Detecting Seasonal Queries by Time-Series Analysis

Milad Shokouhi
Microsoft Research
Cambridge, United Kingdom
milads@microsoft.com

ABSTRACT

Seasonal events such as Halloween and Christmas repeat every year and initiate several temporal information needs. The impact of such events on users is often reflected in search logs in form of seasonal *spikes* in the frequency of related queries (e.g. "halloween costumes", "where is santa"). Many seasonal queries such as "sigir conference" mainly target fresh pages (e.g. `sigir2011.org`) that have less usage data such as clicks and anchor-text compared to older alternatives (e.g. `sigir2009.org`). Thus, it is important for search engines to correctly identify seasonal queries and make sure that their results are temporally reordered if necessary.

In this poster, we focus on detecting seasonal queries using *time-series* analysis. We demonstrate that the seasonality of a query can be determined with high accuracy according to its historical frequency distribution.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval

General Terms

Algorithms, Performance

Keywords

Temporal Queries, Seasonal Query Classification

1. INTRODUCTION

Cyclic queries related to seasonal events have been estimated to account for more than 7% of search traffic [5]. The best result pages for many of these queries change every year, and in some cases such as "us open" the query itself becomes temporally ambiguous [4]. Therefore, it is important for search engines to detect – and predict – seasonal queries and respond to them with temporally relevant results.

Previous work has been mostly focused on identifying seasonal queries based on their reformulations, and the frequent occurrence of years in their words. For instance, Metzler et al. [5] define a query (e.g. "halloween") as *implicitly year qualified* if it frequently appears in the logs followed by years (e.g. "halloween 2009", "halloween 2010"). The authors rely on the number of unique years and their frequencies to classify query seasonality. Other related work includes modeling the temporal profile of queries based on the time-stamps of the documents they return [3]. Dong et al. [6] used a combination of reformulation and post-retrieval features to train a classifier for identifying seasonal queries.

We propose leveraging time-series decomposition techniques for measuring the seasonality of queries. In contrast to previous work, we do not rely on any information about previous query reformulations or document time-stamps.

SIGIR'11, July 24–28, 2011, Beijing, China.
ACM 978-1-4503-0757-4/11/07.

2. TIME-SERIES AND QUERY HISTORY

A time-series [6] is referred to a group of data points at successive time spans with uniform intervals. The data points can represent any quantifiable such as sale records or temperature, and the interval can take any unit of time such as week or year. In the context of web search, one can generate a time-series for each query according to its past frequency at uniform intervals. That is, each data point in the time-series represents the query frequency at a time unit (e.g. day, or month). Having a time-series we can then apply different well-studied decomposition techniques such as STL [1] and HoltWinters [2] to analyze trends and seasonality. Figure 1 depicts how query frequency history at monthly intervals can be treated as time-series and decomposed by Holt-Winters additive exponential smoothing [2] into three main components; *level* ($\mathcal{L}$), *trend* ($\mathcal{T}$) and *season* ($\mathcal{S}$):

$$\mathcal{L}_t = \alpha(\hat{X}_t - \mathcal{S}_{t-s}) + (1-\alpha)(\mathcal{L}_{t-1} + \mathcal{T}_{t-1})$$
$$\mathcal{T}_t = \beta(\mathcal{L}_t - \mathcal{L}_{t-1}) + (1-\beta)\mathcal{T}_{t-1}$$
$$\mathcal{S}_t = \gamma(\hat{X}_t - \mathcal{L}_t) + (1-\gamma)\mathcal{S}_{t-s}$$

here α, β and γ are all between zero and one and can be set during the fitting process using standard techniques such as root mean square error. Parameter t denotes the current time-span, and s specifies the length of the seasonal cycle (12 in our experiments). $\hat{X}_t$ represents the value of the data point at time t (here, the monthly frequency of the query).

For seasonal queries, the *season* component ($\mathcal{S}$) is the most significant factor and noticeably resembles the distribution of the raw data. For instance in Figure 1, the *season* and *xhat* have very different distributions for query "britney spears", while they look remarkably similar for "us open".

Our seasonal query classifier is inspired by this observation; for a given query, we first convert its historical frequency into time-series with monthly splits. We then decompose the time-series by applying Holt-Winters additive

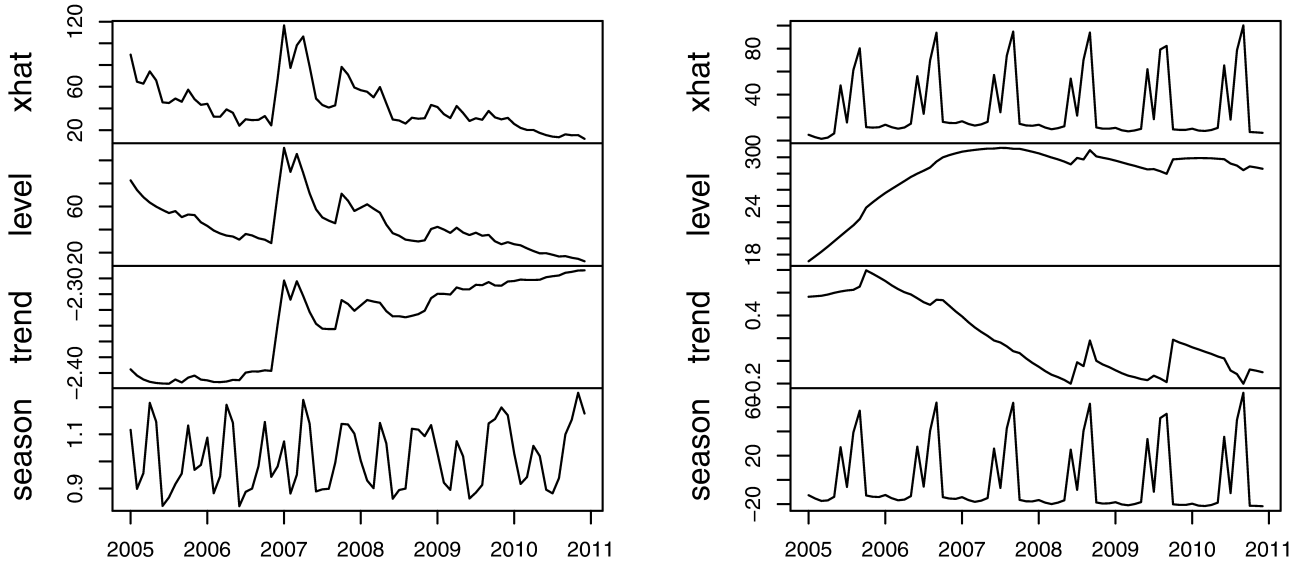

Figure 1: The Holt-Winters decomposition of time-series generated from monthly frequencies of "britney spears" (left), and "us open" (right). The xhat ($\hat{X}$) represents the raw monthly data, and is followed by the decomposed components in each plot. The raw data was collected from Google insight for search.

Table 1: The precision-recall changes in seasonal classification for different values of ω.

ω	.90	.80	.70	.60	.50	.40	.30	.20	.10
Precision	.78	.60	.47	.42	.38	.32	.32	.30	.30
Recall	.49	.68	.77	.81	.82	.84	.85	.89	.89

smoothing [2]. If the decomposed *season* component $\mathcal{S}$ and raw data $\hat{X}$ have similar distributions, we classify the query as seasonal. We use the Cosine similarity function to measure the similarity of the vectors for $\mathcal{S}$ and $\hat{X}$. That is,

$$\omega = \frac{\mathcal{S} \cdot \hat{X}}{\|\mathcal{S}\|\|\hat{X}\|}$$

The query is classified as seasonal if the similarity value ω is greater than a certain threshold. We require at least two spikes at seasonal peaks to reduce the chance of misclassification in cases where distributions are flat or similar due to reasons other than seasonality. Note that our approach is not restricted to any particular decomposition method and can work with other related techniques such as STL [1]. Similarly, we could use KL-Divergence or other appropriate functions for comparing the distributions of $\mathcal{S}$ and $\hat{X}$.

3. EXPERIMENTS

We evaluated our experiments on set of 259 queries annotated manually by professional editors as seasonal or normal (with respectively 74 and 185 queries in each group). We used the search logs of a commercial search engine to generate a time-series for each query according to its monthly frequency between 2006 and 2010 inclusive.

The results summarized in Table 1 suggest that our simple technique can classify half of seasonal queries with 78% precision. Our misclassification cases mainly included queries such as "miss world" that are cyclic but repeat at slightly different time of the year, and "powerball lottery results" that are related to repetitive events but at inconsistent intervals.

4. CONCLUSIONS

We proposed using time-series decomposition techniques for identifying seasonal queries. Our experiments showed that seasonal queries can be classified by (1) transforming the query frequency history into time-series and (2) comparing the decomposed seasonal factor of the time-series and the raw data distributions. Future work includes comparing the effectiveness of various decomposition and similarity functions for seasonal query classification.

5. REFERENCES

[1] R. Cleveland, W. Cleveland, J. Mcrae, and I. Terpenning. STL: A seasonal-trend decomposition procedure based on loess. *Journal of Official Statistics*, 6(1):3–73, 1990.

[2] P. Goodwin. The holt-winters approach to exponential smoothing: 50 years old and going strong. *Foresight: The International Journal of Applied Forecasting*, (19):30–33, 2010.

[3] R. Jones and F. Diaz. Temporal profiles of queries. *ACM Transactions Information Systems*, 25, July 2007.

[4] A. Kulkarni, J. Teevan, K. M. Svore, and S. T. Dumais. Understanding temporal query dynamics. In *Proceedings of WSDM conference*, pages 167–176, Hong Kong, China, 2011.

[5] D. Metzler, R. Jones, F. Peng, and R. Zhang. Improving search relevance for implicitly temporal queries. In *Proceedings of SIGIR conference*, pages 700–701, Boston, MA, 2009.

[6] R. Zhang, Y. Konda, A. Dong, P. Kolari, Y. Chang, and Z. Zheng. Learning recurrent event queries for web search. In *Proceeding of EMNLP conference*, pages 1129–1139, MIT, MA, 2010.