

ThoughtTrace Quick Start Guide



Advanced Search

Below is a list of what syntax is supported by the advanced search in Document Insights.

Terms

By default, each term (or word) in your query is searched separately. So for the query quick brown fox, results will be returned that contain the term "quick" or "brown" or "fox". To match the exact phrase, wrap the terms in double-quotes (i.e.: "quick brown fox")

Wildcards

Wildcard searches can be run on individual terms, using ? to replace a single character, and * to replace zero or more characters.

qu?ck bro*

Field Names

It is possible to limit the search to a specific field by typing the name of the field, followed by a colon, and then the query.

Valid Field Names

- name: Document Name
- originalDocumentName: Original Document Name
- text: Text of the document.

name:(quick brown)
originalDocumentName:"quick brown fox"

Regular Expressions

To use a regular expression, wrap the expression in forward slashes:

`/[0-9]{3}-[0-9]{3}-[0-9]{4}/`
In this example: `/[search criteria]{number of characters}/`

For more regular expressions [Click Here](#).



Fuzziness

We can search for terms that are similar to, but not exactly like our search terms, using the “fuzzy” operator.

This finds all terms with a maximum of two changes, where a change is the insertion, deletion or substitution of a single character, or transposition of two adjacent characters. The default edit distance is 2, but an edit distance of 1 should be sufficient to catch 80% of all human misspellings. It can be specified as: `quikc~1`

`quikc~ brwn~ fokc~`

Proximity Searches

While a phrase query (eg "john smith") expects all of the terms in exactly the same order, a proximity query allows the specified words to be further apart or in a different order. In the same way that fuzzy queries can specify a maximum edit distance for characters in a word, a proximity search allows us to specify a maximum edit distance of words in a phrase:

`"fox quick"~5`

The closer the text in a field is to the original order specified in the query string, the more relevant that document is considered to be. When compared to the above example query, the phrase "quick fox" would be considered more relevant than "quick brown fox".

Boolean Operators

By default, all terms are optional, as long as one term matches. A search for foo bar baz will find any document that contains one or more of foo or bar or baz.

The preferred operators are `+` (this term must be present) and `-` (this term must not be present). All other terms are optional.

For example, this query: `quick brown +fox -news` states that:

fox must be present

news must not be present

quick and brown are optional — their presence increases the relevance

The familiar boolean operators `AND`, `OR` and `NOT` (also written `&&`, `||` and `!`) are also supported but beware that they do not honor the usual precedence rules, so parentheses should be used whenever multiple operators are used together. For instance the previous query could be rewritten as: `((quick AND fox) OR (brown AND fox) OR fox) AND NOT news` This form now replicates the logic from the original query correctly, but the relevance scoring bears little resemblance to the original.



Grouping

Multiple terms or clauses can be grouped together with parentheses, to form sub-queries:

(quick OR brown) AND fox

Reserved Characters

If you need to use any of the characters which function as operators in your query itself (and not as operators), then you should escape them with a leading backslash. For instance, to search for $(1+1)=2$, you would need to write your query as `\(1\+1\) =2`.

The reserved characters are: + - = && || > < ! () { } [] ^ " ~ * ? : \ /

Failing to escape these special characters correctly could lead to a syntax error which prevents your query from running.

NOTE: < and > can't be escaped at all. The only way to prevent them from attempting to create a range query is to remove them from the query string entirely.

