A Proof-of-Concept for Automatic Facet Discovery in E-Commerce Search

Rauf Aliev

June 2017 (Updated September 2025)

Abstract

This paper addresses a significant challenge in e-commerce information retrieval: the failure of standard keyword search systems to correctly interpret complex user queries that contain product attributes. Queries such as ”blue XL Burton jacket” are often processed as a simple set of keywords, leading to irrelevant results and compelling users to engage in a laborious manual filtering process. We present a proof-of-concept (PoC) for an automatic facet discovery system designed to parse user queries, identify terms corresponding to product facets (e.g., color, brand, size), and apply these filters automatically. This research demonstrates a practical methodology for bridging the semantic gap between unstructured free-text search and structured faceted navigation, thereby enhancing result relevance and improving the overall user experience.

1 Introduction

Faceted search is an integral feature of modern e-commerce platforms, fundamentally improving the user's search and discovery experience. From a user-centric perspective, faceted navigation decomposes search results into multiple, orthogonal categories (facets), each with corresponding value counts. This paradigm enables users to iteratively refine, or ”drill down,” into the result set by applying filters in any desired sequence.

The utility of this functionality is particularly evident when interacting with large-scale product catalogs, as it substantially improves product findability, mitigates user frustration, and provides a structured navigational framework. Furthermore, faceted search architectures programmatically generate relevant landing pages for long-tail keyword queries, a long-standing strategy in search engine optimization that was traditionally accomplished through static category pages. A video demonstration of the system is available online[1].

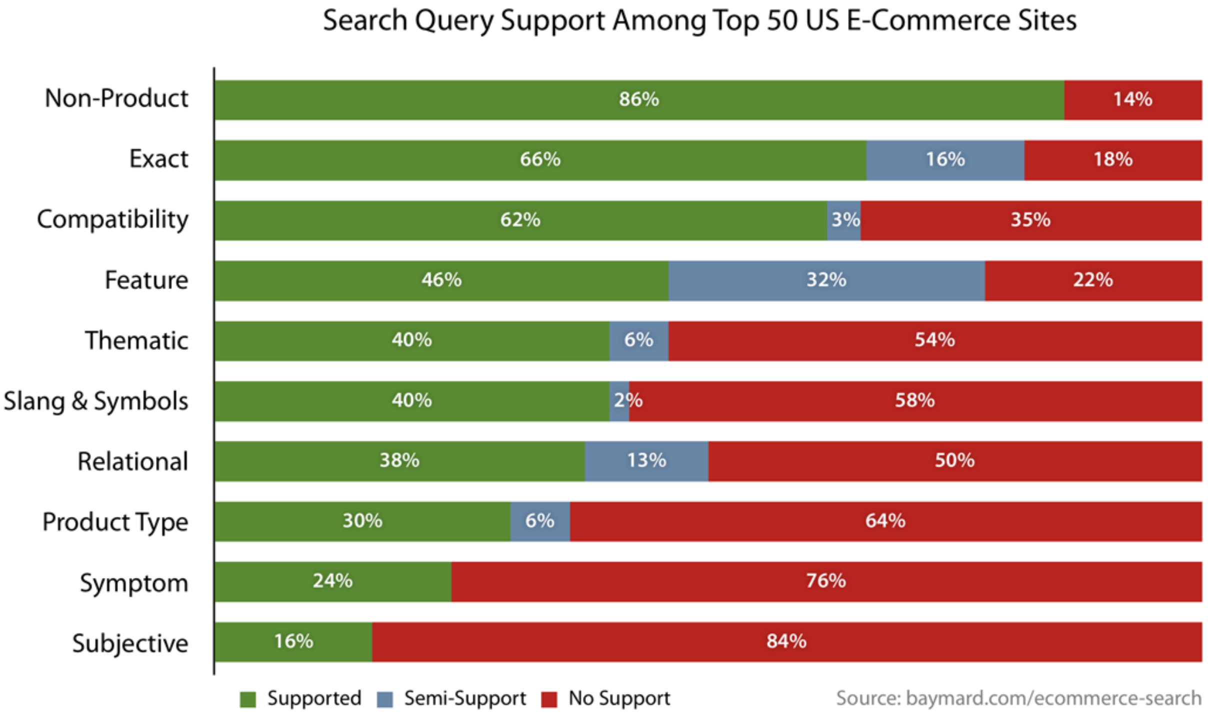
2 A Taxonomy of User Search Queries

Empirical research from the Baymard Institute classifies user search behavior into 12 distinct query types, the majority of which are inadequately supported by out-of-the-box e-commerce search engines. These types are:

1. **Exact Searches**: Queries for specific products via title or model number (e.g., *Keurig K45*).
2. **Product Type Searches**: Broad queries for product categories (e.g., *Sandals*).
3. **Symptom Searches**: Problem-based queries where the user seeks a product as a solution (e.g., *“stained rug”*).
4. **Non-Product Searches**: Informational queries regarding policies or company details.
5. **Feature Searches**: Queries specifying particular product attributes (e.g., *Waterproof cameras*).
6. **Thematic Searches**: Queries for abstract or conceptual categories (e.g., *“Living room rug”*).
7. **Relational Searches**: Queries based on a product's association with another entity (e.g., *Movies starring Tom Hanks*).

[1] https://player.vimeo.com/video/223100657

8. **Compatibility Searches**: Queries for products compatible with another item (e.g., *Lenses for Nikon D7000*).
9. **Subjective Searches**: Queries using non-objective, qualitative terms (e.g., *"High-quality kettles"*).
10. **Slang, Abbreviation, and Symbol Searches**: Queries employing linguistic shortcuts (e.g., *Sleeping bag -10 deg*).
11. **Implicit Searches**: Queries that omit context-dependent qualifiers.
12. **Natural Language Searches**: Queries formulated in complete sentences.



Source: baymard.com/ecommerce-search

Figure 1: A chart from Baymard.com illustrating the poor support for various e-commerce search query types.

While platforms such as SAP Commerce provide foundational support for some of these types, the implementation often lacks the necessary sophistication for accurate intent interpretation. Neither the platform nor its underlying SOLR search engine can natively associate query terms with semantic concepts. The proposed PoC serves as a semantic bridge between these free-text queries and the platform's powerful faceted search capabilities.

3 The Challenge with Conventional Faceted Search

Conventional faceted search implementations present known usability challenges. Although platforms display facets relevant to an initial query, the presence of attribute-related terms within the query itself can paradoxically cause those same facets to be excluded from the results. For instance, in the query "blue armada jacket XXL," a standard search engine processes all four terms as a free-text request, returning only documents containing all four keywords.

This approach is fundamentally flawed and leads to a cumbersome, multi-step user journey.

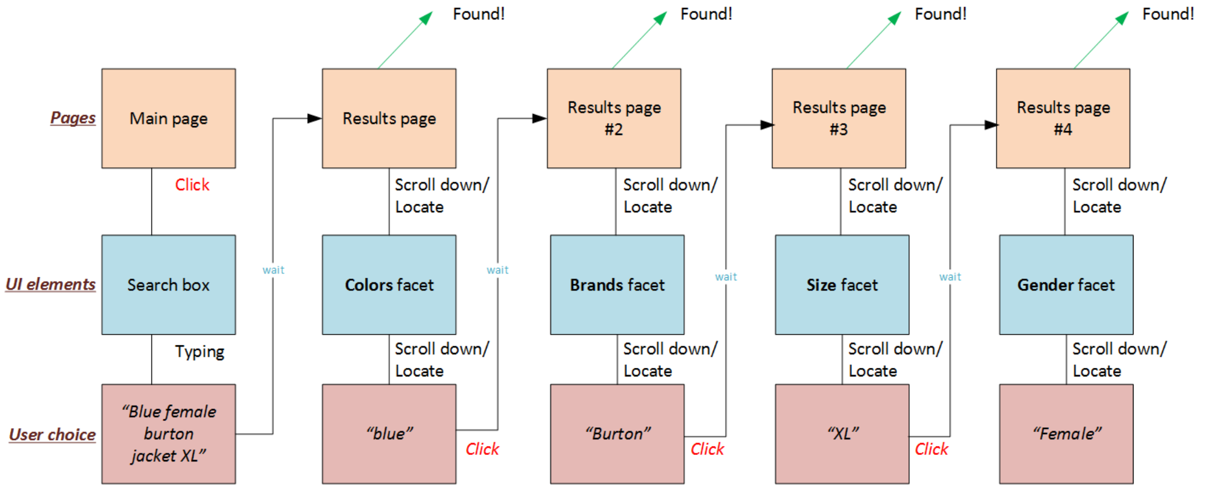
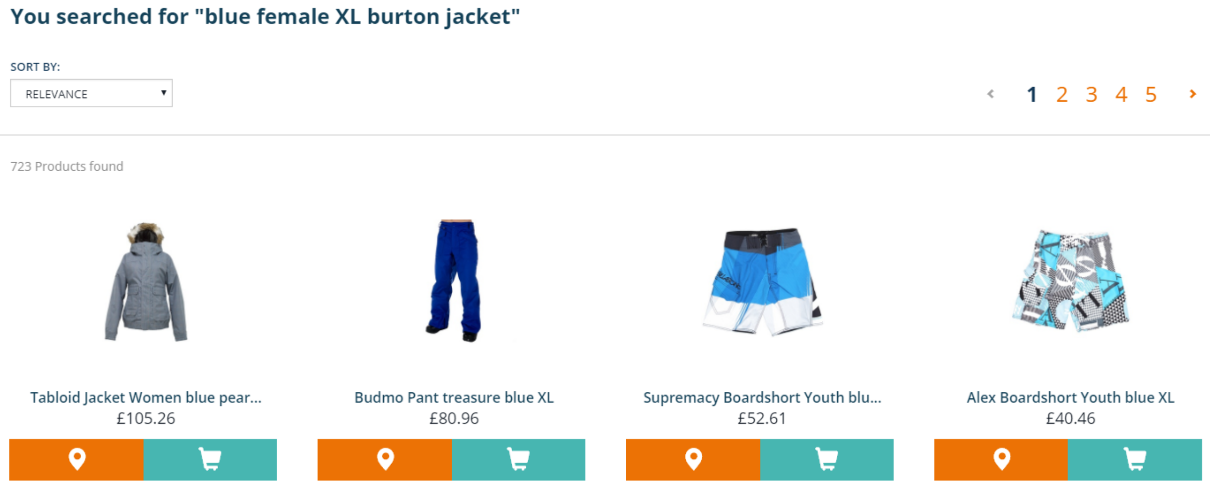


Figure 2: A typical e-commerce interface showing multiple facet categories on a sidebar, which users must manually interact with to refine search results.

A default SAP Commerce implementation, for example, yields highly irrelevant results for the query "blue female XL Burton jacket".


You searched for "blue female XL burton jacket"

SORT BY:
RELEVANCE

‹ 1 2 3 4 5 ›

723 Products found

Tabloid Jacket Women blue pear...
£105.26

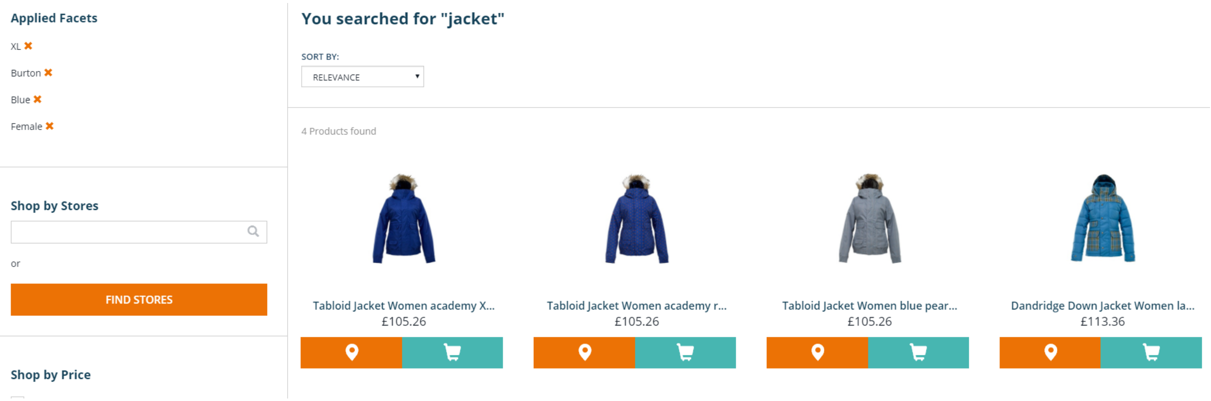
Budmo Pant treasure blue XL
£80.96

Supremacy Boardshort Youth blu...
£52.61

Alex Boardshort Youth blue XL
£40.46

Figure 3: Standard SAP Commerce Cloud search results for "blue female XL Burton jacket", showing irrelevant products.

The results include items that are not jackets, not blue, not for women, and mostly not from the specified brand. In contrast, the proposed PoC delivers highly relevant results for the identical query.



Applied Facets

XL ✖

Burton ✖

Blue ✖

Female ✖

Shop by Stores

or

FIND STORES

Shop by Price

You searched for "jacket"

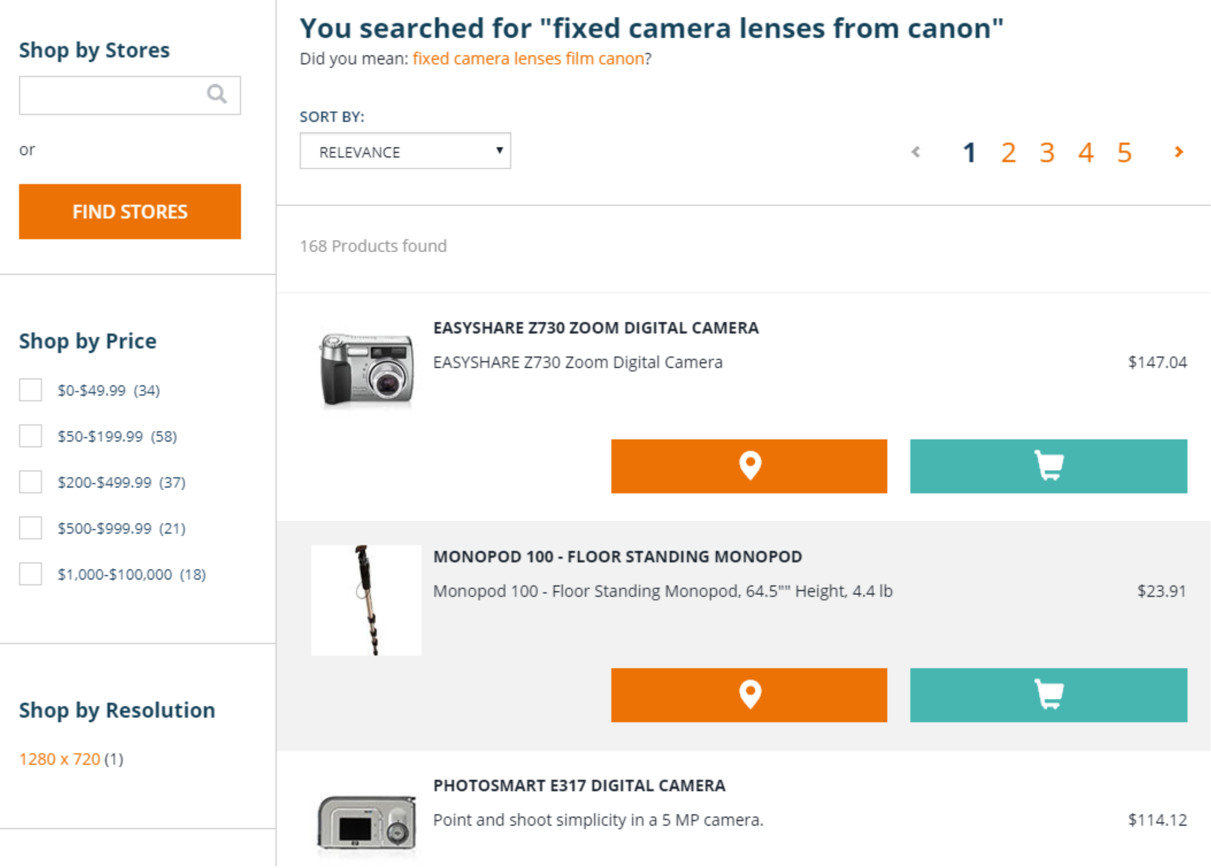
SORT BY: RELEVANCE

4 Products found

Tabloid Jacket Women academy X...	£105.26
Tabloid Jacket Women academy r...	£105.26
Tabloid Jacket Women blue pear...	£105.26
Dandridge Down Jacket Women la...	£113.36

Figure 4: PoC results for the identical query, showing correctly filtered, relevant products.

This performance improvement is consistent across different product domains. For an electronics catalog, the query “fixed camera lenses from canon” on a standard system yields irrelevant products.



Shop by Stores

or

FIND STORES

Shop by Price

□ $0-$49.99 (34)

□ $50-$199.99 (58)

□ $200-$499.99 (37)

□ $500-$999.99 (21)

□ $1,000-$100,000 (18)

Shop by Resolution

1280 x 720 (1)

You searched for "fixed camera lenses from canon"

Did you mean: fixed camera lenses film canon?

SORT BY: RELEVANCE

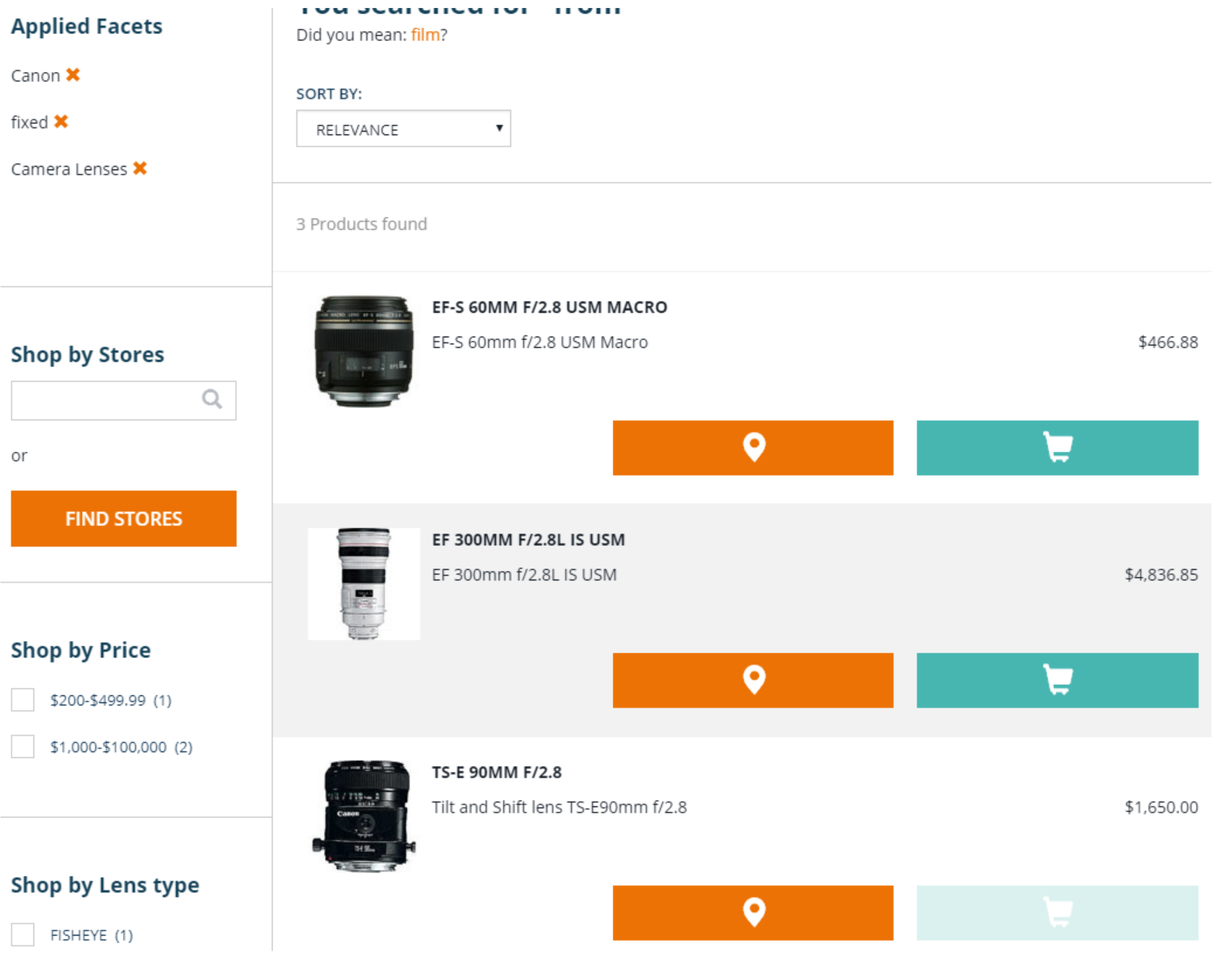
‹ 1 2 3 4 5 ›

168 Products found

EASYSHARE Z730 ZOOM DIGITAL CAMERA	EASYSHARE Z730 Zoom Digital Camera	$147.04
MONOPOD 100 - FLOOR STANDING MONOPOD	Monopod 100 - Floor Standing Monopod, 64.5"" Height, 4.4 lb	$23.91
PHOTOSMART E317 DIGITAL CAMERA	Point and shoot simplicity in a 5 MP camera.	$114.12

Figure 5: Standard search performance for “Fixed camera lens from Canon”.

The proposed system, however, correctly identifies and filters for the requested products.


Applied Facets

Canon ✖

fixed ✖

Camera Lenses ✖

Did you mean: film?

SORT BY: RELEVANCE

3 Products found

EF-S 60MM F/2.8 USM MACRO
EF-S 60mm f/2.8 USM Macro — $466.88

EF 300MM F/2.8L IS USM
EF 300mm f/2.8L IS USM — $4,836.85

TS-E 90MM F/2.8
Tilt and Shift lens TS-E90mm f/2.8 — $1,650.00

Shop by Stores

or

FIND STORES

Shop by Price

□ $200-$499.99 (1)

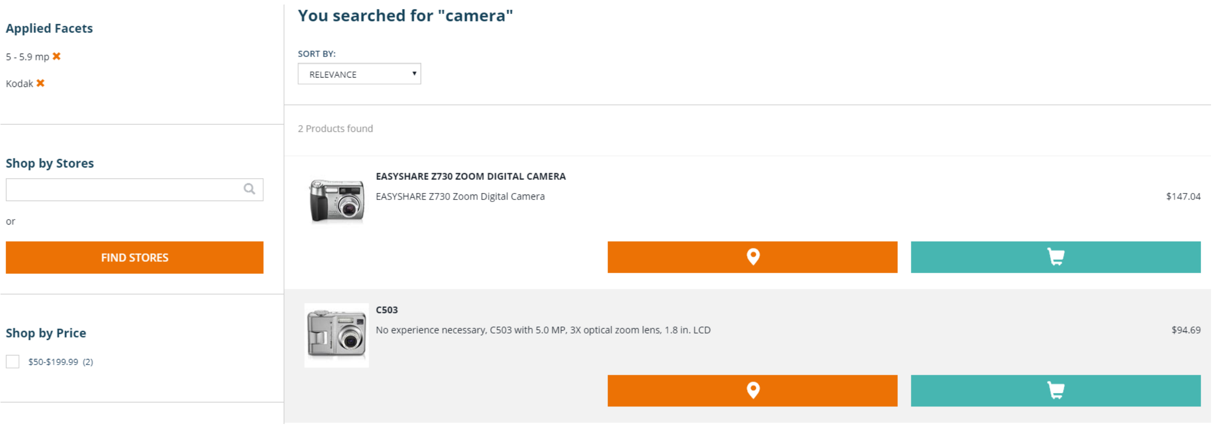
□ $1,000-$100,000 (2)

Shop by Lens type

□ FISHEYE (1)

Figure 6: PoC performance for “Fixed camera lens from Canon”.

The system can also interpret numerical ranges, correctly applying a filter for the ”5-5.9 Mp” facet range for a query like “5 mp kodak camera”.


You searched for "camera"

Applied Facets

5 - 5.9 mp ✖

Kodak ✖

SORT BY: RELEVANCE

2 Products found

EASYSHARE Z730 ZOOM DIGITAL CAMERA
EASYSHARE Z730 Zoom Digital Camera — $147.04

C503
No experience necessary, C503 with 5.0 MP, 3X optical zoom lens, 1.8 in. LCD — $94.69

Shop by Stores

or

FIND STORES

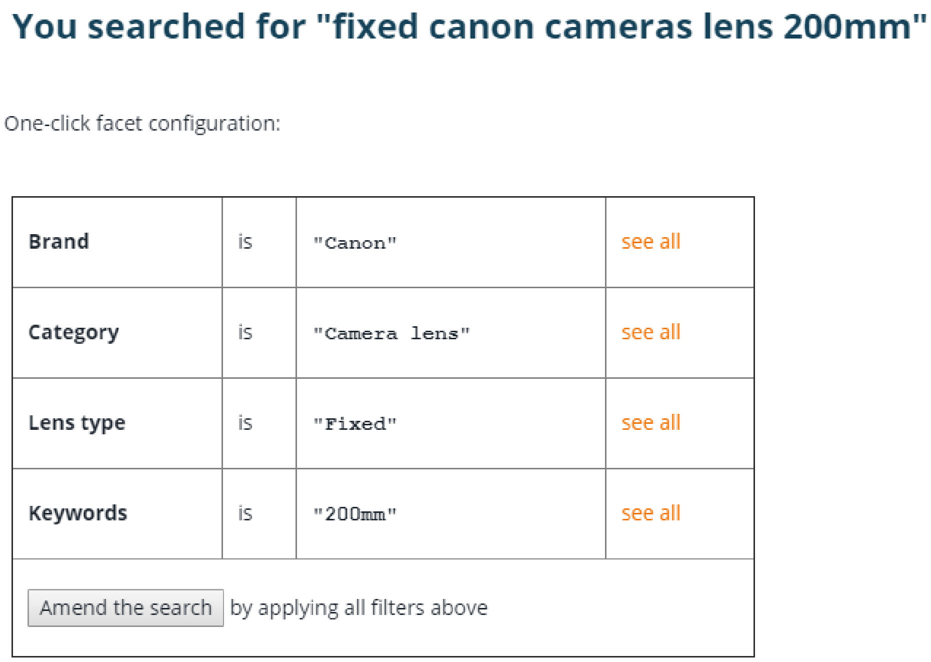
Shop by Price

□ $50-$199.99 (2)

Figure 7: The PoC applying a facet range for the query “5 Mp Kodak camera”.

4 Implementation Strategy

While the PoC implements fully automatic query interpretation, a production deployment would benefit from A/B testing to determine the optimal strategy. An alternative, non-automatic approach involves presenting the interpreted query as a one-click suggestion alongside the standard keyword search results.



You searched for "fixed canon cameras lens 200mm"

One-click facet configuration:

Brand	is	"Canon"	see all
Category	is	"Camera lens"	see all
Lens type	is	"Fixed"	see all
Keywords	is	"200mm"	see all

Amend the search by applying all filters above

Figure 8: A conceptual mockup of a search suggestion panel as an alternative to fully automatic interpretation.

5 Technical Architecture

The system's architecture is depicted below. It operates by analyzing the input query to extract terms that correspond to known facet values.

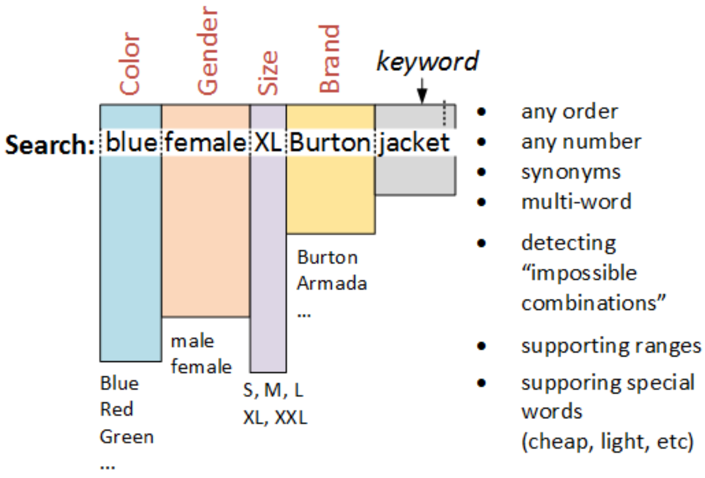
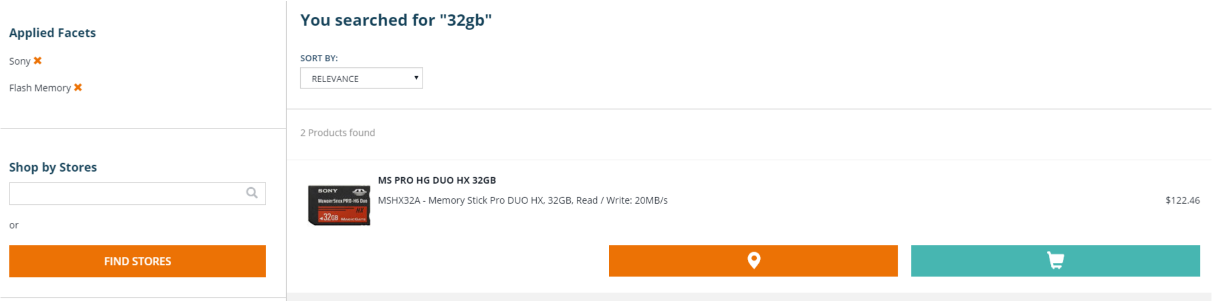


Figure 9: High-level system architecture for the automatic facet discovery system.

A primary technical challenge is resolving ambiguity. For example, the query "*Canon flash memory*" is ambiguous if the product catalog contains no flash memory manufactured by Canon. This ambiguity is resolved when the catalog contains products that satisfy all specified attributes.



Applied Facets

Sony ✖

Flash Memory ✖

Shop by Stores

or

FIND STORES

You searched for "32gb"

SORT BY:

RELEVANCE ▾

2 Products found

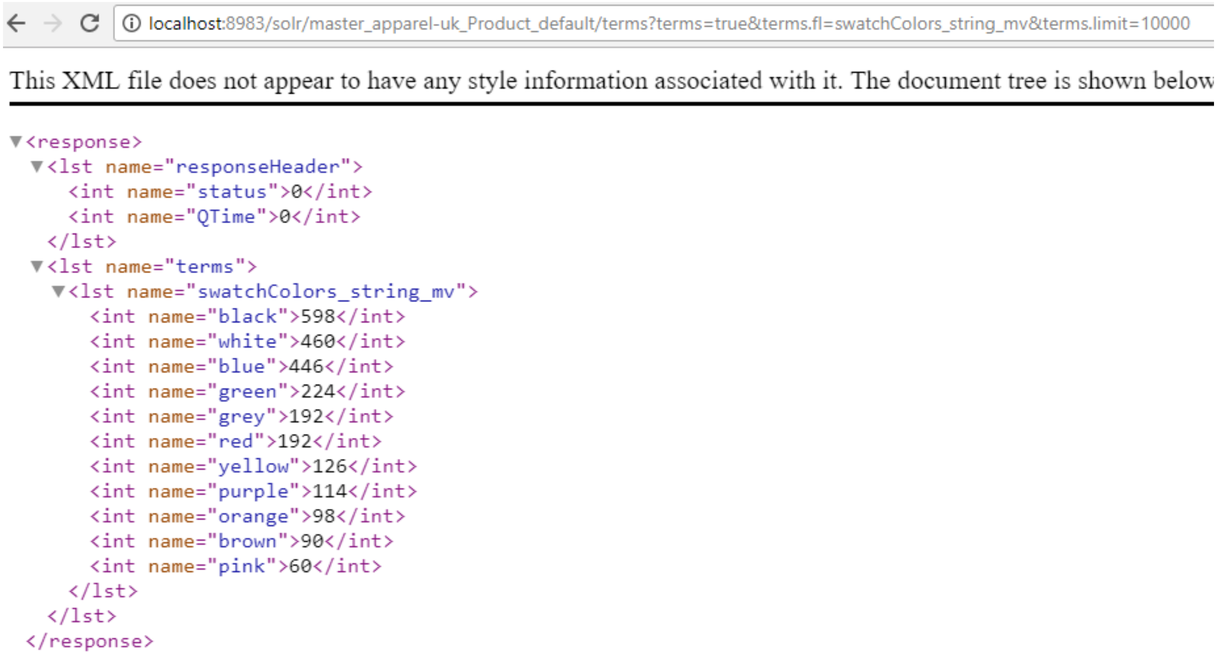
MS PRO HG DUO HX 32GB

MSHX32A - Memory Stick Pro DUO HX, 32GB, Read / Write: 20MB/s

$122.46

Figure 10: The system correctly applying multiple facets for the unambiguous query “Sony Flash memory 32Gb”.

To facilitate this mapping, the system retrieves all available facet values directly from the SOLR index using its built-in ”terms” request handler. For optimal performance, this technique requires SOLR fields configured with a `KeywordTokenizer` and without stemming filters.



localhost:8983/solr/master_apparel-uk_Product_default/terms?terms=true&terms.fl=swatchColors_string_mv&terms.limit=10000

This XML file does not appear to have any style information associated with it. The document tree is shown below

```
▼<response>
  ▼<lst name="responseHeader">
      <int name="status">0</int>
      <int name="QTime">0</int>
    </lst>
  ▼<lst name="terms">
    ▼<lst name="swatchColors_string_mv">
        <int name="black">598</int>
        <int name="white">460</int>
        <int name="blue">446</int>
        <int name="green">224</int>
        <int name="grey">192</int>
        <int name="red">192</int>
        <int name="yellow">126</int>
        <int name="purple">114</int>
        <int name="orange">98</int>
        <int name="brown">90</int>
        <int name="pink">60</int>
      </lst>
    </lst>
  </response>
```

Figure 11: The SOLR terms component response, listing all values for a given facet field.

The PoC employs an efficient strategy by only processing facets that are returned by an initial standard keyword search.

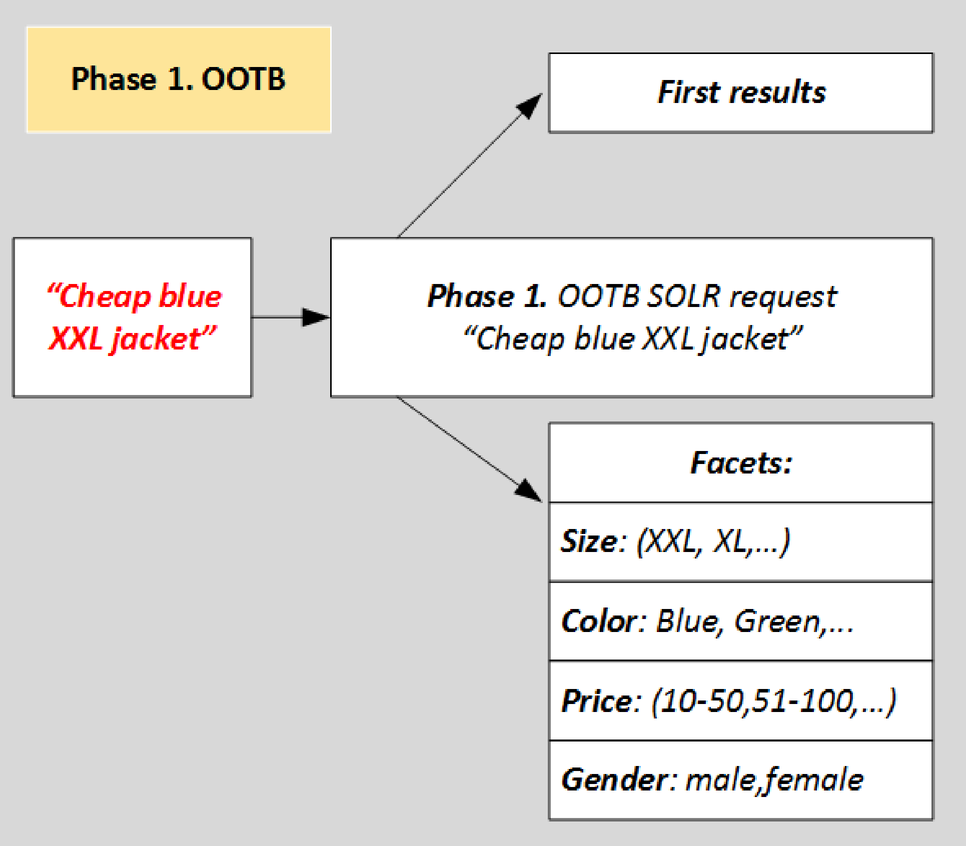


Figure 12: Step 1: The system performs an initial full-text search to retrieve a set of relevant facets.

Once terms are mapped to facets, the remaining query words are categorized as stopwords, special commands, or residual keywords.

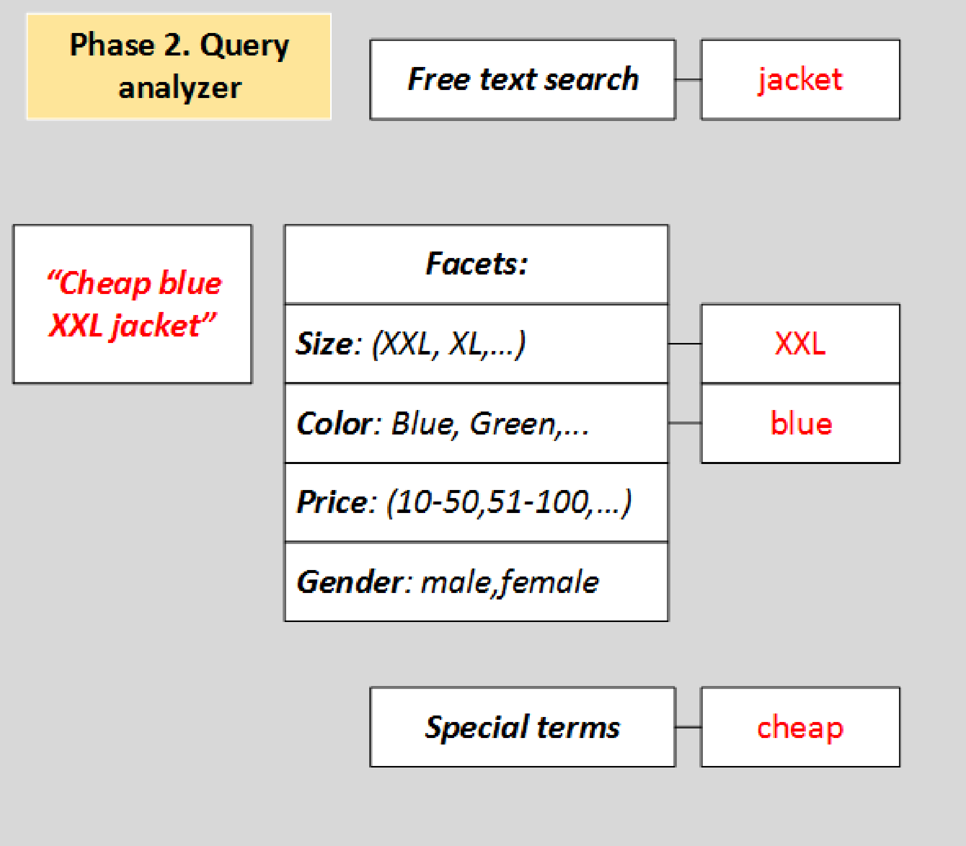
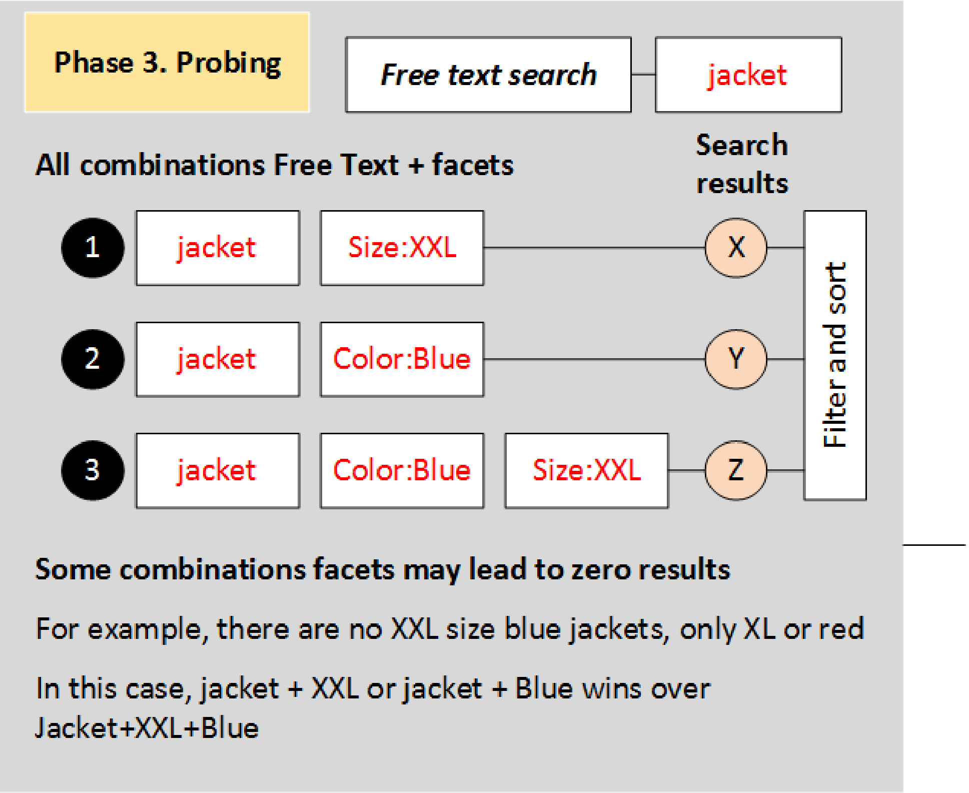


Figure 13: Step 2: Query terms are mapped against the values of the retrieved facets.

The system then constructs a new, hybrid query combining the discovered facet filters with the remaining keywords.



Phase 3. Probing

Free text search

jacket

All combinations Free Text + facets

Search results

Some combinations facets may lead to zero results

For example, there are no XXL size blue jackets, only XL or red

In this case, jacket + XXL or jacket + Blue wins over Jacket+XXL+Blue

Figure 14: Step 3: A new, hybrid query is constructed using the identified facets and remaining keywords.

To handle conflicts where a term matches multiple facets (e.g., ”red” as a color vs. part of the brand ”Red Hat”), the PoC implements a simple disambiguation logic: it executes a count for each interpretation and proceeds with the option that yields a non-zero result set.

6 Conclusion and Future Work

This paper has presented a proof-of-concept for an automatic facet discovery system. By programmatically parsing unstructured user queries to identify and apply corresponding structured facet filters, the proposed system effectively bridges the semantic gap between keyword-based retrieval and faceted navigation.

The current implementation, while promising, has several recognized limitations. The system's reliance on exact-match string comparisons is inherently brittle. Furthermore, the conflict resolution logic is heuristic-based and could be substantially improved. The PoC does not yet support complex linguistic constructs.

Future work will focus on integrating more sophisticated Natural Language Processing (NLP) techniques. The exploration of libraries such as OpenNLP is underway to incorporate capabilities like stemming and synonym expansion. Further research will also involve developing a more robust disambiguation model, potentially leveraging machine learning. Finally, a quantitative evaluation through rigorous A/B testing is required to measure the system's impact on key performance indicators. Ultimately, the continued development of such intelligent query interpretation systems is a critical step toward creating more intuitive and efficient information retrieval experiences.