasonal relevance

	Low	Base	High
Pairs of query-month	32.4%	36.8%	30.7%
Query volume	24.2%	37.1%	38.8%
# Purchases	24.6%	33.1%	42.4%

Table 5: Distribution of product sales over three segments of product seasonal relevance

	Low	Base	High
Pairs of product-month	36.3%	28.9%	34.8%
# Purchases	24.5%	27.9%	47.6%
# Units	36.0%	19.7%	44.3%

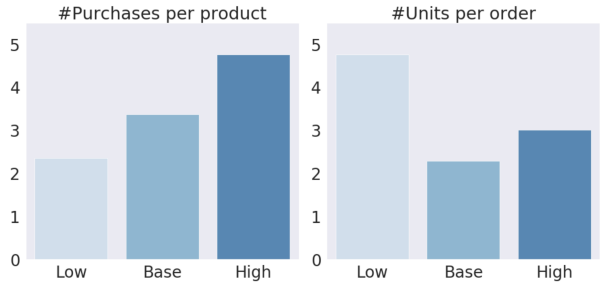


Figure 5: Efficiency in driving sales by segment of product seasonal relevance. The y-axis has been rescaled to omit absolute numbers

if we refer back to Table 4, queries also take up a good amount of query volume (24.2%) when they are minimally seasonally relevant. The story could be that people need out-of-season items from time to time. While it's hard to find such items during those times in brick-and-mortar stores, they could be available in an e-commerce store. Moving on to Figure 5, the first graph demonstrates the positive correlation between product seasonal relevance and the capability of driving purchases. Interestingly, units per order does not follow the same trend, with out-of-season products driving a disproportionally large number of units per order. We believe that sellers have discounts for products during off-season and people tend to purchase more units in one single order.

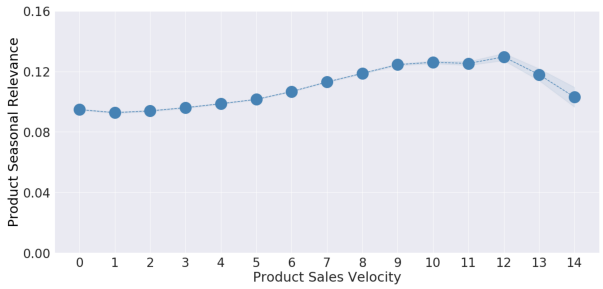


Figure 6: Average seasonal relevance of products by levels of sales velocity, with a 95% confidence interval marked by the blue band. The sales velocity numbers on the x-axis have been rescaled to omit absolute numbers.

4.3 Incorporate Product Seasonal Relevance into Search Ranking

Given the seasonal patterns in queries and product sales, as well as the implicit customer preference behind them, we propose that incorporating product seasonal relevance into ranking would improve search results and help customers find more relevant products. Meanwhile, one would expect that there is a natural interaction between the seasonal relevance of queries and that of products and thus both should be considered by the ranking model. While that can be achieved as our framework of modeling seasonal relevance applies to both queries and products, we decided to take a phased approach to understand the incremental gain, and to leave the adoption of query seasonal relevance as future work.

4.3.1 Seasonal Relevance versus Behavioral Features. Many learning-to-rank (LTR) models rely on behavioral features that track user interactions with items [13], such as clicks and purchases. Those features capture the rise and drop in demands, which one would expect to correlate with seasonality. One of such features is sales velocity, defined as the historical sales of a product aggregated using an exponential decay. We compared seasonal relevance of 700 K products with their sales velocity, over one year window, to understand whether the former would add any value to search ranking that is different from the latter. Their relationship is described by the non-monotonic curve in Figure 6. As shown by the curve, products with the largest sales velocity are not necessarily the most seasonally relevant and vice versa. The non-monotonic relationship is rooted in the difference between the definition of sales velocity and that of seasonal relevance. While the former is an absolute measure, the latter has a contrast and comparative nature which is required to quantify seasonal patterns. The analysis gives us the

evidence that seasonal relevance can provide a new dimension of information to search ranking, and we discuss how we combine the two, in subsequent sections.

4.3.2 Seasonal Relevance as a Ranking Feature. To incorporate seasonal relevance into search ranking, we built new ranking features based on product MSC model predictions, similar to the practice in [12, 22]. While there are other approaches that we considered, they all have limitations for our use case: *(i)* Use seasonal relevance as a filter. It applies well to recommendation systems as in [24]. However, users shopping on e-commerce stores do search for out-of-season products, so a hard filtering would yield poor search results; and *(ii)* Treat seasonal relevance as a prior and add a seasonal relevancy boost on top of the original relevance score as in [8], or to upweight seasonally relevant products in the loss function when training the ranking model. Either of those two methods could hurt the overall quality of search results without careful tuning. In our approach, we rather focused on feature engineering and built two distinct ranking features:

(1) LogSR - a scaled log transformation of seasonal relevance
(2) VelSR - a composite feature combining seasonal relevance with sales velocity, with the motivation to consider the products' inherent seasonal attributes, as well as the changing and sometimes hard-to-predict dynamics

Both features grow monotonically as seasonal relevance increases.

4.3.3 Rankers with Seasonal Relevance. We trained three LambdaMART [2] LTR rankers on 6 MM queries sampled from one year time window, using the same loss function and optimization procedures. Among them, the baseline ranker used neither of the features proposed in Section 4.3.2, while the other two used the LogSR feature and the VelSR feature, respectively. For simplicity, we will refer to the latter two as the LogSR ranker and the VelSR ranker respectively, and the SR rankers collectively. One confounding factor is that the VelSR feature combines seasonal relevance with sales velocity. In order to isolate its contribution as a composite feature from that of the sales velocity, we made sure that the latter was used as a stand-alone feature in all rankers. In the VelSR model, the VelSR feature ranked among the top 10 based on information gain, whereas the LogSR feature ranked after the 15th position in the LogSR model due to its being agnostic to absolute sales and thus having a weaker correlation with purchases. Figure 7 shows how relevance score changes along with the two SR features. Note that a relevance score is a value output by a ranker, and products with a high relevance score will be pushed to the top of search results. Both graphs in Figure 7 show an overall upward trend, except for the tick at 0 in the x-axes that represents a missing value. While the trend is monotonically increasing and fairly smooth in the VelSR model, it is more zig-zagged in the LogSR model, which can be explained by the low feature importance of the LogSR feature. Another observation is the decreasing rate of growth in the curves. When products have a seasonal relevance lower than 0.057 (corresponding to 800 in LogSR), the rankers demote them aggressively and generate a relevance score well below the average (marked by the blue line). Once the seasonal relevance reaches 0.10 (corresponding to 1400 in LogSR), the contribution of SR features saturates and the rankers let other features to factor in more.

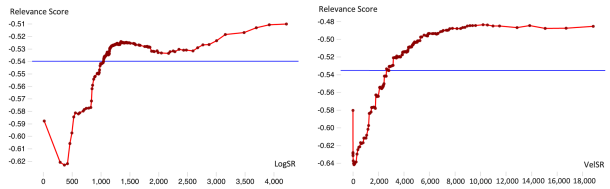


Figure 7: Changes in relevance score as LogSR (left) and VelSR (right) increase. The graphs are generated with the LogSR ranker and the VelSR ranker, respectively.

4.3.4 Offline Evaluation. We ran offline evaluation comparing the two SR rankers with the baseline ranker, on a test dataset of 2 MM queries sampled from one year time window. Performance was measured in terms of NDCG@10 on *(i)* the entire dataset, *(ii)* queries with at least one daily occurrence (head) and the rest (tail), and *(iii)* segments of queries based on their seasonal relevance. We report percentage differences relative to the baseline ranker in Table 6. LogSR had a performance comparable with that of the baseline on the entire dataset, but with an advantage in tail queries. VelSR, on the other hand, had a lower NDCG@10 than the baseline did in both head and tail queries. However, the gap is smaller on tail queries. Both the SR rankers performed better on tail queries than they did on the head, possibly due to the limited coverage of existing ranking features on tail queries. Further segmenting the query set based on their seasonal relevance, we did not see a clear pattern as the rankers were not trained to target any of the specific segments. While offline evaluation is a convenient way to measure model performance, it should be noted that offline evaluation in our case will suffer severely from counter-factual biases. For an unbiased evaluation, we ran an online A/B test.

4.3.5 Online Evaluation. The online A/B test was ran for 4 weeks. We studied the performance of rankers from three perspectives: *(i)* user behavior, *(ii)* quality of search results, and (iii) purchases of seasonally relevant products.

User behavior. User behavior is a mirror of customer experience, directly reflecting whether the rankers surfaced products that appealed to customers. We tracked clicks, add-to-carts and purchases to understand the impacts to the entire customer journey. Note that we collected only search-attributed user activities to prevent the signals from being diluted by downstream actions. As presented in Table 7, LogSR increased purchases in queries with a high seasonal relevance, but did not yield statistically significant changes to other customer engagement and purchase metrics. Whereas for VelSR, we saw a lift across the board in all three metrics of user behavior. In particular, VelSR drew 0.86% more clicks to queries with a high seasonal relevance, and 1.69% more add-to-carts and 2.07% more purchases to queries with a moderate seasonal relevance.

Quality of search results. We leveraged human judgement to understand the quality of search results presented by the rankers. The focus was on relevancy of search results to the query, rather than customers' affection of the products. Products shown under a query were labeled as *relevant* or *irrelevant* by human judges. We then measured the average percentage of relevant products, weighted

Table 6: Offline evaluation on test data. The dataset was split into head and tail, representing the set of queries with at least one daily occurrence and the rest. The queries were further segmented based on their seasonal relevance. Cells report the percentage gains in NDCG@10 of the LogSR ranker (above) and that of the VelSR ranker (below), *wrt* the baseline ranker. The statistical significance is denoted by * (p-value < 0.05).

Ranker	Overall	Head	Tail	Head			Tail		
				Low	Base	High	Low	Base	High
LogSR	+0.01%	-0.16%*	+0.05%*	-0.15%*	-0.05%	-0.15%*	+0.04%	+0.05%*	+0.06%*
VelSR	-0.10%*	-0.36%*	-0.03%*	-0.42%*	-0.25%*	-0.33%*	-0.02%	-0.01%	-0.07%*

Table 7: Online results of A/B test. Cells report the percentage gains in clicks, add-to-carts and purchases of the LogSR ranker (left) and that of the VelSR ranker (right), *wrt* the baseline ranker. Evaluation was ran on the overall query traffic, as well as segments of queries based on their seasonal relevance. The statistical significance is denoted by * (p-value < 0.05).

	LogSR				VelSR			
	Overall	Low	Base	High	Overall	Low	Base	High
Clicks	-0.39%	-1.05%	-0.17%	-0.42%	+0.62%*	+0.41%	+0.49%	+0.86%*
Add-to-carts	+0.04%	-1.47%	+0.49%	+0.12%	+1.22%*	+0.77%	+1.69%*	+0.71%
Purchases	+1.15%	+1.63%	+0.05%	+2.38%*	+2.20%*	+3.74%	+2.07%*	+1.59%

Table 8: Sales distribution over 3 segments of product seasonal relevance, during online A/B test. Cells report the change *wrt* the baseline.

	Low	Base	High
LogSR	-0.31%	+0.13%	+0.18%
VelSR	-0.26%	-0.06%	+0.32%

by their positions in search results. Both SR rankers substantially improved search relevance over the baseline, but the improvement brought by VelSR was 12% more pronounced than that from LogSR.

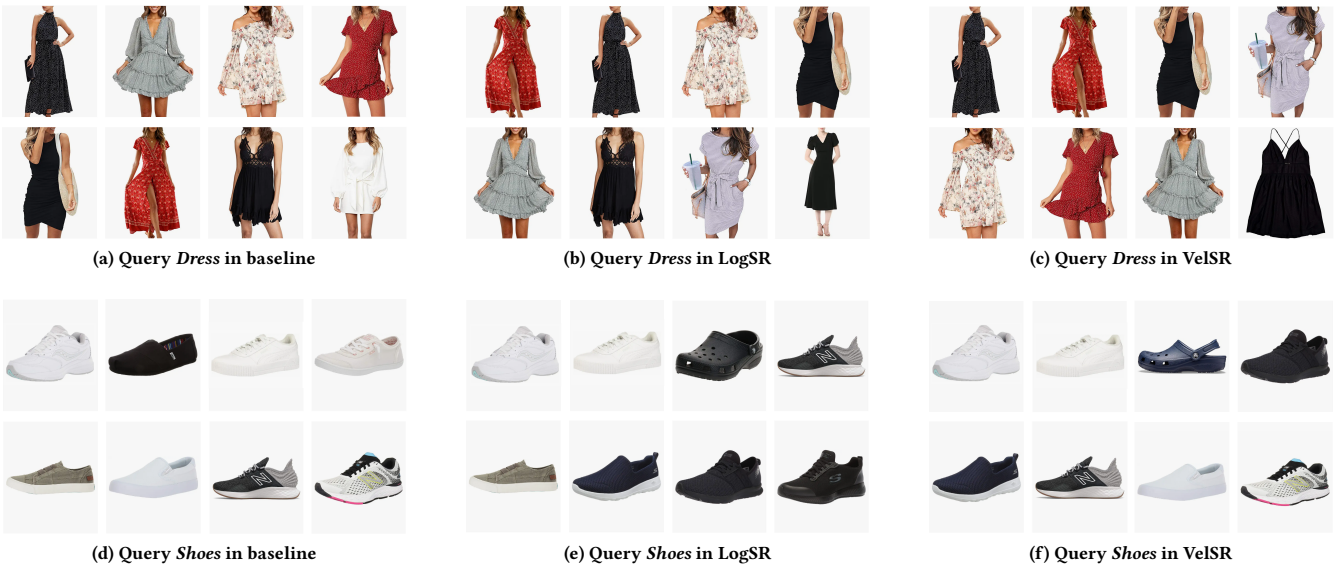
Purchases of seasonally relevant products. At last, we analyzed how SR rankers impacted the sales of seasonally relevant products. While both lifted the average seasonal relevance of purchased products by 0.20% and 0.24% respectively, only VelSR statistically increased the sales of highly seasonally relevant products, by 1.13%. It turned out that compared to the baseline, LogSR skewed the sales distribution from segment *Low* to *Base* and *High*, while VelSR shifted it from *Low* and *Base* to *High*, as demonstrated in Table 8.

Main findings. The experiment results suggested the overall superiority of VelSR over LogSR and its better performance in driving sales of highly seasonally relevant products. We believe there are two reasons behind. First, as mentioned previously, the VelSR feature ranked higher in the VelSR ranker than the LogSR feature ranked in the LogSR ranker, based on feature importance. Therefore, VelSR could more effectively change search results and surface highly seasonally relevant products. Second, VelSR leverages sales velocity, a quantity that is a proxy of many factors that affect purchase decisions, such as reviews. Sales velocity also by itself correlates with seasonal relevance to some degree, as demonstrated in Figure 6. We noticed the drastic differences in offline and online results, which validates our hypothesis that the seasonal relevance signal is offering a new dimension that has not been covered by existing ranking features and not captured in historical customer behavior entirely. Proven by online experiments, the proposed features help surface more seasonally relevant products which leads to higher customer engagement and purchases.

4.3.6 Qualitative Analysis. Finally, in Figure 8, we include the top 8 positions of search results rendered in May during the online experiment and illustrate how incorporating seasonal relevance into search ranking would qualitatively change customer experience. For the query *Dress*, a white knit long sleeved dress was ranked at the 8th position by the baseline ranker. Its seasonal relevance to May is as low as 0.03, so LogSR pushed it down to the 32nd position. VelSR also moved it down, but to the 12th position considering its popularity in the past. Another long sleeved dress in blue was ranked at the 2nd position by the baseline. It has floral print with a summer style and thus is more seasonally relevant to May compared to the previous one. However, it is still less seasonally relevant than the other dresses, so both SR rankers pushed it down but kept it within the top 8 positions. The story is similar for the off-shoulder dress in floral print, which was ranked at position 3 in the baseline results. For the query *Shoes*, both SR rankers surfaced a pair of clogs to position 3, while there were no clogs in the top 8 positions of baseline results. Clogs in general are popular during summer. They on average have a seasonal relevance of 0.10 to May. The two query examples demonstrate the effectiveness of using the seasonal relevance signal to promote the visibility of in-season products.

5 RELATED WORK

We discuss lines of work that are relevant to the study and application of temporal aspects in information retrieval (IR) and recommender systems.



(a) Query *Dress* in baseline (b) Query *Dress* in LogSR (c) Query *Dress* in VelSR

(d) Query *Shoes* in baseline (e) Query *Shoes* in LogSR (f) Query *Shoes* in VelSR

Figure 8: Top 8 positions in search results for the query *Dress* and the query *Shoes*, during online A/B test

Temporal profiling of queries and documents. A large body of work in IR is devoted to profile queries and documents from the perspective of time, including [7, 8, 14, 15, 23]. The research in [14, 23] quantify temporal dynamics of queries based on query frequency, while [7] builds temporal profiles of each document by learning how its content changes over time. All three work utilize time series analysis and focus on classifying the items of interest into categories such as seasonal and non-seasonal. In comparison, [8, 15] associate items of interest with units of time. Both work formulate temporal relevance as the probability that a document is relevant at point in time. Their probabilistic setting is in common with our approach. [8] estimates query-time relevance by counting the number of documents matching with the given query at different time points. [15] applies statistical text mining on documents and aggregates across time to derive document-time relevance. Compared to our method, [15] performs statistical modeling on unigrams and does not utilize semantics of natural languages.

Temporal search ranking. Temporal information is incorporated with search ranking in [5, 7, 8, 12, 22]. Both [8, 12] evaluate the temporal similarity between a query and a document. However, [12] uses the temporal similarity as a ranking feature for learning a time-aware ranking model, while [8] applies it as a boost on top of query-document topical similarity to ensure *(i)* documents are topically relevant to the query; and *(ii)* the query is temporally relevant to the publish date of documents. [7] rolls up document-level temporal information to query-level and uses the idea of divide-and-conquer to train separate rankers targeting queries with various temporal profiles. [5] focuses on recency in web search. It models query timeliness, which is defined as the degree to which a query requires freshness of search results. Then similar to [8], it includes document freshness as a dimension of relevance, and up-weighs or down-weighs it depending on the query timeliness. [22] targets spatial search. It derives temporal ranking features, for use in the LTR framework, which measure the relevance of a place to the time of search based on past timestamped check-ins at the given place. Its approach resembles our proposed method, but the derived features are memory-based instead of predictive.

Temporal recommender systems. Dynamic recommender systems have drawn much attention from researchers in recent years, which leverage temporal information for more accurate recommendations [4, 16, 17, 21, 24, 27, 28]. Temporal order of events is modeled by [16, 17] to account for users' preference transition and to provide personalized recommendation. The former leverages Gaussian process, while the latter adopts recurrent neural networks. [16, 27] incorporate periodic patterns, as in this paper, but with finer granularities such as time of the day and day of the week. Similar to the language modeling task we propose, [16] makes use of semantic information to model the relationship between time and category of items to identify categories that would most likely appeal to users at a given time frame. [24, 28] consider seasonal changes. In particular, [24] identifies seasonal products in e-commerce stores by studying each product's monthly ordered units. It approaches seasonal relevance in a way akin to ours, with two differences: *(i)* its method relies on historical sales and does not apply to new products, while our predictive approach handles cold-start scenarios; and *(ii)* it uses seasonal relevance in a binary fashion with a hard threshold to find low-seasons of products, whereas we leverage the full scale of seasonal relevance scores and transform them into ranking features.

6 CONCLUSIONS AND FUTURE WORK

In this study, we formally introduce the concept of seasonal relevance in standard learning-to-rank setup for e-commerce search. We also present quantitative analyses of how much e-commerce search traffic is actually affected by seasonality through empirical

study of queries of a major e-commerce store and outline the scope and impact. Proposed features based on neural models provide a principled way to model seasonal relevance which helps to generalize and reduce data specific noise. Comprehensive offline and online experiments highlight the value in handling seasonality in e-commerce search. The A/B test on 784 MM searches strongly suggests that the proposed methods present higher seasonally relevant products which result in statistically higher purchases and better customer experience.

We propose three directions for future research to improve upon the work presented in this paper: *(i) Enriched product information.* While product titles have 100% coverage, they can also be noisy. Users reference other types of product information to make purchase decisions, such as product images. Those could be included in the neural network model to help predict seasonal relevance. *(ii) Location-aware seasonal relevance.* Customers' perception of seasonal relevance depends on factors such as climate and culture. They are manifested in the regional variation of seasonality. Thus, location information can be built into seasonal relevance modeling. *(iii) Ranking with query seasonal relevance.* As mentioned previously, there could be a natural interaction between seasonal relevance of queries and that of products. Therefore, the former can be additionally incorporated into search ranking to further improve e-commerce search.

REFERENCES

[1] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics* 5 (2017), 135–146.

[2] Christopher JC Burges. 2010. From ranknet to lambdarank to lambdamart: An overview. *Learning* 11, 23-581 (2010), 81.

[3] Ricardo Campos, Gaël Dias, Alípio M Jorge, and Adam Jatowt. 2014. Survey of temporal information retrieval and related applications. *ACM Computing Surveys (CSUR)* 47, 2 (2014), 1–41.

[4] Chen Chen, Hongzhi Yin, Junjie Yao, and Bin Cui. 2013. Terec: A temporal recommender system over tweet stream. *Proceedings of the VLDB Endowment* 6, 12 (2013), 1254–1257.

[5] Shiwen Cheng, Anastasios Arvanitis, and Vagelis Hristidis. 2013. How fresh do you want your search results?. In *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*. 1271–1280.

[6] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078* (2014).

[7] Na Dai, Milad Shokouhi, and Brian D Davison. 2011. Learning to rank for freshness and relevance. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*. 95–104.

[8] Wisam Dakka, Luis Gravano, and Panagiotis Ipeirotis. 2010. Answering general time-sensitive queries. *IEEE Transactions on Knowledge and Data Engineering* 24, 2 (2010), 220–235.

[9] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).

[10] Sebastian Hofstätter, Navid Rekabsaz, Carsten Eickhoff, and Allan Hanbury. 2019. On the Effect of Low-Frequency Terms on Neural-IR Models. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval* (Paris, France) *(SIGIR'19)*. Association for Computing Machinery, New York, NY, USA, 1137–1140. https://doi.org/10.1145/3331184.3331344

[11] Abhay Jha. 2017. Disjoint-Support Factors and Seasonality Estimation in E-Commerce. In *Machine Learning and Knowledge Discovery in Databases*. Springer International Publishing, Cham, 77–88.

[12] Nattiya Kanhabua and Kjetil Nørvåg. 2012. Learning to rank search results for time-sensitive queries. In *Proceedings of the 21st ACM international conference on Information and knowledge management*. 2463–2466.

[13] Shubhra Kanti Karmaker Santu, Parikshit Sondhi, and ChengXiang Zhai. 2017. On application of learning to rank for e-commerce search. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 475–484.

[14] Anagha Kulkarni, Jaime Teevan, Krysta M Svore, and Susan T Dumais. 2011. Understanding temporal query dynamics. In *Proceedings of the fourth ACM international conference on Web search and data mining*. 167–176.

[15] Abhimanu Kumar, Matthew Lease, and Jason Baldridge. 2011. Supervised language modeling for temporal resolution of texts. In *Proceedings of the 20th ACM international conference on Information and knowledge management*. 2069–2072.

[16] Wenchao Li, Xin Liu, Chenggang Yan, Guiguang Ding, Yaoqi Sun, and Jiyong Zhang. 2020. STS: Spatial–Temporal–Semantic Personalized Location Recommendation. *ISPRS International Journal of Geo-Information* 9, 9 (2020), 538.

[17] Yifei Ma, Balakrishnan Narayanaswamy, Haibin Lin, and Hao Ding. 2020. Temporal-Contextual Recommendation in Real-Time. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2291–2299.

[18] Donald Metzler, Rosie Jones, Fuchun Peng, and Ruiqiang Zhang. 2009. Improving Search Relevance for Implicitly Temporal Queries. In *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval* (Boston, MA, USA) *(SIGIR '09)*. Association for Computing Machinery, New York, NY, USA, 700–701. https://doi.org/10.1145/1571941.1572085

[19] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781* (2013).

[20] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality. *arXiv preprint arXiv:1310.4546* (2013).

[21] Idris Rabiu, Naomie Salim, Aminu Da'u, and Akram Osman. 2020. Recommender system based on temporal models: a systematic review. *Applied Sciences* 10, 7 (2020), 2204.

[22] Blake Shaw, Jon Shea, Siddhartha Sinha, and Andrew Hogue. 2013. Learning to rank for spatiotemporal search. In *Proceedings of the sixth ACM international conference on Web search and data mining*. 717–726.

[23] Milad Shokouhi. 2011. Detecting seasonal queries by time-series analysis. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*. 1171–1172.

[24] Henrik Stormer. 2007. Improving e-commerce recommender systems by the identification of seasonal products. In *Twenty second Conference on Artificial Intelligence*. 92–99.

[25] Manos Tsagkias, Tracy Holloway King, Surya Kallumadi, Vanessa Murdock, and Maarten de Rijke. 2021. Challenges and Research Opportunities in ECommerce Search and Recommendations. *SIGIR Forum* 54, 1, Article 2 (Feb. 2021), 23 pages. https://doi.org/10.1145/3451964.3451966

[26] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *arXiv preprint arXiv:1706.03762* (2017).

[27] Quan Yuan, Gao Cong, Zongyang Ma, Aixin Sun, and Nadia Magnenat Thalmann. 2013. Time-aware point-of-interest recommendation. In *Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval*. 363–372.

[28] Farhad Zafari, Irene Moser, and Tim Baarslag. 2019. Modelling and analysis of temporal preference drifts using a component-based factorised latent approach. *Expert Systems with Applications* 116 (2019), 186–208.