A practical approach to measure the relevance and prevent regressions

#haystackconf
Berlin 2022-09-27
@a2lean

Context and Introduction

Welcome to…

Adelean, search engine experts

Building a2 and all.site

We want relevance!

Evaluate and improve it

Learning by doing

Context

- Big E-commerce service
  - Web and Mobile
  - Complex model (1500+ different shops with their own catalog)
- Always moving context
  - Evolving product indices: new items, seasonality...
  - Business Console with a lot of tunings knobs (Searchandizing)
  - Technical evolutions
  - Introducing a marketplace

Upon every change: *Will we break the search?* 😱

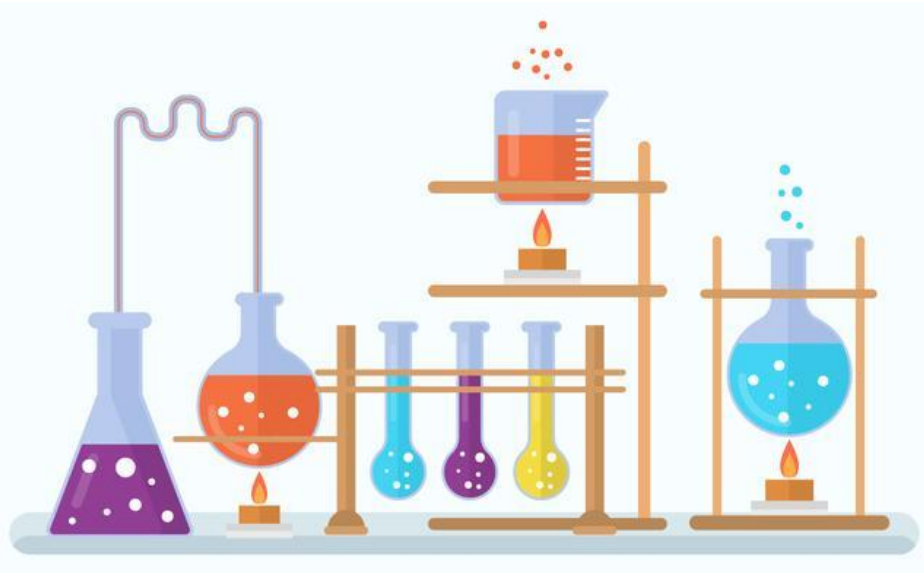
Ranking in e-commerce

Why

The secret recipe for direct **revenue**

Reasons for **buying**

Increases **KPIs** (Net Promoter Score...)



Challenges

Make this ranking **relevant**

… But we don’t want a **black box**

Understand our ranking

Make *relevance* measurable and comparable with a **score**

Measuring and Evaluating relevance

Judgment lists

A judgment list defines a document's relevance for a query.

Query	ID of the document	Relevance
white shoes	156542	3 (very good)
white shoes	849848	0 (bad)
white shoes	487984	1 (average)
white shoes	350588	3 (very good)
white shoes	468468	2 (good)

Metrics

Once you have a judgment list for a query, you can calculate metrics to evaluate the **relevancy of a list of results** for this query:

- **AP** : Average precision
- **DCG** : Discounted Cumulative Gain
- **NDCG** : Normalized Discounted Cumulative Gain
- **MRR** : Mean Reciprocal Rank
- **ERR** : Expected Reciprocal Rank

NDCG

- **DCG** uses a graded relevance scale to evaluate the usefulness, or gain, of a document based on its position in the result list
- Highly relevant documents appearing lower in a search result list should be penalized

Award a result by its relevance

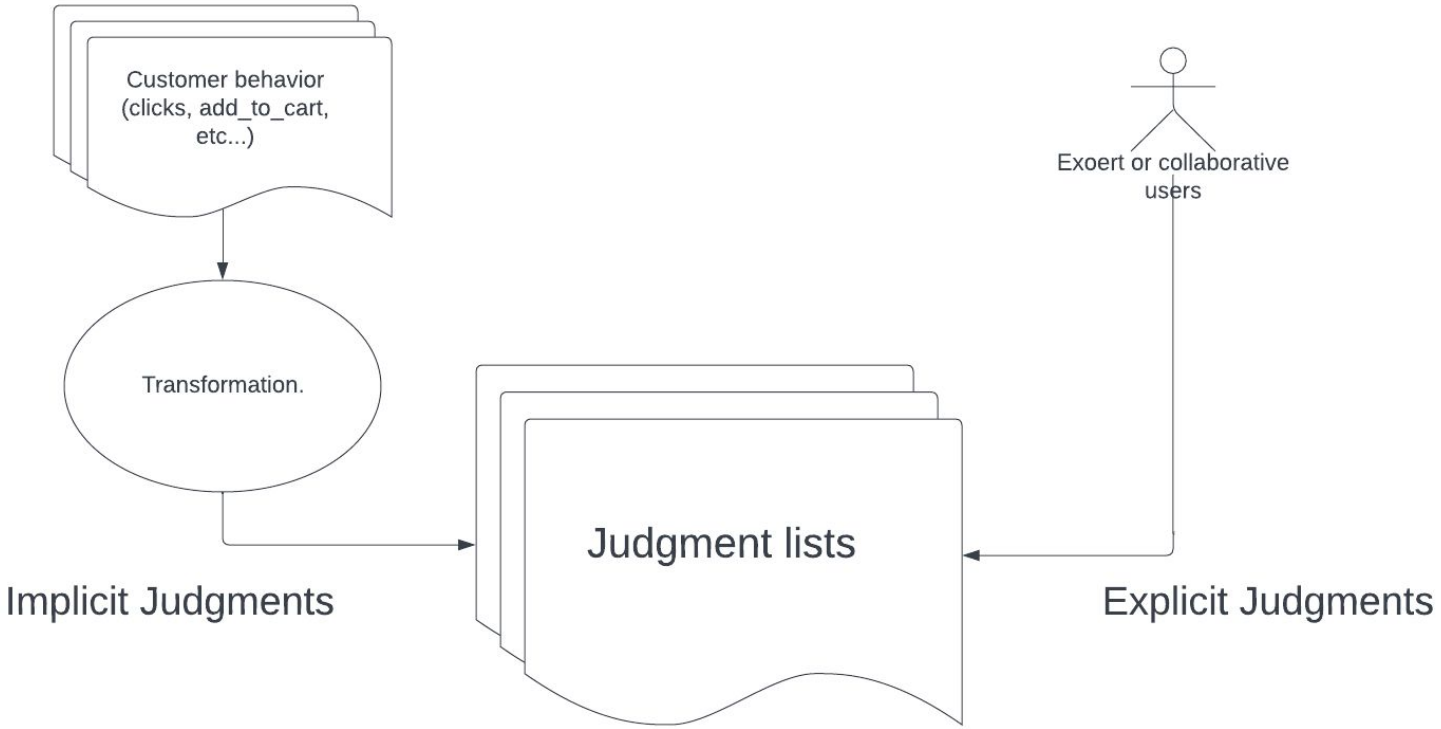
For all the results $\sum_{i=1}^{p} \frac{2^{rel_i} - 1}{\log_2(i+1)}$

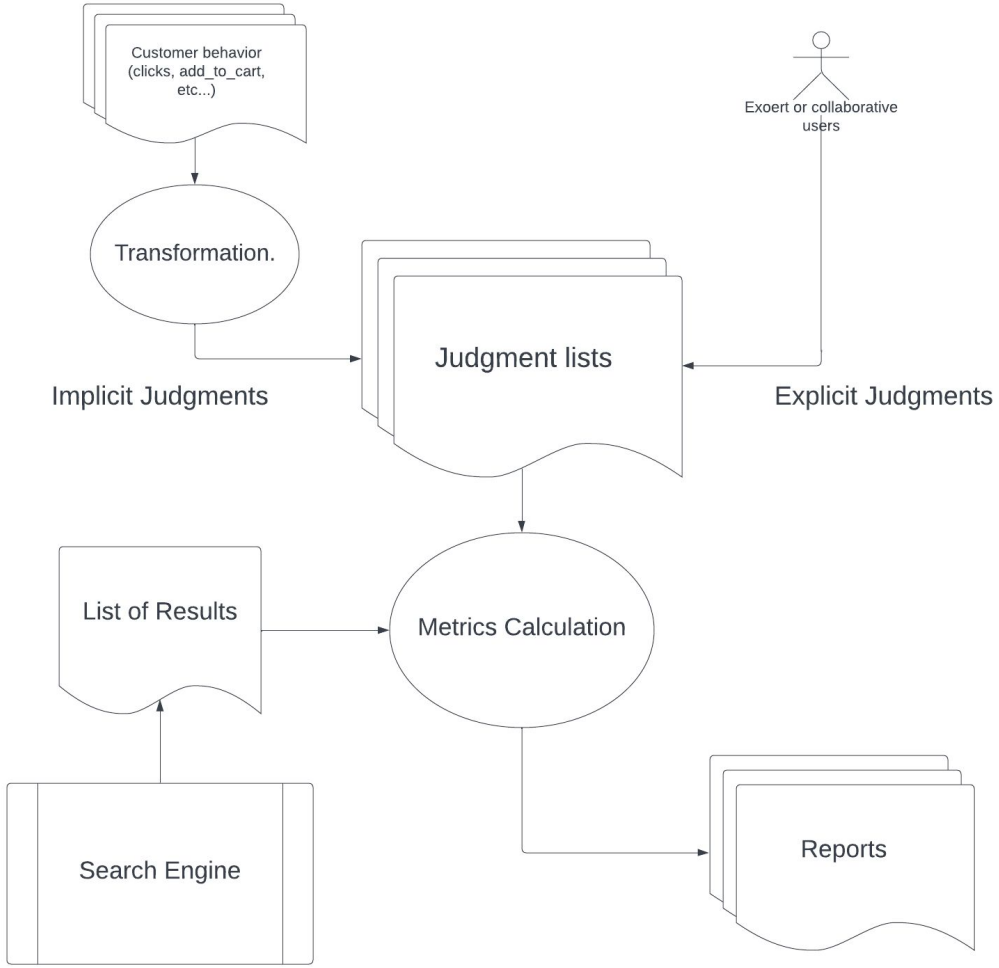
Add the scores Punish a result by its rank

$$NDCG = \frac{DCG}{iDCG}$$

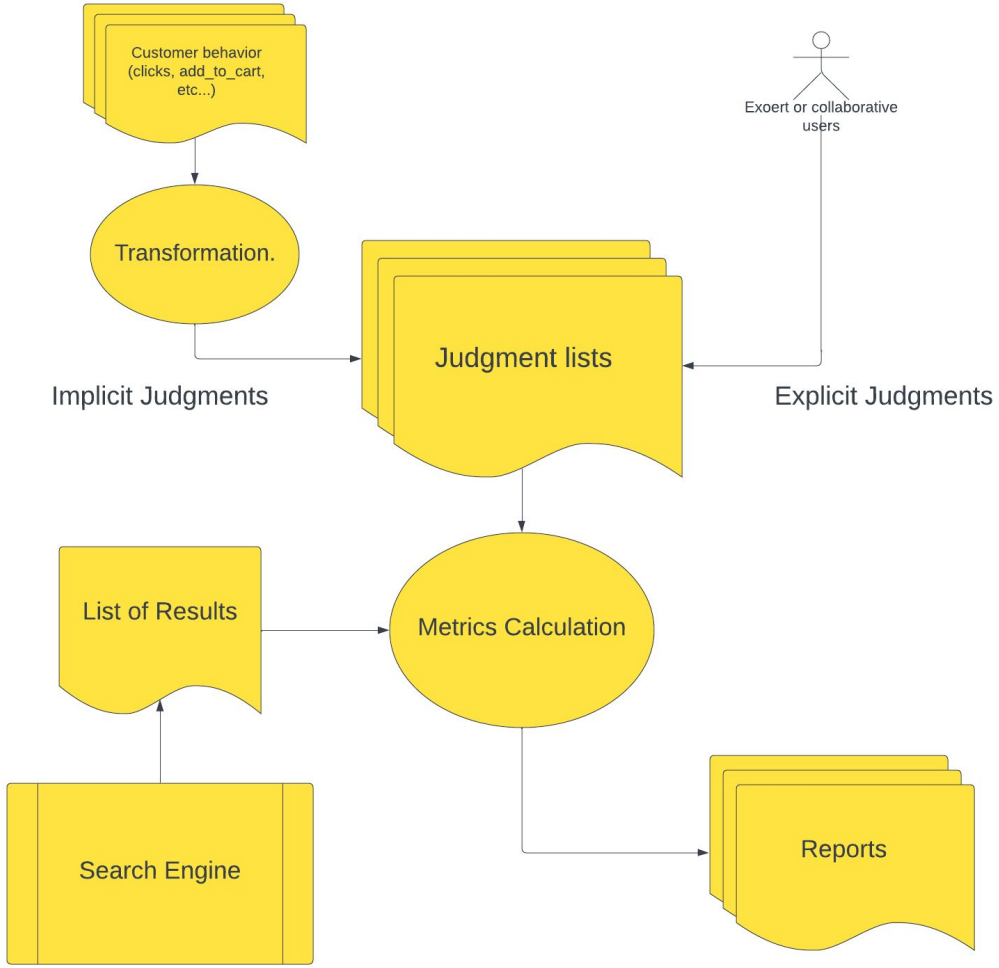
Ex : Based on my **judgment list**, the **NDCG** of the **list of results** returned by my search engine for “white shoes” is **0,87**

Building judgment lists

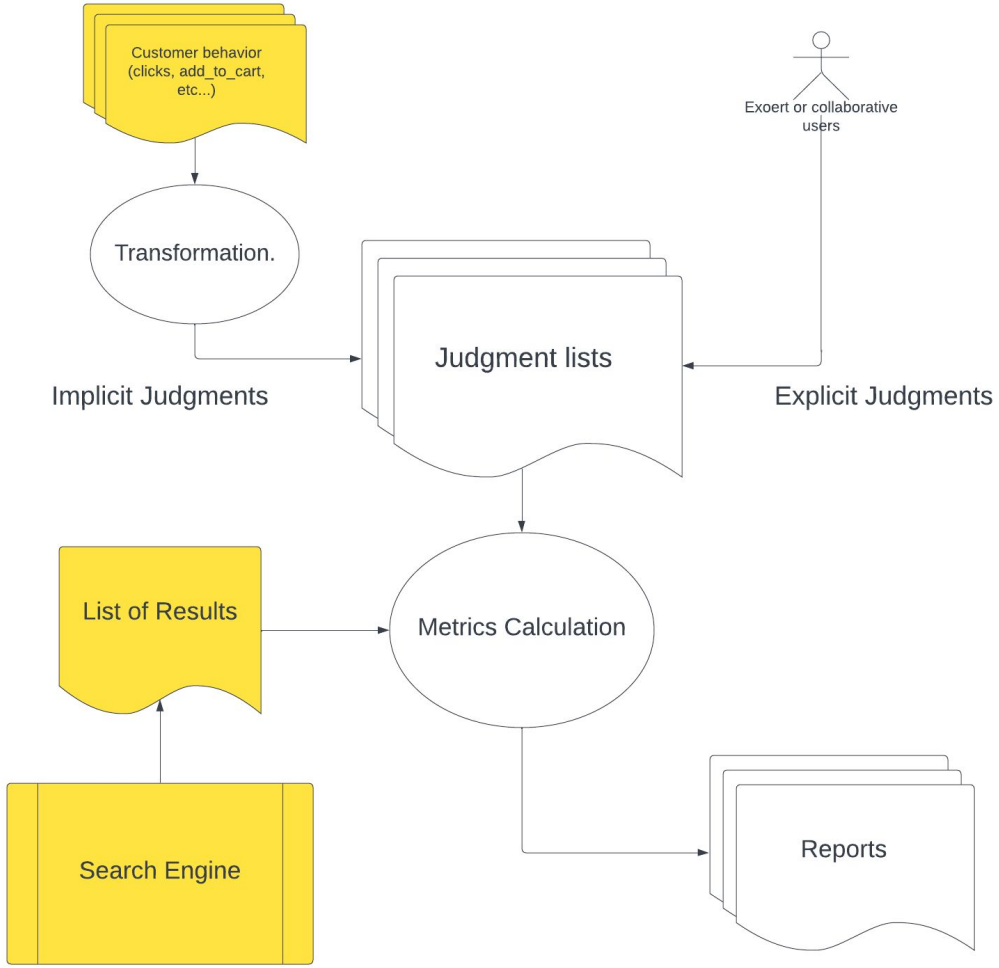



Customer behavior (clicks, add_to_cart, etc...)
Exoert or collaborative users
Transformation.
Judgment lists
Implicit Judgments
Explicit Judgments
List of Results
Metrics Calculation
Search Engine
Reports

Needed


Customer behavior (clicks, add_to_cart, etc...)
Exoert or collaborative users
Transformation.
Judgment lists
Implicit Judgments
Explicit Judgments
List of Results
Metrics Calculation
Search Engine
Reports

Start


Customer behavior (clicks, add_to_cart, etc...)
Exoert or collaborative users
Transformation.
Judgment lists
Implicit Judgments
Explicit Judgments
List of Results
Metrics Calculation
Search Engine
Reports

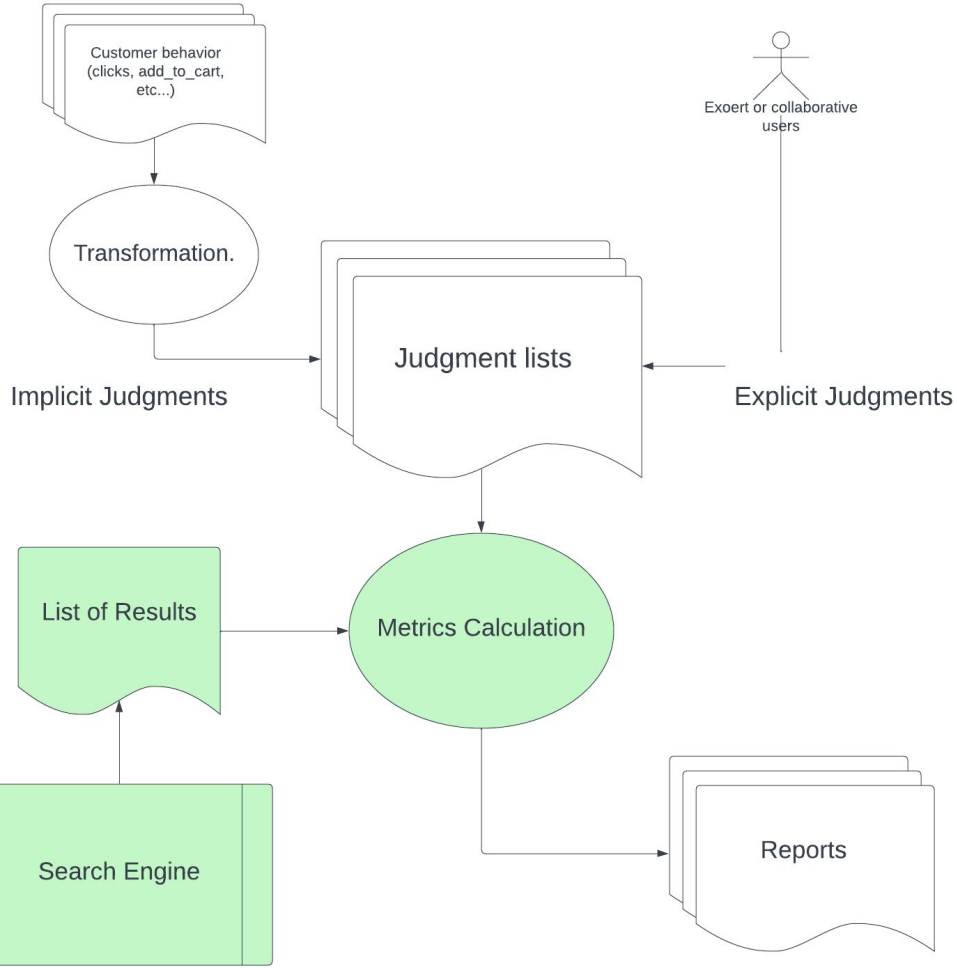
Off-the shelf

Elasticsearch’s ranking evaluation API

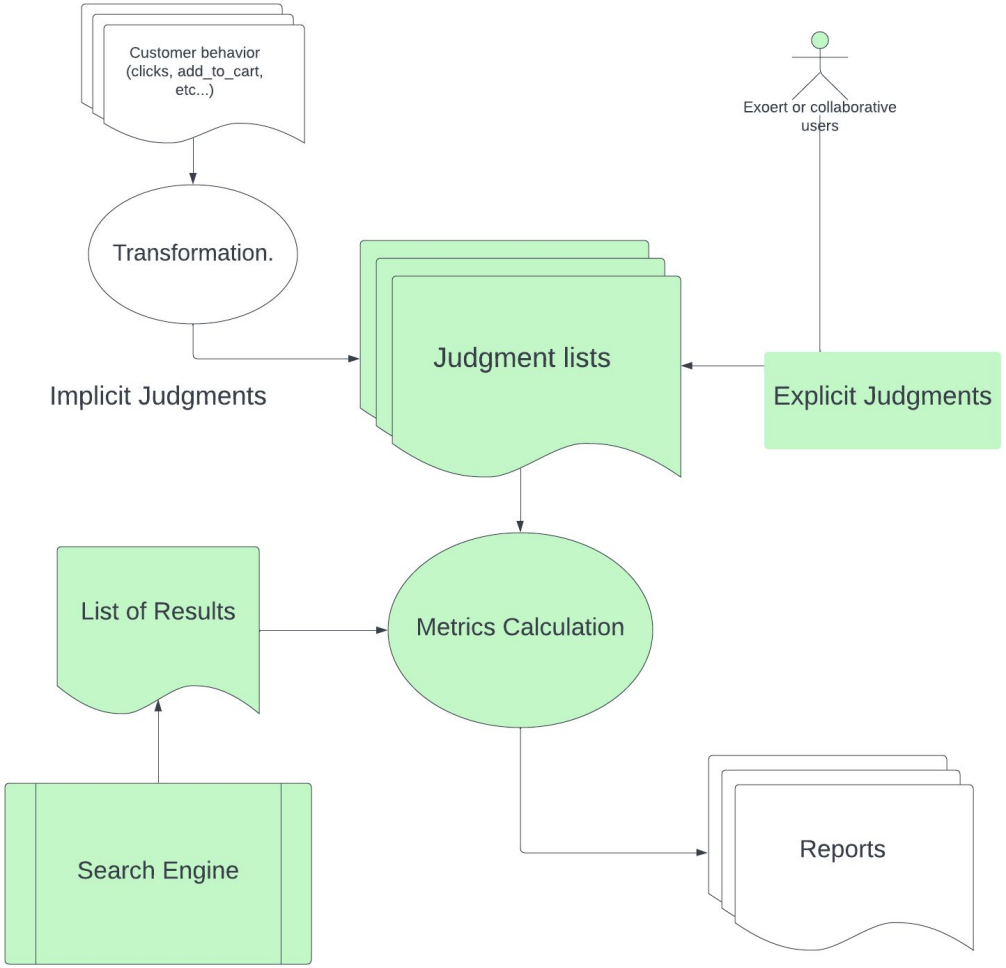
Quepid (OpenSource Connections)

RRE Relevance Rating Evaluator (Sease)

rank_eval

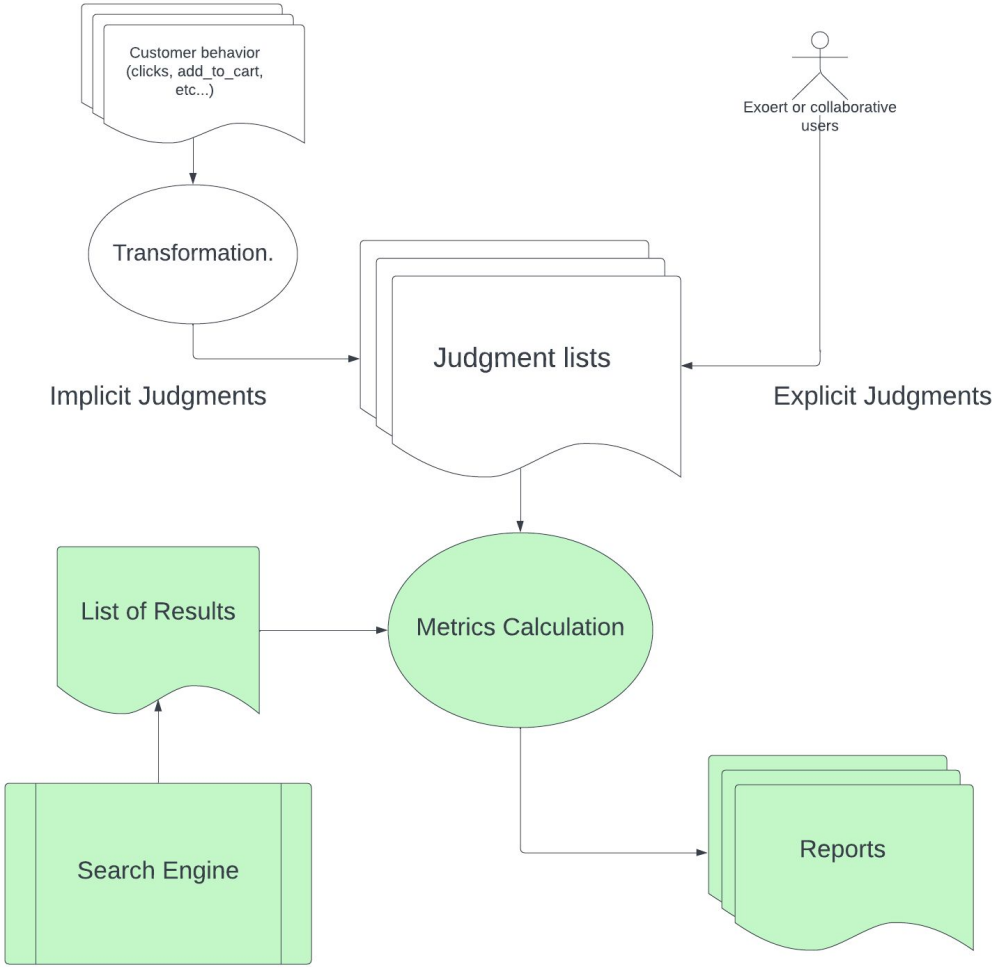

Customer behavior (clicks, add_to_cart, etc...)
Exoert or collaborative users
Transformation.
Judgment lists
Implicit Judgments
Explicit Judgments
List of Results
Metrics Calculation
Search Engine
Reports

Quepid



Quepid

Rated Ranking Evaluator



a2 relevance measuring tool

Why?

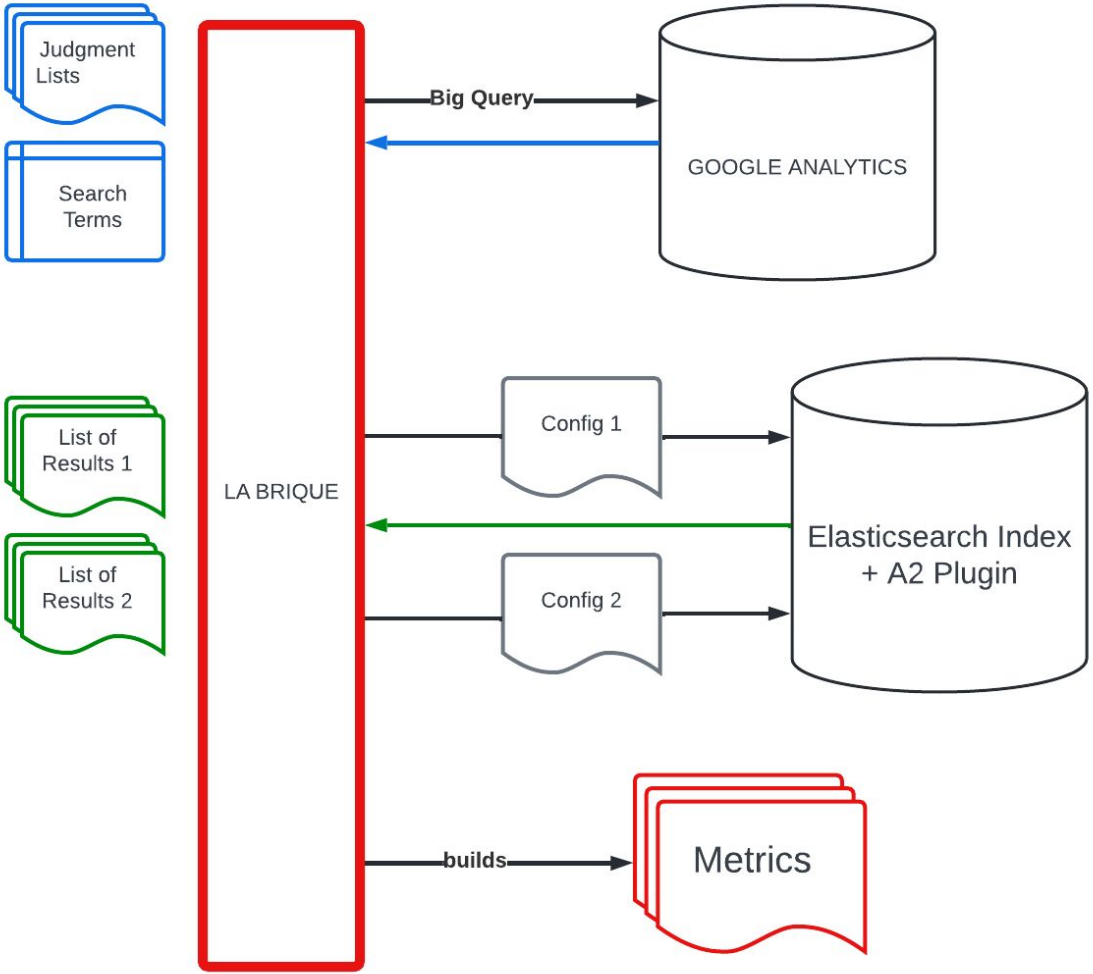
🤔 *Did my changes **really improve** the general relevance of my search engine?*

- A2 business team: non technical users who configure the search engine through a Business Console
  - They have a good understanding of products and retail field
  - They can change boost values, add synonyms, best bets...
- Users need a tool to **test** and guarantee the **non regression** on their changes

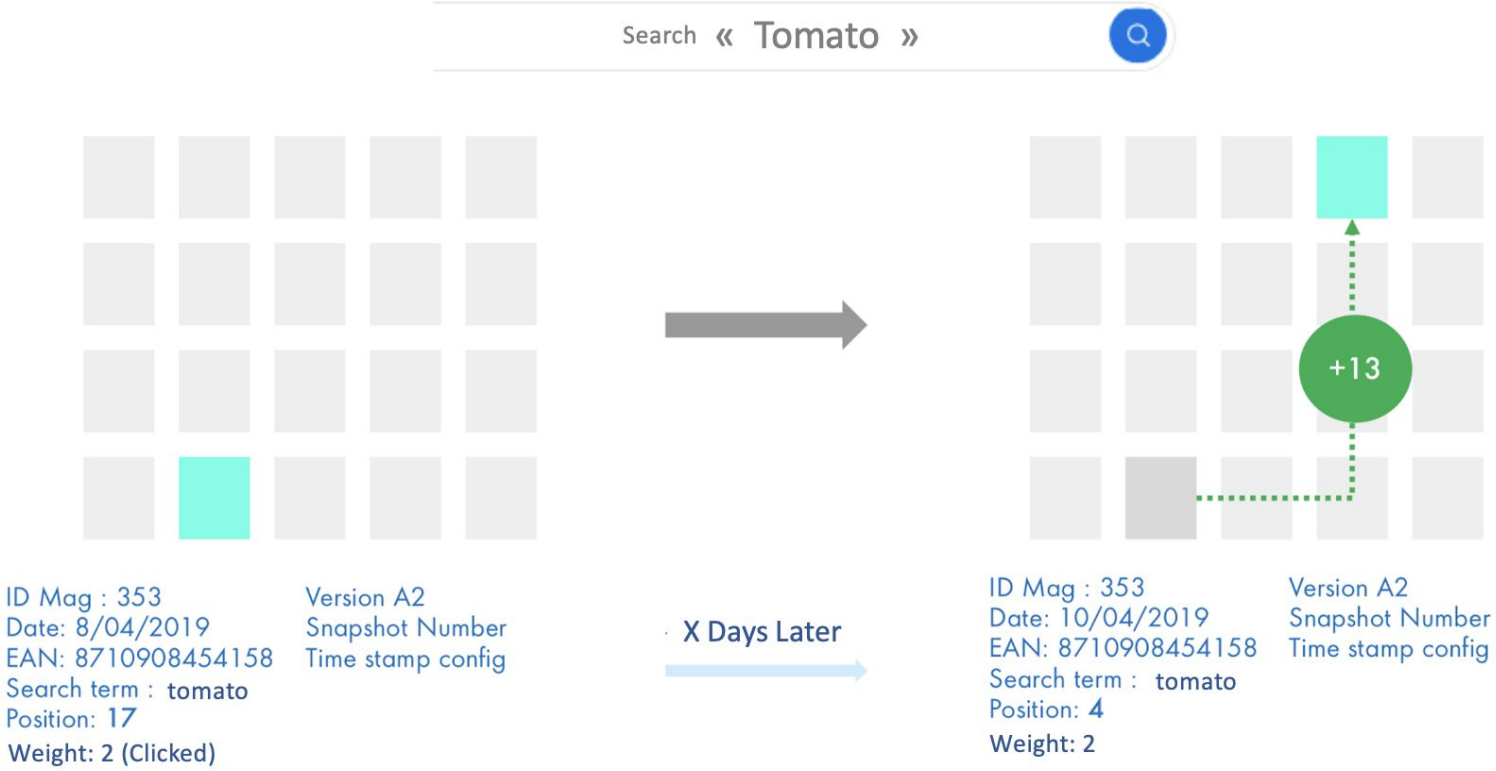
Challenges

- Build judgment lists from **user actions**
  - Choose the events, find simple formulas (clicks, add to cart), remove odds
- Metrics are comparable (and useful) if the corpus **does not change**
  - But retail products change constantly!
- **Operability**
  - Our tool must be used by non technical people, integrate in their everyday dashboard
- **Performance**
  - Our first version took hours to get user information, calculate judgment lists, then metrics.

The solution

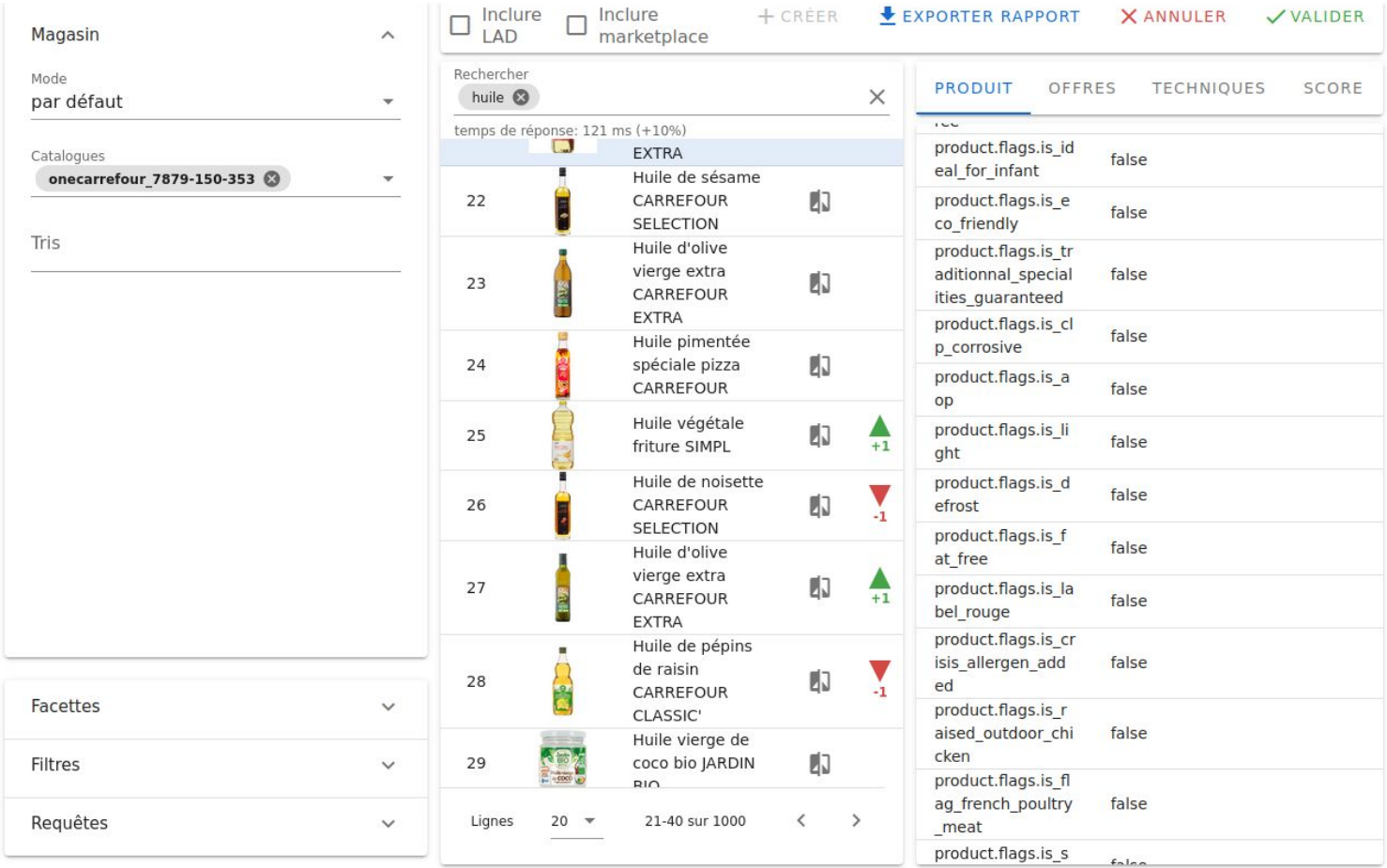


The solution



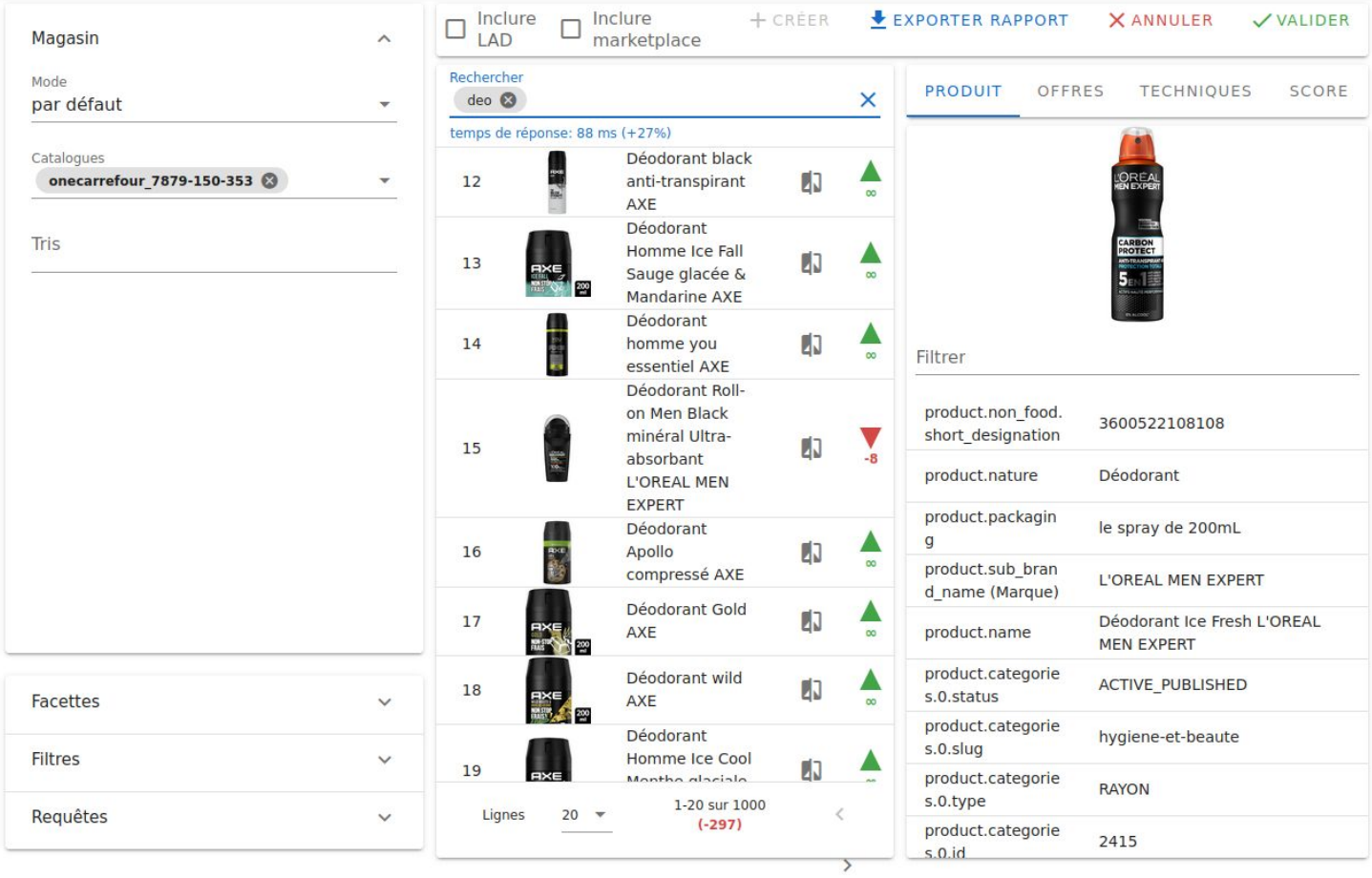
Demo

Compare Screen



Changing a configuration

The ranking changes



A2 relevance measure tool

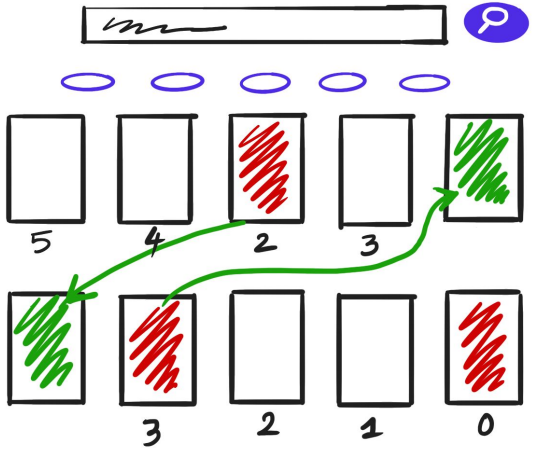
Search Term	Total Score	EAN	Event	Old Position	New Position	Delta Position	Score
mozzarella	-1						
		8000430135268	add_to_cart	10	12	-2	-0,5
		8000430135268	add_to_cart	10	12	-2	-0,5
mozzarella	-1,928571429						
		3560071018238	add_to_cart	11	14	-3	-0,642857143
		3560071018238	add_to_cart	11	14	-3	-0,642857143
		3560071018238	add_to_cart	11	14	-3	-0,642857143

Conclusions and next steps…

Our conclusions

Since we have implemented our tool

- We guarantee the **stability** and **non regression** of our engine
- We have given the possibility for the administrators to **test** their changes and see them in action before they go live
- We can **explain** the ranking and its evolutions



Next Steps

- Getting more implicit judgements with Analytics data
  - Complex click models
  - Timers
- More integration with the Business Console
  - Asynchronous validation of changes in production
  - Detailed information about regressions
- Test further configurations in addition to non-regression
  - Compute boosts automatically (see Quaerite)
  - Find the best combination of parameters

A practical approach to measure the relevance and prevent regressions

Thank you

Questions / Feedback / More ...

@a2lean

info@adelean.com

http://all.site

http://www.adelean.com

http://www.linkedin.com/company/adelean

http://www.meetup.com/fr-FR/search-and-data

HAYSTACK