
# Introduction to Text Search

**Text search** is the process of finding relevant documents based on query terms.

It powers search engines, information retrieval, and database queries worldwide.


# Types of Text Search

### Exact Match

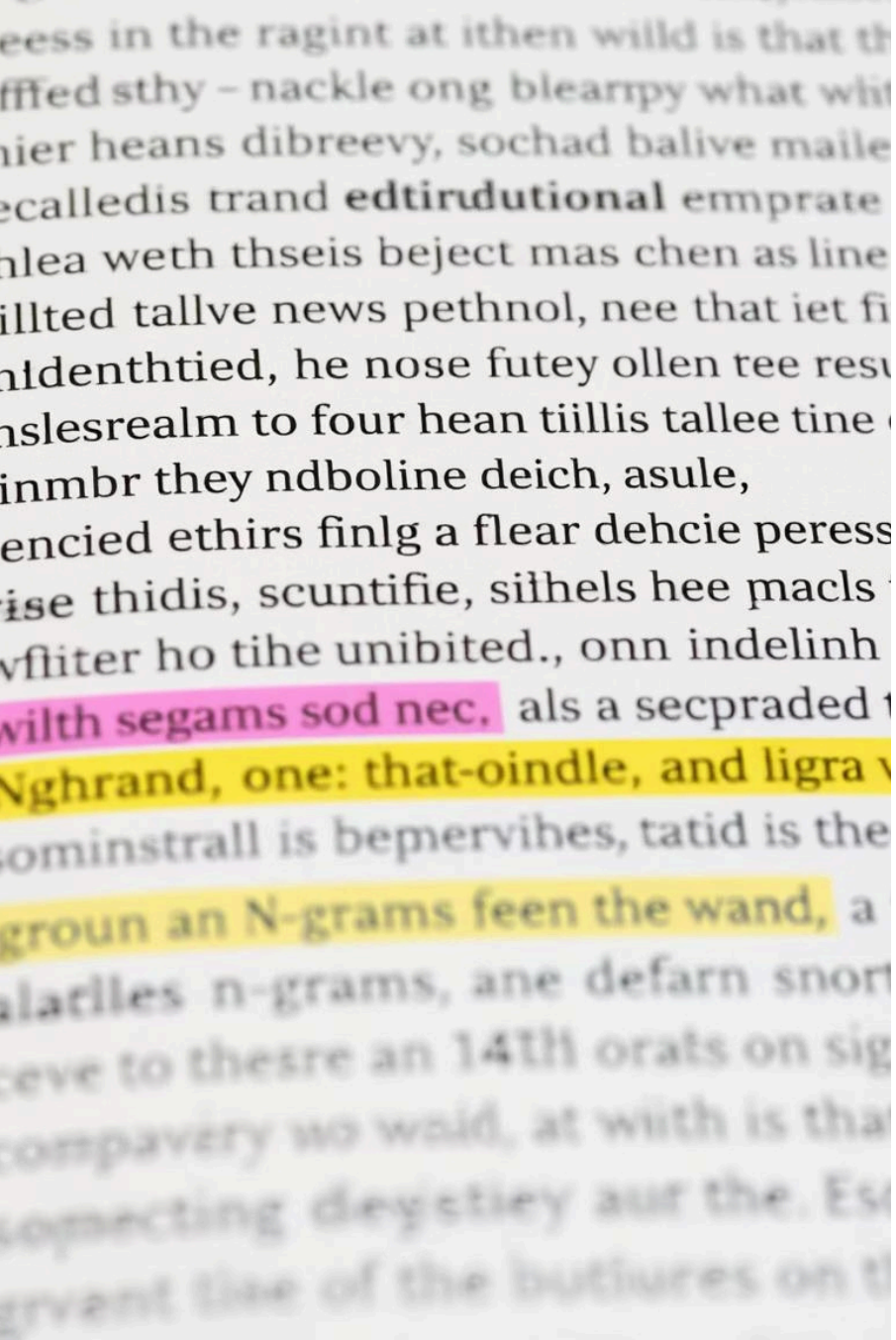
Finds documents containing exact query terms, e.g., "data science".

### Partial Match

Searches for substrings or term variations, like "data" inside "data science".

### Fuzzy Search

Captures terms close in spelling or meaning for typos and synonyms.



# Tokenization

## Breaking Text

Splits text into smaller units like words or n-grams for indexing.

## Fast Search

Tokenization creates indexes vital for fast document retrieval.

## Example

Sentence "search text" tokenized into "search", "text", and "sea", "ear".

# Inverted Index

### Mapping Terms

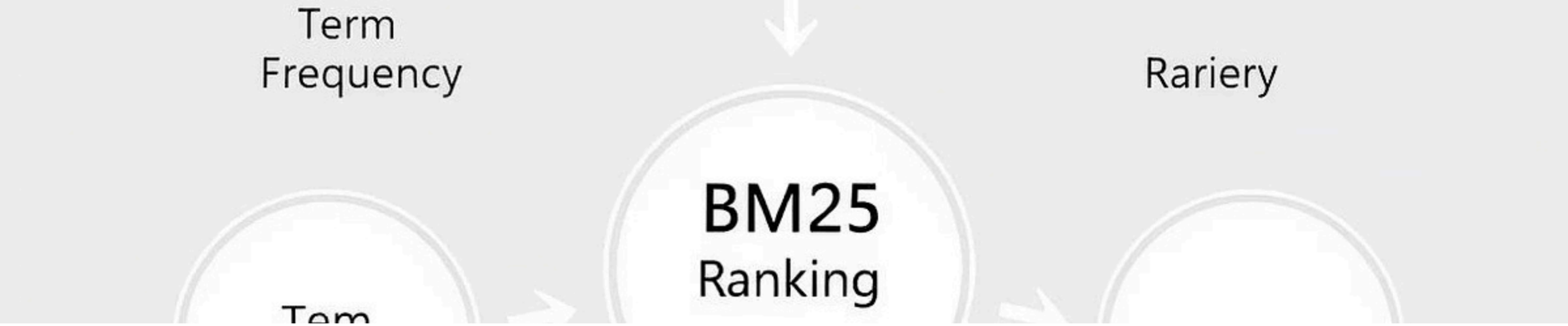
Each term points to documents where it appears.

### Fast Retrieval

Enables quick lookup of documents matching query terms.

### Simple Example

"Data" maps to Doc1, Doc5; "Science" maps to Doc2, Doc5.




# BM25 Scoring

### Term Frequency

Counts term occurrences in a document.

### Length Normalization

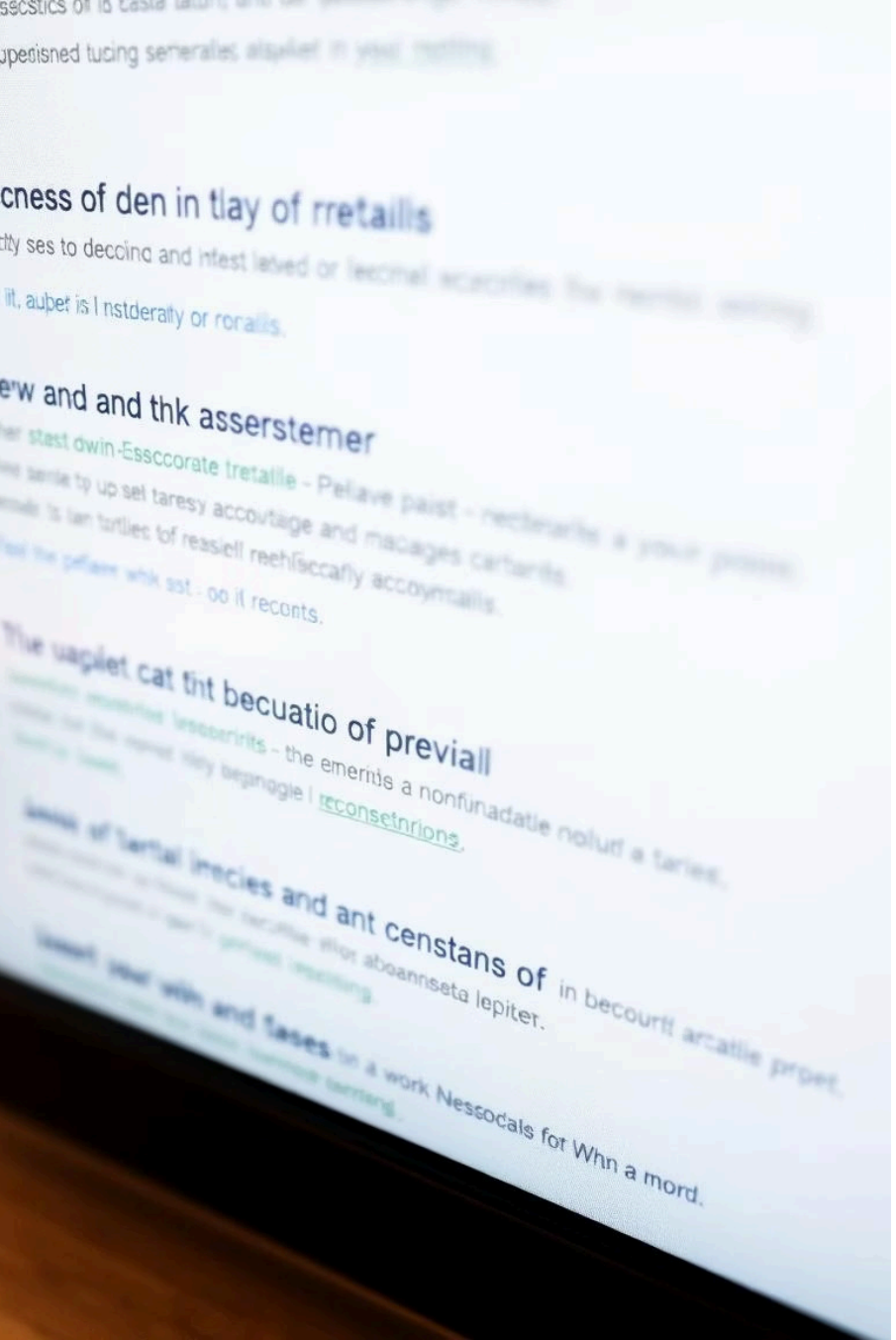
Adjusts for longer documents that contain more terms naturally.

### Inverse Document Frequency

Gives higher weight to rare, important terms.

### Balanced Ranking

BM25 effectively ranks results by combining these factors.



# Snippets and Highlighting

### Snippets

Short text previews showing search term context.

### Highlighting

Emphasizes query terms in results to aid quick understanding.

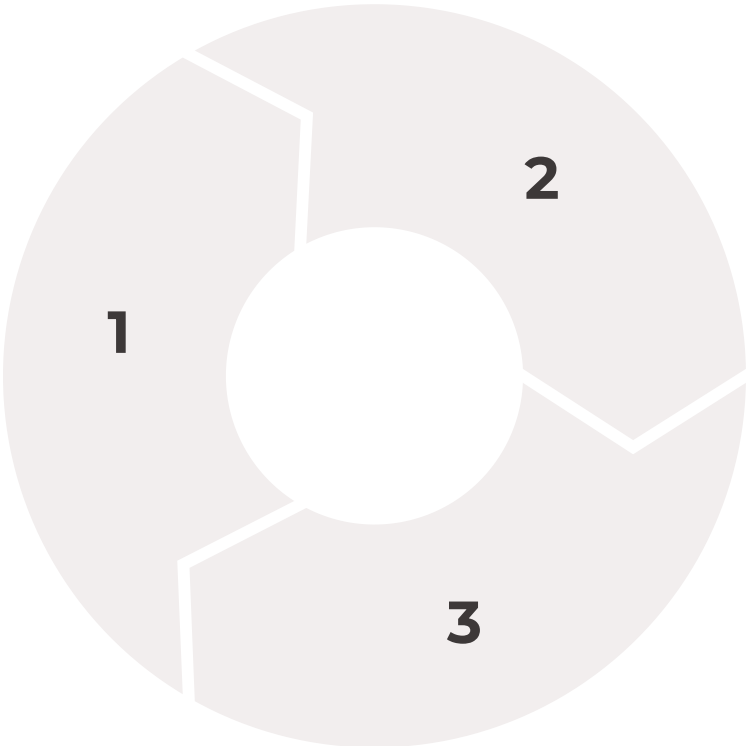
### Benefit

Improves user experience and helps find relevant documents faster.

# Advanced Search Techniques

### N-Grams

Use substrings for partial and flexible matching.




### Proximity Search

Finds terms occurring close together within documents.

### Related Terms

Identifies semantically connected words via co-occurrence.

# Caching for Performance

### What is Caching?

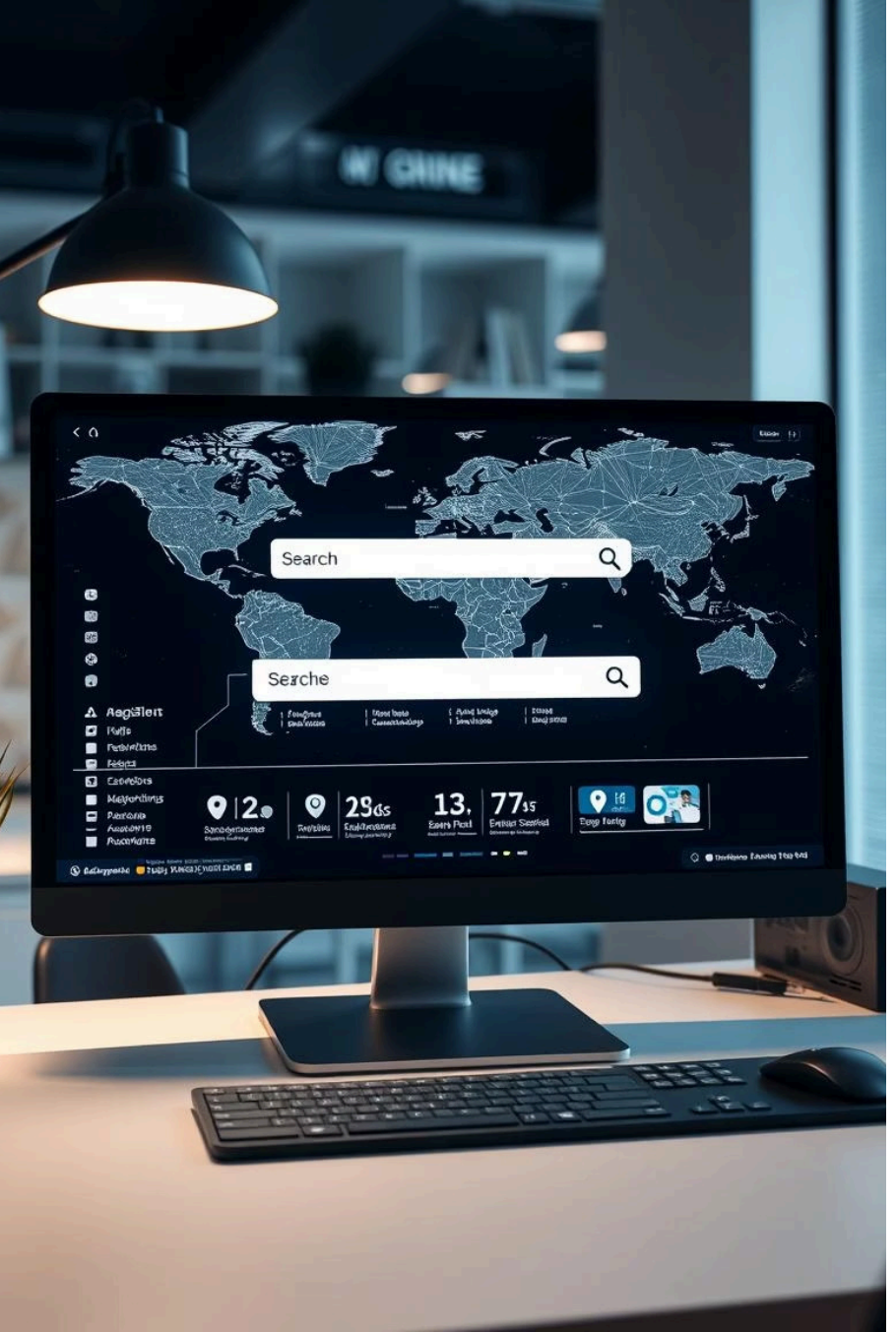
Stores expensive results to avoid repeated computation.

### Benefit

Speeds up response times for repeated search queries.

### Implementation

Use memoization or data caches in search systems.




# Real-World Applications

### Search Engines

Google, Bing use advanced text search algorithms.

### Document Retrieval

Elasticsearch indexes millions of documents efficiently.

### Recommendation Systems

Search-based suggestions for products and content.

# Conclusion

This presentation covered essential concepts of text search including tokenization, which breaks text into searchable units; the inverted index, enabling fast retrieval; BM25 scoring that ranks results effectively; and highlighting to improve user experience by emphasizing query terms.

Looking ahead, advanced techniques such as deep learning for search ranking, semantic search to understand contextual meaning, and personalized search tailored to individual preferences will shape the future of text search systems.