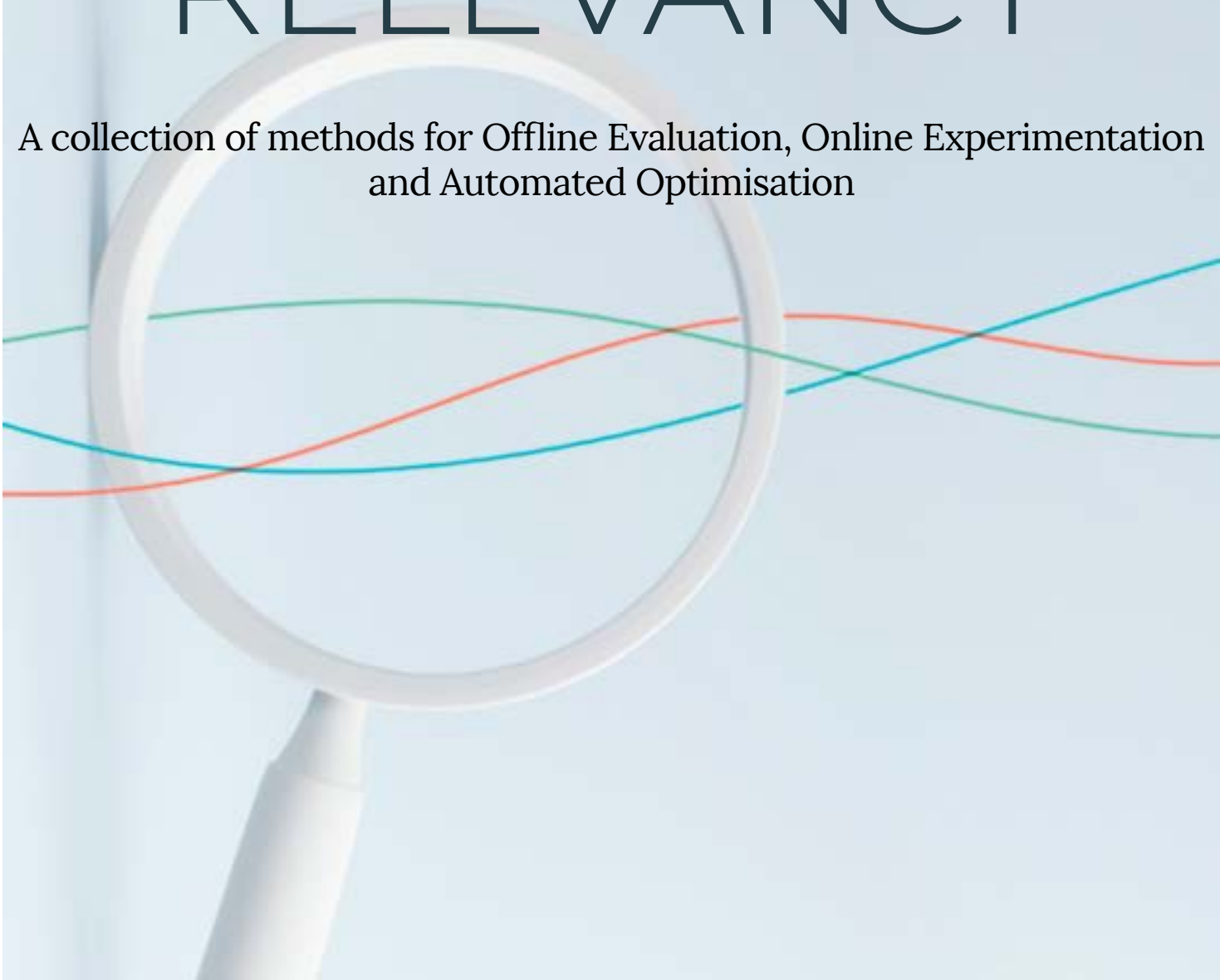


# MEASURING SEARCH RELEVANCY



A collection of methods for Offline Evaluation, Online Experimentation  
and Automated Optimisation

empathy.co

## CONTENTS

- / Motivation
- / Judging “relevance” of products per query
- / Metrics
- / Parameter Optimization
- / Experimentation Strategy

# MOTIVATION

The interpretation of search results is subjective, aiming to establish:

- a) A systematic approach to search improvement.
- b) A common set of criteria to measure the effect of changes that makes a good result better.

A formal approach is a step towards a common language in the formulation of search results, a change of focus from single examples to search performance as a whole.

At the same time, we propose automated methods that are applicable down to the query level.

This approach increases the speed of search evolution by enabling the evaluation of search performance offline.

We strive towards fearless, continuous online experimentation, which will enable us to react rapidly and understand customer needs more deeply.

## THE DOCUMENT PRESENTS:

- A judgement on the “relevance” of a product for a given query (judgements; offline).
- An evaluation of results in an aggregated/automated fashion, measuring query performance (offline).
- How fine-tuning parameters can determine the behaviour of the ranking algorithm (offline).
- Continuous experimentation in an online setting.

# CONTENTS

01  
MOTIVATION

02  
JUDGING PRODUCT  
“RELEVANCE” PER QUERY

03  
METRICS

04  
PARAMETER  
OPTIMIZATION

05  
EXPERIMENTATION  
STRATEGY

# JUDGING PRODUCT "RELEVANCE" PER QUERY

Event data gives us:

- Type of user-product interaction (*click*, *add2basket*)
- Result position of interaction
- Related query
- ProductId

For user interactions, the goal is to extract an estimate of a product's relevance for any particular query. The performance of a product might vary widely per query, due to the difference in the intentions expressed by the query.



## Factors to consider

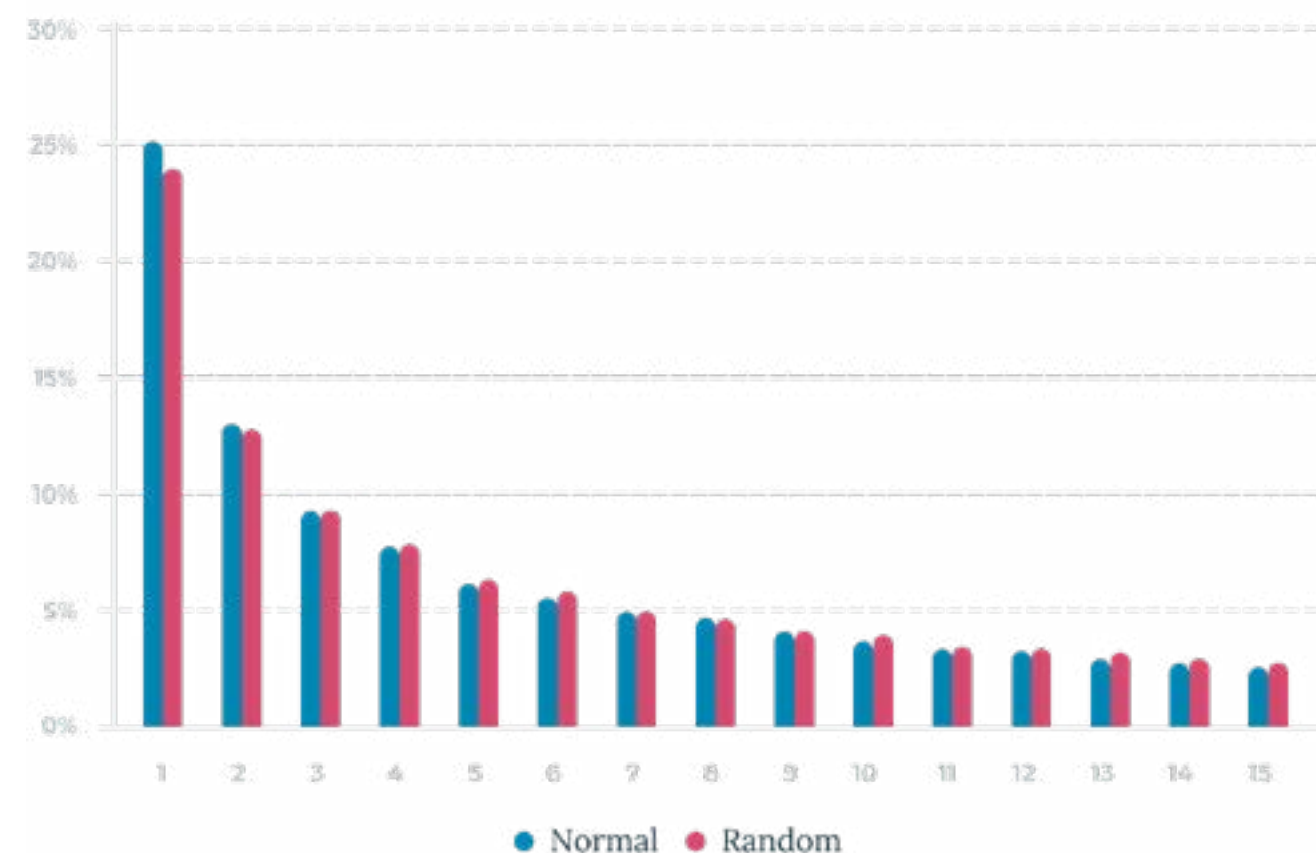
### How important is an event type as an indicator of relevance?

Very important. Certainly, an *add2basket* is a stronger signal than a *click*, but a *click* does still signal an interest. Furthermore, fewer corresponding *add2cart* events may indicate pricing issues or other factors.

### Does the interaction position play a role?

Generally speaking, click probability sharply decreases with click positioning. But how much of this is actually due to differences in score? The data, as presented in Mices 2019 by Roman Grebennikov (based on mid-to-medium-sized stores) upon randomly scrambling search results, suggests the differences might not be so big.

click ranks



[https://mices.co/mices2019/slides/grebennikov\\_search-real-time.pdf](https://mices.co/mices2019/slides/grebennikov_search-real-time.pdf)



Data suggests people still click top-sorted positions more often, either due to historical experiences of good sorting (due to Google, etc.) or due to a lack of interest in searching the later positions.

So are clicks at position 1 and clicks at position 10 worth the same?

Unlikely. As has been widely proven, users avoid long scrolls and changing pages, interpreting these actions as high in effort and, therefore, unpleasant.

Therefore, if products with a high probability of being interacted with present less interaction than those at later positions, those with clicks at later positions must be assigned greater value, as they are less likely to happen due to a required extra effort.

We thus arrive at a rough formula for a single event contribution.

$$\text{single-event-contribution} = \text{eventTypeWeight} * \text{positionWeight}$$

If we aggregate this over all query-product pairs and normalise it, we arrive at aggregated judgements of products for a given query, where higher values indicate a better fit.

RESULT

The score per query-product pair, reflects a product's suitability for a given query.

## What is this good for?

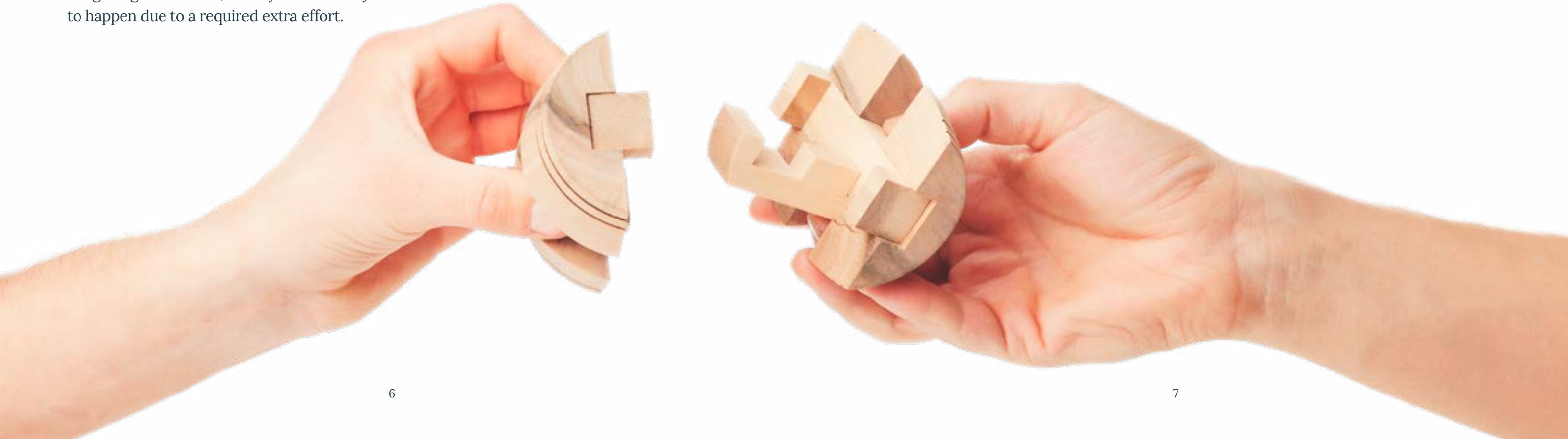
Query-product pair judgements are the basis for aggregated scoring of search result quality and descriptions of distributions (where best-fitting products are placed). At the same time, by visualising score distributions the effect of changes on

product relevance distribution is made clearly visible. There are many information retrieval metrics for generating aggregated scores. A sample is shown in the Metrics section below, together with an explanation of what they provide.

## Is this the only way?

No, a judgement list is just that: a judgement. In order to make these judgements a reflection of what is to be optimised, a set of criteria need to be agreed upon. In this case, a vital aspect to emphasise is the utilisation of users' feedback.

If required, judgements may also incorporate other goals, such as revenue. That being said, bringing in desired business KPIs might actually decrease the search experience for the users if the modified result does not reflect the user's needs.



# 03

## METRICS

There are many metrics in the IR (Information Retrieval) world that attempt to estimate the quality of search results. These are not always tailored towards the e-commerce shopping experience and often assume there is an information need that can be satisfied (due to the origin of document retrieval).

Yet, these metrics are widely used to estimate the quality of e-commerce search results and optimisations based on normalised discounted cumulative gain (NDCG) are often debated. Some examples of these different assumptions can be seen below.

With so many types of metrics possible and appealing, including custom ones, it is easy to overdo it and make the optimisation problem harder.

After all, these are subjective agreed measures on what a 'good search' looks like, and what becomes clear is that using just a handful of metrics is the most reasonable approach.

## Families and Assumptions

While there are many types of metrics, they can be classified into **Position** and **Cascade** models.

### POSITION MODELS

Relevancy depends on position, not on other documents. To arrive at a more detailed picture of the quality of results, we assume an evaluation of the first K positions of the result for different levels of K (e.g K = 5, 10, 25 top results).

#### Binary Relevance

Apply a threshold to the judgement, if it is above assume relevant, otherwise it is irrelevant.

$$\text{precision} = \frac{|\{\text{relevant documents}\} \cap \{\text{retrived documents}\}|}{|\{\text{retrived documents}\}|}$$

$$\text{recall} = \frac{|\{\text{relevant documents}\} \cap \{\text{retrived documents}\}|}{|\{\text{relevant documents}\}|}$$

### Graded Relevance

#### DCG

$$s_0 + \sum_i \frac{s_i}{\log_2(i+2)}, \forall i \geq 1.$$

The more higher scores in front, the better.

#### NDCG

$$\text{normalized DCG} = \frac{DCG}{\text{ideal DCG}}$$

idealDCG = DCG on sequence after descending sort by score.

CASCADE MODELS

Relevancy depends on position and previous results.

ERR

(expected reciprocal rank;

[https://www.researchgate.net/publication/220269787\\_Expected\\_reciprocal\\_rank\\_for\\_graded\\_relevance](https://www.researchgate.net/publication/220269787_Expected_reciprocal_rank_for_graded_relevance))

While DCG is additive under the independence assumption (document contribution is only dependent on its position, not on previous results), ERR also takes context into account.

The evaluation of a product at position  $i$  depends on products at positions 1 to  $i - 1$

- Documents shown below *very relevant documents* are discarded.
- Expected reciprocal time users will take to find a relevant document is considered.
- Likelihood of users examining the document at rank  $i$  depends on user satisfaction with previously observed documents.

Mapping function  $R(g)$ , where  $g$  is the judgement for the respective element, mapping grade to probability.

$$ERR = \sum_{r=1}^n \frac{1}{r} \prod_{i=1}^{r-1} (1 - R_i) R_r.$$

Algorithm 2 Algorithm to compute the ERR metric (5) in linear time.

**Require:** Relevance grades  $g_i, 1 \leq i \leq n$ , and mapping function  $R$  such as the one defined in (4).  
 $p \leftarrow 1, ERR \leftarrow 0.$   
**for**  $r = 1$  to  $n$  **do**  
   $R \leftarrow R(g_r)$   
   $ERR \leftarrow ERR + p R/r$   
   $p \leftarrow p (1 - R)$   
**end for return**  $ERR$

As seen in the ERR algorithm, the contribution of the  $r$ -th document depends on the probability of the results that came before it being irrelevant where the probability it is relevant can be inferred from the judgements).

Statistical significance tests can be applied to judge significance of metrics differences.

In general, metrics reflecting the overall “goodness” of the search result provide a base for fast iterations, resulting in higher iteration speed and confidence.

Metrics: Iterations

More information exists regarding queries, users, interactions and results presentation. A query with 5 hits tends to have lower optimisation potential than one with 100 hits. A brand query, for instance, is not as broad as a food query. Some examples of possible interactions:

CLASSIFICATION OF QUERY TYPES / INTEND

Broad vs narrow queries:

- E.g. Classified by interaction % on field values.

Type-dependent ranking configurations.

WHICH PRODUCTS THAT WEREN'T INTERACTED WITH CAME BEFORE THE ONE INTERACTED WITH

Extension of event information. Provides more info than only actual positive click-data.  
More resource intensive.

CONDENSING LONG-TAIL INTERACTIONS BY GROUPING

Session information can be used to group queries and increase the data available.

EVALUATION PER USER-TYPE

Needs distinct judgments:

- E.g. Classification as inspiration-seeking or healthy-food-enthusiast likely changes behavior.
- Leads to type-influenced ranking.

TARGETED RESULT MODIFICATIONS / EXPERIMENTS

Products that are new or buried in the result list are likely to have less interactions than top-sorted products. A scheme to overlook this existing sorting deviation may be beneficial in those cases such as boosting new products higher up initially.

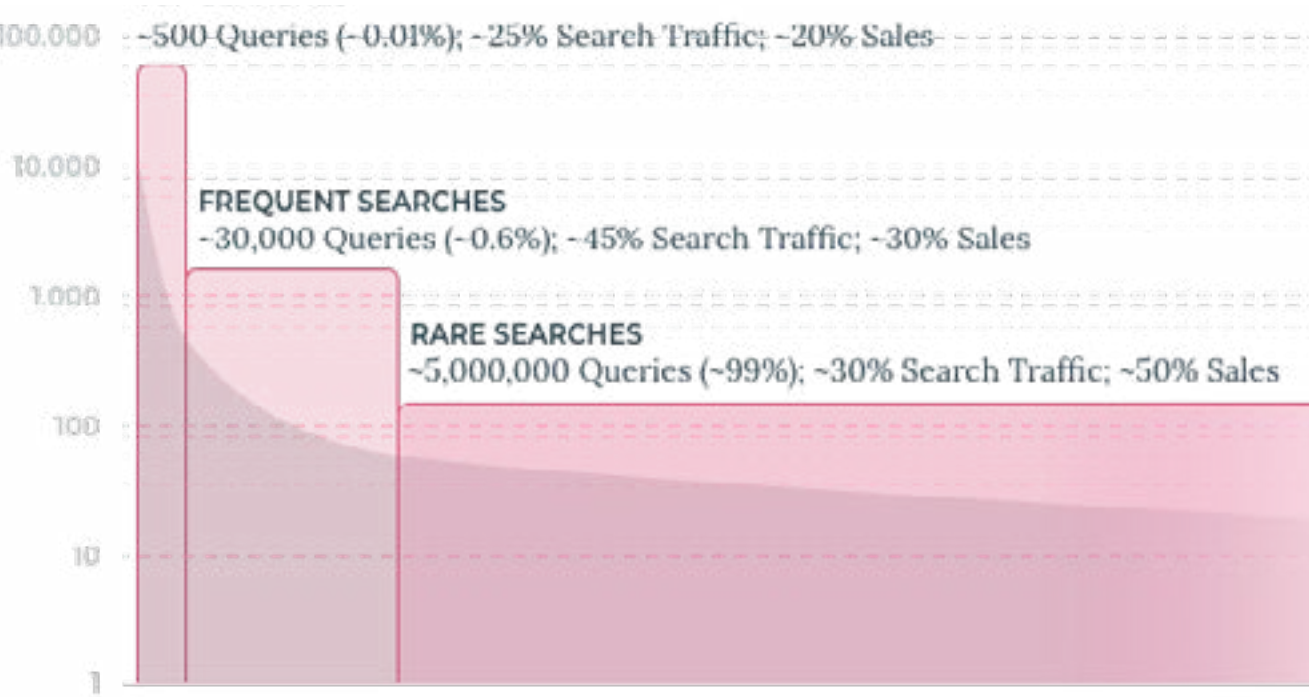


MANY MORE

Regarding the long-tail, consider the query distribution chart below, it was published by one of Germany's biggest online shops. For a relatively low number of queries, there is a lot of volume and impact. You can see that the second half of overall revenue was made in the long tail, which means many queries each with small amounts of data.

Due to the mentioned constraints, optimisation may focus only on the short tail, but, while optimising top searches is a good starting point, it may be more beneficial long-term to optimise the long tail.

Top searches



Source:  
[https://mices.co/mices2019/slides/wagenmann-kuersten\\_offline-evaluation-in-ecommerce-search-applications.pdf](https://mices.co/mices2019/slides/wagenmann-kuersten_offline-evaluation-in-ecommerce-search-applications.pdf)

Descriptive Stats

Score distributions

Looking at the distribution of scores provides more information, and additional descriptions of search result compositions are given by:

- Min, max.
- Mean, variance of scores.
- Sum of scores / Sum of scores for optimally sorted sequence (i.e. the ideal seq sorted over all, then first N is picked).

Descriptive Stats

Result Composition

When measured as counts over first N results (e.g N = 5, 10, 25), measures of result composition are:

- Distinct brand count.
- Distinct sub-commodity count.
- Number of personalised results.
- Avg. top-position personalised result.
- Number of contextualised results.
- Avg. top-position contextualised result.
- Count results utilising a synonym.
- Popularity.

Up to here, most of the above can be done without changes to the actual search mechanics (leaving out the user/query-type based ranking).

# PARAMETER OPTIMIZATION

With our criteria now defined using the sequence evaluation metrics, some initial brute force can be good to obtain the required set of base scores in order to optimise the respective measure.

As for relative boosting, we can expect relative values to have more importance than absolute weight values. For calculated metric values, an average score over the query sample would be most suitable.

Where queries correspond to distinct types of intents and customer behaviour, we have found that clustering these queries into distinct types and calculating separate scores can lead to well defined optimal values.

One way of defining “context” is by checking the set of products that define it, either explicitly or via an example query that can generate this representative result set and the actual results for a given query.



Schematical representation of an offline evaluation scheme.



The Empathy search platform combines a flexible query-matching algorithm with agility controls to manually refine the scoring mechanisms. This allows to dynamically adapt search to user behaviour and deliver personalised results, assigning relative weights to distinct components, including further personalisation and complementary sciences.

On top of feature acting on search requests, other features include search suggestions, dynamic related queries sets (related tags) and predictive suggestions of next search terms (next queries). The **Explain Tool** allow brands to clearly and visually expose the search result composition due to the additive nature of these applied schemes.

To simplify the testing process of settings and find the optimal configuration, the parameters defining the functioning of the search system are exposed in the API, exposing parameters to find optimal balance. Additionally, taking optimisation offline based on past user behaviour, we gain an automated way to determine the best settings for each search system, making manual adjustments purely optional.



# EXPERIMENTATION STRATEGY

## Be (and afford to be) fearless

For true experimentation, we aim to alleviate as much risk as possible. This usually involves highly anticipated A/B Tests, which place a high toll on development speed if done correctly, resulting in several weeks passing without significant changes accomplished.

Here, we define two key ways that enable us to throw in any model that comes to mind, at any time, without being at risk of burning down the house.

# Continuous multi-competitor model: Interleaving and Multi-armed bandits

## I) INTERLEAVING

*Interleaving is one way to yield some info about every model in every query. This is opposed to N1 %, N2 %, N3 % proportion split, where only NX % of queries yield any info about the quality of the variant applied to the query. Here, multiple models are applied per result, and each model is assigned a particular probability of getting a slot assigned in the search result. Each slot is, therefore, a random, interleaved draw based on the assigned probabilities.*

“...It reliably identifies the best algorithms with considerably smaller sample size compared to traditional A/B testing.”

“We find that interleaving is very sensitive: it requires >100× fewer users than our most sensitive A/B metric to achieve 95% power.”

In the long run, a rapid testing approach enables us to benefit from many small, quick changes, rather than waiting long periods for the few larger ones. The more optimised a system becomes, the higher the sample sizes needed and the experiment durations due to smaller effect sizes.

Read more on how Netflix utilises interleaving here:

<https://netflixtechblog.com/interleaving-in-online-experiments-at-netflix-a04ee392ec55>

## II) MULTI-ARMED BANDIT

*“In probability theory, the multi-armed bandit problem (sometimes called the K- or N-armed bandit problem) is a problem in which a fixed limited set of resources must be allocated between competing (alternative) choices in a way that maximises their expected gain, when each choice’s properties are only partially known at the time of allocation, and may become better understood as time passes or by allocating resources to the choice”*

Wikipedia entry on “multi-armed bandit”  
[https://en.wikipedia.org/wiki/Multi-armed\\_bandit](https://en.wikipedia.org/wiki/Multi-armed_bandit)

This approach reduces risk of bad models, probabilities applied to the interleaving scheme are adjusted based on continuously generated evidence (events). This combines more effective data generation with effective controlling of relative probabilities of models.

## Experimentation Conclusion

The combined approach allows quick testing and ranking models online iteration, especially if compared to A/B testing. Online is per se superior to offline testing, as offline testing only tries to emulate or approximate the real behaviour. The risk of experimentation is reduced when a scheme is supplied to either reduce or increase

the probabilities of models dynamically or to alter the criteria to turn models off. A combined approach requires a central state of search, that focuses attention on the computability or serving efficiency of various models. It must also consider the potential for the way relevancy adjustments are applied to a result set.





# A central state of search

Improving search is a continuous, dynamic process that can involve numerous methodologies. It is, therefore, essential to continually contextualise all experiments through a full definition of the current state of search:

- All models applied.
- All parameters set.
- All applied rules.
- All ways to modify the above, with controlled registration of adjustments.
- An environmental set of characteristics in which the current state operates: time of year, marketing events, etc.

Results that are modified by external logic after leaving the search system can not be utilized for evaluation of the search system, thus corresponding events ideally need to be filtered out. If utilisation of these is necessary, a clear central state for event filtering should be utilized, making all variants of modifications fully transparent. These additional external factors -if applied- effectively reduce the amount of available data per variant.

Given the importance of experimentation online and offline, parameters which change search behaviour need to be configurable as the search system is requested. This allows flexible investigation of the effect of changes and how these models are implemented must be given special attention to allow fast calculation of multiple models.



## THE LOCATION OF THE CENTRAL STATE

What we have proposed calls for the continuously-improving regulation of the search process. As it stands, the process acts on incoming feedback from users (via their events) and meets with a regulation process, which can then act upon that state to adapt it.

To avoid multiple state management or an externalisation of this control to the requesting system, we suggest for it to be a separate control process which is a part of the overall search system. This would need to assume the same adaptive processes as the system itself.

### Author:

Andreas Wagenmann

### Design:

Alicia García

### Collaborators:

Vanessa Farinha, Jorge Leal





eMPATHY.CO

LONDON | NEW YORK | ASTURIAS | GALICIA