

Expert mode

The image displays the Overton Expert mode search interface, showing two nested search groups. A magnifying glass highlights the 'NEAR' operator in the second group.

Group 1:

- Search query: ("climate chan*":ti,ab OR "global warming":ti,aidesc OR "net zero"~2:ft)
- Operator: Mixed
- Row 1: Title/Snippet | climate chan* | Exact phrase
- Row 2: OR | Title/All description | global warming | Exact phrase
- Row 3: OR | Full text | net zero | Proximity | 2 words
- Buttons: + Add row, Add nested group

Group 2:

- Search query: ((rail NEAR/3 emission*)):ft
- Operator: AND
- Row 1: Full text | rail | NEAR | emission*
- Buttons: + Add row, Add nested group

Sections in this guide

[What is the expert query builder?](#)

[Overton's query syntax](#)

[How to use Overton's expert query builder](#)

[Quick reference](#)

What is the expert query builder?

Overton's expert query builder utilises an Embase-style search syntax to allow you to create advanced searches when looking at our database of policy documents. Our expert builder allows you to visually construct and update your query. This guide will help you understand how our expert search works and how you can use our search builder.

Key considerations when constructing your search query:

- Policy documents don't have abstracts. We sometimes pull descriptive text from webpages for the documents we collect, these are referred to as "snippets". Within Overton, around 8% of documents include a snippet. We recommend searching against the AI generated document description if you are looking to search within a summary of policy documents, as this is available for a much wider range of documents (around 90%).
 - Titles of policy documents are automatically added to our database and may contain errors. Bear this in mind if you restrict your search strategy to titles only.
-

Overton's query syntax

[Prefix](#)

[Basic keyword searching](#)

[Phrase searching](#)

[Truncation](#)

[Wildcards](#)

[Boolean operators](#)

[Combining concepts](#)

[Proximity operators](#)

[Phrase proximity with ~n](#)

[Phrase proximity with NEAR](#)

[Phrase proximity with NEXT](#)

[Combining proximity with truncation and wildcards](#)

[Cautions](#)

[Field selection and validation](#)

[Text field codes](#)

Prefix

Our expert query builder uses a prefix when generating the search term, which tells our system how to interpret the query. This prefix is ``complex:``.

Basic keyword searching

Enter one or more words to find documents containing those terms.

None

`complex:vaccine`

Retrieves documents whose title or description containing *vaccine*.

Unless you specify a field (see [Field searching](#)), terms are matched across the default text fields: the **title** and translated title, the **snippet**, and the **AI-generated description and theme**. Full text is **not** searched by default.

There is **no stemming**. Searching for **nurse** will not automatically match *nurses*, *nursing*, or *nursed*. Use truncation (see [below](#)) to capture word variants.

Phrase searching

Enclose two or more words in double quotation marks to match them as an exact phrase.

None

complex:"climate change"

Retrieves documents containing the phrase *climate change* (the words adjacent and in that order), rather than the two words appearing separately anywhere in the document.

You can truncate inside a phrase:

None

complex:"climate chan*"

Matches *climate change*, *climate changes*, *climate changing*, and so on.

When adding a search term you are able to select from the following match types:

- **Keyword:** Find documents containing these words, however far apart they are
- **Exact phrase:** Match an exact phrase. All words must appear in the order given.
- **Proximity:** Match these words within a set number of words of each other, in the order given. See more on [proximity operators](#).

You can select these using the dropdowns within the builder.

The image shows a search builder interface. On the left, there are three empty input fields labeled 'Search term'. To the right of the first input field is a dropdown menu. The dropdown menu is open, showing three options: 'Keywords' with a 'T' icon, 'Exact phrase' with a '""' icon, and 'Proximity' with a '~' icon. The 'Keywords' option is currently selected.

[The "match type" menu]

Truncation

An asterisk (*) matches any number of characters at the end of a word.

None

complex:vaccin*

Matches *vaccine*, *vaccines*, *vaccinate*, *vaccination*, *vaccinating*, and other words beginning with *vaccin*.

Truncation can be used inside phrases ("climate chan*") and combined with proximity operators.

Wildcards and truncation are **not** supported at the start of a word — for example, *nald is not valid.

Wildcards

A question mark (?) matches exactly one character. This is useful for spelling variants.

None

complex:immuni?ation

Matches both *immunisation* and *immunization*.

None

complex:wom?n

Matches *woman* and *women*.

Use ? for single-character variation and * for variable-length endings; the two can appear in the same query.

Boolean operators

Classic Boolean operators are available within the query builder. **AND**, **OR**, and **NOT** work as expected text search terms. Search terms can be grouped, see [the Groups section](#) for more information on grouping functionality.

⋮	1	Title/Snippet	climate chan*	6699		
⋮	2	OR	Title/Snippet	global warming	T	
⋮	3	AND	Title/Snippet	net zero	~	- 2 words +
⋮	4	NOT	Title/Snippet	ocean*	T	

[An example of search rows using AND / OR / NOT booleans]

None

complex:vaccine AND immunisation

Both terms must be present.

None

complex:vaccine OR vaccination

Either term may be present.

None

complex:vaccine NOT influenza

Documents containing *vaccine* but not *influenza*.

None

complex:(vaccine OR vaccination) AND policy

Either *vaccine* or *vaccination*, combined with *policy*. The parentheses ensure the **OR** is evaluated before the **AND**.

Combining concepts

Build each concept as a bracketed block — synonyms joined with **OR** — then join the concepts together with **AND**. This is the standard building-block approach used in systematic searching.

None

```
complex:("artificial intelligence" OR "machine learning" OR "deep learning") AND (radiology OR "medical imaging")
```

- **Concept 1:** AI-related terms, any of which may appear
- **Concept 2:** imaging-related terms, any of which may appear
- The **AND** requires at least one term from *each* concept

Build Write Examples Remove all terms

1 ("artificial intelligence" OR "machine learning" OR "deep learning") Title/Snippet +1

1 Title/Snippet +1 artificial intelligence 66/99

2 OR Title/Snippet +1 machine learning 66/99

3 OR Title/Snippet +1 deep learning 66/99

+ Add row Add nested group

AND (radiology OR "medical imaging") Title/Snippet +1

1 Title/Snippet +1 radiology T

2 OR Title/Snippet +1 medical imaging 66/99

+ Add row Add nested group

complex: ("artificial intelligence" OR "machine learning" OR "deep learning") AND (radiology OR "medical imaging")

Phrases 4 / 16 Booleans 4 / 64

Search

[An example of combining concepts]

You can attach field codes to a whole block and mix in proximity and truncation freely:

None

```
complex:("net zero"~2 OR "carbon budget") AND (transport OR aviation):ti,ab
```

Retrieves documents about net-zero or carbon-budget topics that also mention transport or aviation **in the title or snippet**.

Proximity operators

Proximity operators find terms that appear near one another, which is far more precise than a plain **AND**. This is especially valuable for searching full-text of policy documents. You are able to use these proximity operators with normal keyword searches and wildcards.

Phrase proximity with ~n

You can also apply proximity *inside* a quoted phrase using a tilde followed by a number. This loosens an exact phrase so the words may appear within *n* words of each other.

None

complex:"net zero"~2

Matches *net zero* where the two words appear within 2 words of one another.

 Build

 Write

Examples ▾

 Remove all terms

 1

Title/Snippet +1 ▾

net zero

 ▾

 2 words 

[The phrase proximity operator]

Cautions:

- Proximity cannot be used with non-text qualifiers, and a Boolean operator cannot sit inside a proximity expression. For example, ("**climate** **AND** **environment**) **NEAR** "net zero" is invalid — the meaning of **AND** inside a **NEAR** is undefined.
- ADJ4**, **N3**, and similar operators from other platforms are **not** supported. If used, they are treated as ordinary search words. Use **NEAR/n** or **NEXT/n** instead.

The screenshot shows a search interface with a query bar containing '2 OR Title/Snippet +1 TMCJ'. Below the query bar, a dropdown menu is open, showing a 'NEAR' operator selected. The dropdown also shows 'within' and '5 words' as options. Below the dropdown, the query 'TMC-Joint*' is visible, followed by a 'T' dropdown menu.

[The NEAR proximity operator]

They work **between search terms only** — you cannot place a proximity operator next to a field qualifier or a Boolean group.

Phrase proximity with NEAR

NEAR matches terms within a set number of words of each other, **in any order**.

None

complex:vaccine NEAR immunisation

Matches *vaccine* and *immunisation* within the default distance (6 words), in either order.

Control the distance with **NEAR/n**:

None

complex:vaccine NEAR/3 immunisation

Matches the two terms within 3 words of each other, in any order. The maximum distance is **12**.

Phrase proximity with NEXT

NEXT works exactly like **NEAR**, but **order matters** — the first term must precede the second.

None

complex:climate NEXT/2 change

Matches *climate* followed by *change* within 2 words.

Combining proximity with truncation and wildcards

Proximity operators work happily alongside `*` and `?`:

None

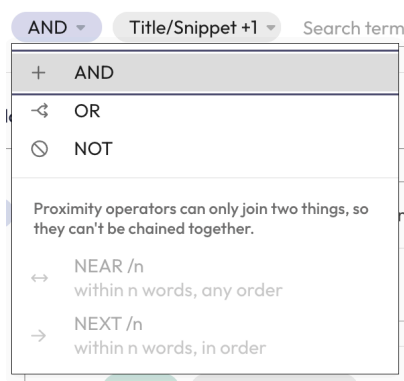
`complex:carpometacarp* NEAR/3 thumb`

None

`complex:vaccin* NEAR immuni?ation`

Cautions

- Proximity cannot be used with non-text fields (e.g. publication dates)
- Boolean operators cannot sit inside a proximity expression when using AND. For example, ("`climate`" AND `environment`) NEAR "`net zero`" is invalid — the meaning of AND inside a NEAR is not supported. However, these can be used when adding proximity expressions when using OR.



[The boolean options menu]

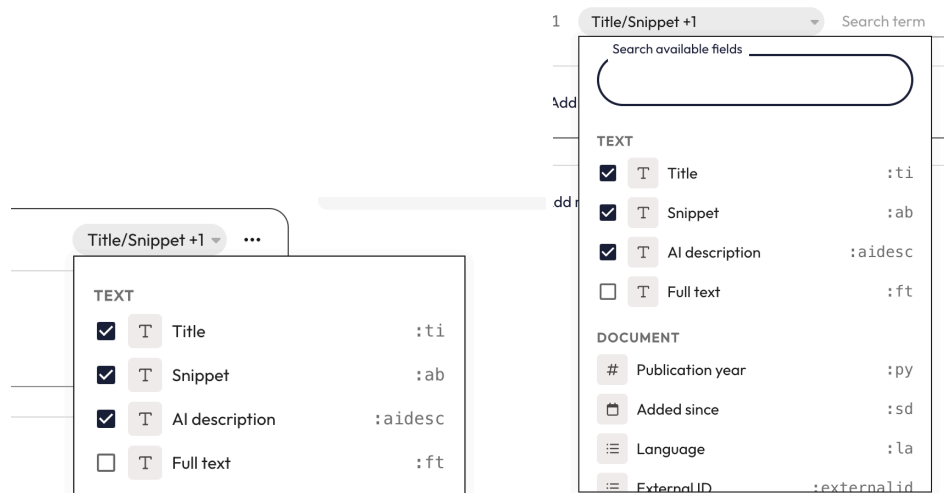
- `ADJ4`, `N3`, and similar operators from other platforms are **not** supported. If used, they are treated as ordinary search words. Use `NEAR/n` or `NEXT/n` instead.

Field selection and validation

You are able to select which fields you would like to search in. This can be done both at the group level and at the row level.

If you would like to apply a search field to an entire group theme you would only be able to do this for text fields.

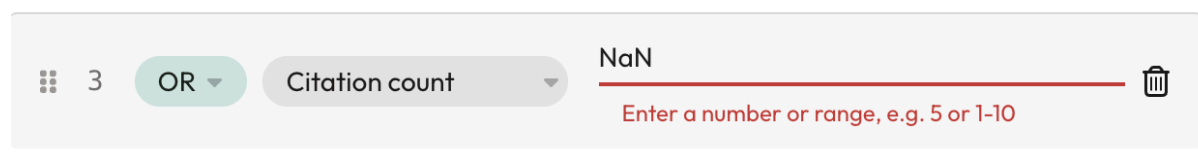
When applying a field selection to a single row, there are additional fields available. You are able to use the search box to find the available fields that can be searched within.



[The fields menu, for groups and for rows]

Some fields will require a specific data type like numbers. For example, publication year.

If incorrect text is inputted, the builder will show an error



[A row erroring due to incorrect input]

Text field codes

Each field code can be searched in a different way, depending on its type:

- **Text** fields allow you to search for individual words or phrases, and the engine will look for documents containing them.
- **Exact** fields require your search term to match the entire value precisely, used for identifiers like external IDs or domain names, where partial matches aren't helpful.
- **Integer** fields hold numeric values and can be searched using exact numbers or ranges.

Code	Type	Field
:ti	text	Document title and translated title
:ab	text	Document snippet (sometimes extracted from landing pages)
:aidesc	text	LLM generated description & LLM generated theme
:ft	text	The full text of the document
:id	exact	Overton assigned policy document ID or PDF ID
:externalid	exact	A site specific identifier assigned to an output, similar to a catalogue number or DOI. We only collect these for certain sources.
:py	integer	Publication year (YYYY). Can be a range e.g. 2010-2020
:sd	date	Added since date. Returns documents since (greater than or equal to) this date (YYYY-MM-DD)
:la	exact	Language of the policy document. Must be ISO693_3 e.g fre, eng, ger

:cy	exact	Country of the policy document source in Overton, see countries on policy sources index page
:sourceid	exact	Policy_source_id of the policy document source
:sourcesector	exact	Sector of the policy document source e.g Public Sector
:sourcetype	exact	Type of the policy document source e.g. Government
:sourcefunction	exact	Function of the policy document source
:sourceauthor	exact	Policy organisation/author.
:citeddomain	text	Find documents that cite specific domains.
:citecount	integer	Number of incoming citations to the policy document, can be a range e.g 1-10
:citecountx	integer	Number of incoming citations (including same source citations) to the policy document, can be a range e.g. 0-5
:citedjournal	exact	Find documents that cite works published in specific journals.
:citedror	exact	ROR that must be cited by policy document results e.g. https://ror.org/01nrxf90

:citeddoi	exact	Find documents that cite works published in specific journals.
-----------	-------	--

Points worth keeping in mind:

- **Full text is not searched by default.** It is included only when you add :ft. Restricting to title and snippet keeps result sets manageable and searches fast; full-text searches with wildcards and proximity are more demanding and can return very large sets.
- **To exclude the AI description** from a default search, restrict to title and abstract with :ti,ab.
- When you are searching in a field that requires an exact match, if searching for any phrase that contains a space then you need to add quotation marks. We will do this automatically if you are using our expert mode builder. For example:
complex: "American Economic Review":citedjournal

How to use Overton's expert query builder

[Opening the expert query builder](#)

[Build mode vs write mode](#)

[Visualising your query](#)

[Example searches](#)

[Building Queries](#)

[Matching text search modes](#)

[Selection fields](#)

[Ordering rows](#)

[Groups](#)

[Actions for groups](#)

[Example search strategies](#)

[Caveats and limits](#)

[Searching limits](#)

[Exact phrases](#)

[Boolean operators](#)

[Proximity searches \(NEAR and NEXT\)](#)

[Some search options cover more than one field](#)

Opening the expert query builder

When viewing our advanced search builder, use the “switch to expert mode” button to switch to the new expert query builder. This will present you with our new builder

 Switch to expert mode

[Expert mode button]

← Advanced search query builder

Search for   Switch to standard mode

 You're using our expert query builder 
It supports multiple fields per phrase/group, wildcards (* & ?) anywhere in a term, and NEAR/NEXT proximity operators between search terms.
Let us know how you get on!

 Build  Write Examples ▾  Remove all terms Your query will appear here as you build it.

 ▾ Group () Title/Snippet +1 ▾ ...

 1 Title/Snippet +1 ▾ Search term T ▾

+ Add row  Add nested group

 Search

[Preview of expert mode query builder]

By default, the document search uses Overton's standard search logic. Only queries that start with **complex:** will use the new Embase search system.

None

complex: your query here

Everything after **complex:** is interpreted using the rules below. A query entered **without** the **complex:** prefix uses standard search and ignores this syntax.

Note: If a complex query is malformed, the search returns **0 results** (rather than a default "everything" view) and will show an error message at the top of the page. If you get no results unexpectedly, check the syntax first.

Build mode vs write mode

The new complex query builder has “build” and “write” modes. Build mode allows you to write your query row by row, whereas write mode allows you to edit the query directly. You can move between both modes.



[Build/write mode toggle button]

Note: If the query in the write box is badly formed, then an error will show and switching to Build mode is blocked.

You can also bring a previously used query from Overton or another platform into our new query builder using “write” mode by copy and pasting the content in. Note, while Overton uses an Embase-style query syntax, not all queries are possible.

If there are any issues parsing your query, this will appear in an error message.

Build Write

Examples

Remove all terms

Query

((('carpometacarpal joint'/de) AND thumb/de) OR 'trapeziometacarpal joint'/de OR Rhizarthrosis/de OR 'wrist joint'/mj OR 'wrist'/mj OR (((carpometacarp* OR carp*-metacarp* OR cmc-joint* OR basilar* OR basal* OR base) NEAR/3 (thumb OR first-carp*)) OR trapeziometacarp* OR trapezi*-metacarp* OR Rhizarthro* OR TMCJ OR TMC-Joint*):ab,ti,kw OR (wrist*):ti) AND (innervation/de OR denervation/de OR 'muscle denervation'/de OR neurotomy/de OR (innerwat* OR denerwat* OR neurotom*):ab,ti,kw) NOT ([Conference Abstract]/lim OR [Conference Review]/lim OR [preprint]/lim OR 'clinical trial':it) NOT ([animals]/lim NOT [humans]/lim)

ⓘ

This query can't be read

- "/" only belongs to NEAR and NEXT here — suffixes like "/exp" or "/lim" are Embase syntax.
- Square brackets aren't part of this query language — limits like "[2015-2024]/py" are Embase syntax.
- Phrases are wrapped in double quotes here, not single ones.
- ":it" isn't a field the builder knows.

[Example of pasted query which cannot be parsed]

Visualising your query

The new query visualiser will appear as you build your search. The query can be copied using the “copy” button that is part of the visualiser.

Each new “group” of terms will appear on a new line, with the operator to the left. Booleans in groups will wrap onto new lines when needed, so that the full query is always visible on screen.

```
complex: ("climate chan*" OR "global
warming" OR "net zero"~2):ti,ab
  AND (transport OR aviation OR (rail
NEAR/3 emission*)):ft
  AND ("United Kingdom":cy OR
"Ireland":cy)
  AND 2015-2024:py
  NOT "press release":ti
```

Phrases	5 / 16	Booleans	9 / 64

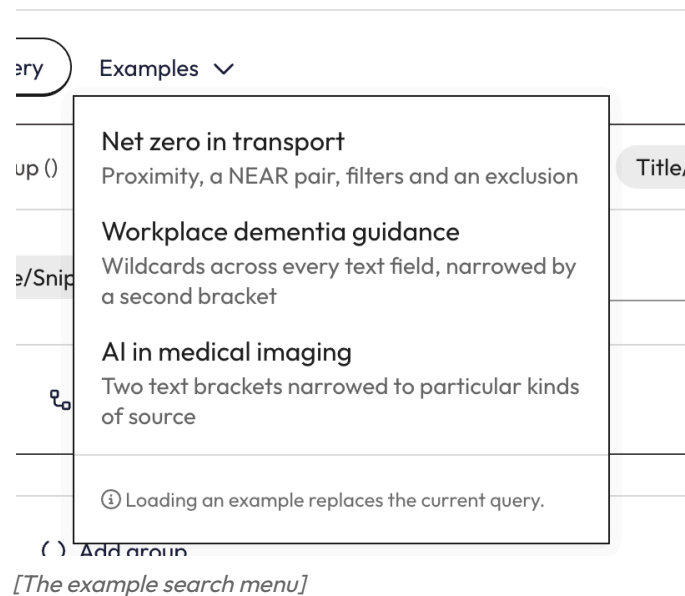
[Example of query preview]

Limitations for phrases and booleans are shown at the bottom of the query visualiser.

Note. To manually edit a query after constructing it in the builder, switch to “write” mode

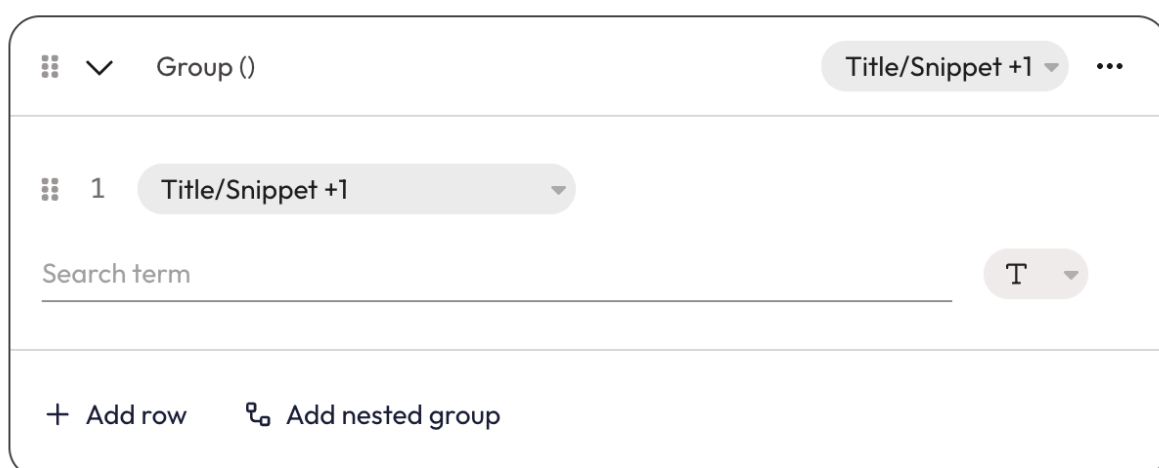
Example searches

If you would like to see examples of how to use the complex query builder, we have included different searches that you can view to see how the builder can be used.



Building Queries

The query builder will always begin with a “group” by default, but you are able to remove this and start with a row if you prefer.



[Preview of the default “empty” search, showing a single group]

Matching text search modes

The text matching search modes that are available in our advanced search builder are still available in our expert search mode.

Keywords

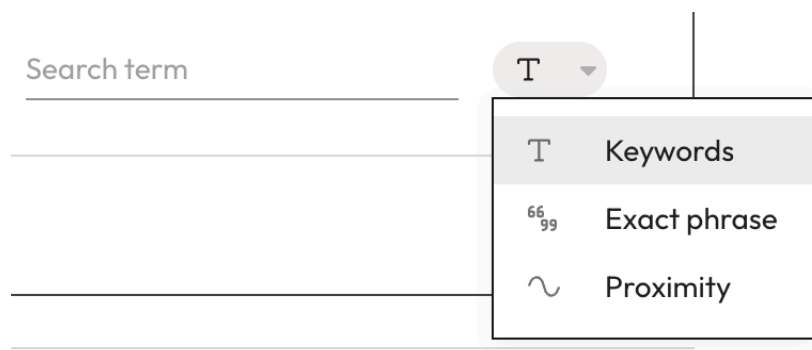
Find documents containing these words, however far apart they are

Exact phrase

Match an exact phrase. All words must appear in the order given.

Proximity

Match these words within a set number of words of each other, in the order given.



[The “match type” menu]

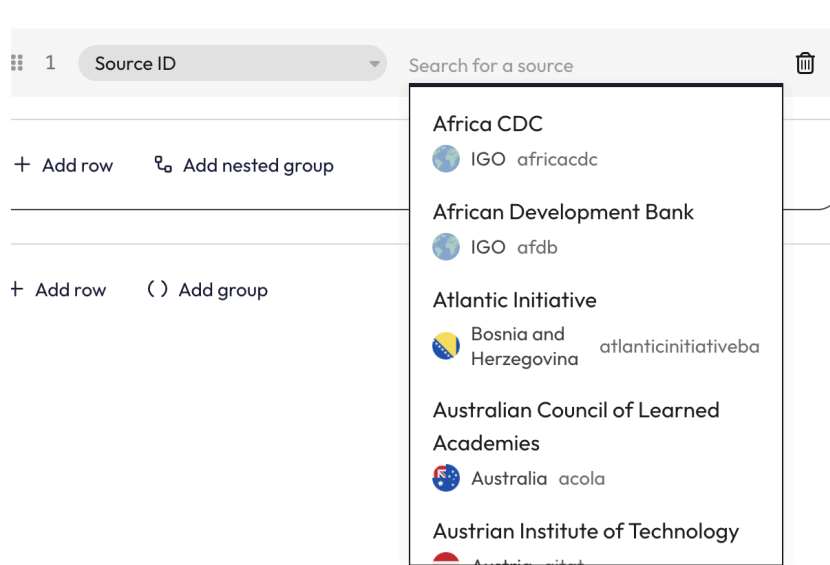
Note: Truncate or stand in for characters anywhere in a term: * for any number, ? for one.

None

complex: "type 2 diabet*" AND "randomi?ed trial"

Selection fields

Some fields, e.g. source ID, can only handle exact matches. For these, we will provide a dropdown menu where you are able to choose from the available options.



[Example of a specific field type dropdown, showing the sources available to populate the “source ID” field]

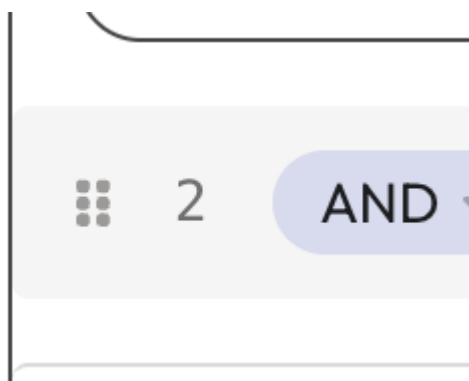
Note: All potential options will appear in this list but we may not have relevant results related to the option that you have selected.

Ordering rows

All terms and groups can be ordered or reordered at your convenience. Each row will also appear with a count so you are able to easily track your queries. Counts will appear and update automatically as they are reordered.

Counts appear and restarts with each new group added.

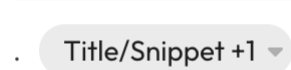
Each group and term can be dragged/dropped into a new order using the handle.



[The drag handler button]

Groups

Groups allow you to group related parts of your query together. Groups can have a single or multiple search fields applied to every row within the group. By default, each group will have the following search fields applied: title, snippet and AI description.



[The pill/chip which activates the field type menu for a group]

However, you are also able to apply a selection of fields directly for each row within a group. If you choose to set these on a row level this will disable the “group” level field selection. Any non-updated terms will inherit the group’s previous default field selection, and the group field will update to display “mixed”.

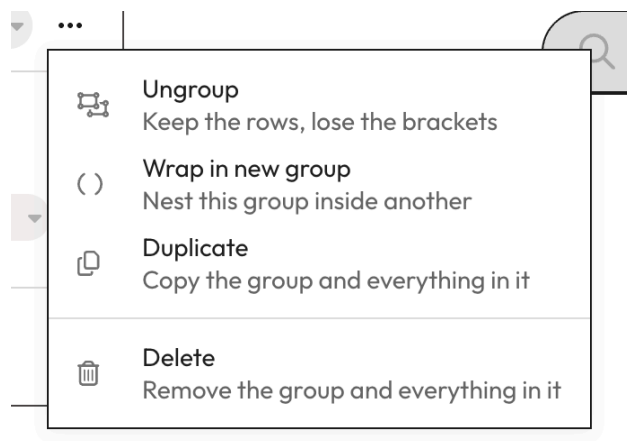


[The pill/chip which activates the field type menu for a group, showing that the groups rows are searching a mix of fields]

Actions for groups

There are additional actions that you can make to your groups.

- **Ungroup:** Allows you to separate a group into each of individual row
- **Wrap in new group:** Allows you to nest a group inside a new group, to easily add comparisons between groups
- **Duplicate:** Will add an identical new group
- **Delete:** Will delete a group and all the terms within it



[The group actions menu]

Example search strategies

1. AI applications in medical imaging

None

```
complex:("artificial intelligence" OR "machine learning" OR "deep learning") AND  
(radiology OR "medical imaging")
```

Documents combining an AI concept with an imaging concept.

2. Climate and net zero, as adjacent ideas

None

```
complex:("climate chan*") NEAR "net zero"
```

Documents where a *climate change* variant appears within 6 words of the phrase *net zero* — capturing discussions that link the two, rather than mentioning them in unrelated passages.

3. Net zero in transport, restricted to title and snippet

None

```
complex:("net zero"~2 OR "carbon budget") AND (transport OR aviation):ti,ab
```

A focused, fast search that keeps results to documents where the relevant terms appear prominently.

4. A multi-synonym concept block

None

```
complex:("workplace dement*" OR "workforce dement*" OR "occupation* dement*")  
AND (guide OR policy OR framework OR toolkit OR recommendation)
```

Combines several truncated synonyms for workplace dementia with a set of document-type cue words — demonstrating how a large synonym list scales using **OR** within a concept and **AND** across concepts.

Caveats and limits

A few behaviours and limits that affect how you build queries:

- **The **complex:** prefix is required** to use any of this syntax.
- **No stemming** — use truncation (**nurs***) to capture word forms.
- **No leading wildcards** — ***** and **?** cannot begin a word.
- **ADJn / Nn operators are not supported** — use **NEAR/n** or **NEXT/n**.
- **Maximum proximity distance is 12 words.**
- **A malformed query returns 0 results** — check syntax if a search comes back empty unexpectedly.
- At this time, we cannot support complex queries in **Chinese, Japanese and Thai**.

Searching limits

Exact phrases

Each exact phrase, meaning text inside quotation marks, can be up to 16 words long. This limit applies only to exact phrase searching. Keywords outside quotation marks are not affected.

Boolean operators

Each group (or concept) in your search can contain up to 64 Boolean operators (AND, OR, NOT). The limit applies to each group separately, so when you add a new group, this group's count starts again from zero.

For example:

None

Group 1 (population): pupils OR students OR "young people" OR adolescents OR teenagers OR children...

None

Group 2 (topic): "mental health" OR wellbeing OR anxiety OR depression OR resilience...

Group 1 can use up to 64 operators, and Group 2 can also use up to 64. If you need more synonyms than one group allows, you can split them into two groups for the same concept.

Proximity searches (NEAR and NEXT)

NEAR and NEXT let you find terms that appear close to each other. To do this, the search compares every term within the proximity operators with every term within the group/concept and repeats that comparison in each field it searches (for example, title, snippet and full text). There's a limit of 256 of these comparisons per search.

You can work out how many your search uses like this:

terms added to group × terms added to proximity operator × number of fields searched

Some search options cover more than one field

Some field tags search more than one field behind the scenes, so each pair of terms counts twice:

Field tag	What it searches	Counts as
Title (:ti)	Title and translated title	2 fields
AI description (:aidesc)	Document description and document theme	2 fields

Here is an example of a search using these fields:

overton

None

```
complex:(climate OR warming OR emissions OR carbon):ti,aidesc  
NEAR/10 (health OR wellbeing OR mortality OR morbidity):ti,aidesc
```

4 terms $\times$ 4 terms = 16 pairs. Title searches 2 fields and AI description searches 2 fields, so that's $16 \times 4 = 64$ comparisons.

Because the number multiplies, it can grow quickly. If you reach the limit, you can:

- **Search fewer fields.** For example, searching title only instead of title and AI description halves the count.
 - **Use fewer synonyms in the NEAR/NEXT part.** Keep the broader list of synonyms in a separate group combined with AND.
-

Quick reference

Technique	Syntax	Example	Retrieves
Activate advanced search	complex:	complex:vaccine	Enables this syntax
AND	a AND b	complex:vaccine AND immunisation	Both terms present
OR	a OR b	complex:vaccine OR vaccination	Either term present
NOT	a NOT b	complex:vaccine NOT influenza	First term, excluding second
Grouping	(...)	complex:(a OR b) AND c	Controls precedence
Phrase	" ... "	complex:"climate change"	Exact phrase
Truncation	*	complex:vaccin*	Any number of trailing characters
Wildcard	?	complex:immuni?ation	Exactly one character
Proximity, unordered	NEAR / NEAR/n	complex:vaccine NEAR/3 immunisation	Terms within <i>n</i> words, any order (default 6, max 12)

Proximity, ordered	NEXT / NEXT/ <i>n</i>	complex:climate NEXT/2 change	Terms within <i>n</i> words, in order
Phrase proximity	"..."~ <i>n</i>	complex:"net zero"~2	Phrase words within <i>n</i> words
Field — title	:ti	complex:diabetes:ti	Title and translated title
Field — snippet	:ab	complex:diabetes:ab	Document snippet
Field — AI description	:aidesc	complex:diabetes:aidesc	AI description and theme
Field — full text	:ft (or :kw)	complex:diabetes:ft	Full document text
Multiple fields	:x,y	complex:diabetes:ti,ab	Any of the listed fields