

Using Elsevier Text Mining API to Retrieve KeyFacts Information

This guide describes how to use the Elsevier Text Mining API to retrieve KeyFacts information for a search result set. If the search results set is comprised of only one document then the KeyFacts information will mimic the Indexed Taxonomy Terms shown for a document on the detailed document view page.

Table of Contents

Using Elsevier Text Mining API to Retrieve KeyFacts Information.....	1
Revision History	1
General Workflow	2
Additional Information.....	3
How to use the Export KeyFacts endpoint in Swagger UI	3
Helpful Hint – Using the API code link information from the User Interface (Export Wizard)	5
Helpful Hint –Translating the API code link information from the User Interface (Export Wizard) for use in Swagger UI (Decode the list of columnIds)	7
Helpful Hint –Create KeyFacts API call templates to easily replicate desired reports	8
How to use the Export Task/TaskID endpoint in Swagger UI	9
How to use the Export Task/TaskID/Result endpoint in Swagger UI	10

Revision History

Name	Date	Reason For Changes	Version
George Jiang	September 27, 2017	v1.0 of document	1.0

General Workflow

A pipeline consisting of several API requests are needed to retrieve the KeyFacts data from the system.

1. Use Export KeyFacts endpoint

GET

</export/keyfacts>

Start 'Relevant Terms' Export

This endpoint will begin an export job and return the export task ID for that job once a user has provided all the information i.e. search query (query), searched index (searchTarget), search mode (mode), relevant term category(ies) (categoryTitle), desired data (columnIds), export format (exportFormat), etc.

2. Use Export Task/TaskID endpoint

GET

</export/task/{taskId}>

Export Task Status

This returns the status information for a specific export task ID. The export task ID from the location field of the KeyFacts endpoint response header should be used here.

Please note that some may consider this step optional but it is part of best practices to have this step as part of the pipeline since some export jobs may result in large export processes and the subsequent step will fail if the system has not completed the export task.

3. Use Export Task/TaskID/Result endpoint

GET

</export/task/{taskId}/result>

Retrieve Export Result

This returns the actual export report link for a specific export task ID in the Request URL box.

The export task ID from the location field of the KeyFacts endpoint response header should be used here.

4. Retrieve exported data using Request URL link from Export Task/TaskID/Result

Using the Request URL information will retrieve the exported report.

Additional Information

How to use the Export KeyFacts endpoint in Swagger UI

All applicable fields should be populated and options selected. Tips on populating selected fields are detailed below.

GET
/export/keyfacts
Start 'Relevant Terms' Export

Implementation Notes

Relevant Terms Export outputs most frequent terms accumulated from the most relevant documents as a table that may include document attributes, relevant text with highlighting and taxonomy term annotations (GenBank ID, CS numbers etc.)

Parameters

Parameter	Value	Description	Parameter Type	Data Type
insttoken	<input type="text"/>		query	string
categoryTitle	<div>Diseases</div>	Name of Category	query	Array[string]
query	<input type="text" value="asthma ; drugs"/>	Search query	query	string
sort	<div>r (default)</div>	Sorting. 'r' - by Relevance 'd' - by Date, descending - Newest First 'a' - by date, ascending - Oldest First.	query	string
searchTarget	<div>full_index</div>	Qualified index name.	query	string
mode	<div>BasicSearch</div>	Search Mode.	query	string
chooseTerms	<div>all (default)</div>	Type of terms	query	string
exportFormat	<div>xlsx (default)</div>	Output format	query	string
chooseDocumentCount	<div>1 (default)</div>	Number of documents per relevant term	query	string
columnIds	<div>["0","composited","doi","pmid","pmcid","rePo</div>	JSON array of ids for the columns to be included in the specified order	query	string
so_a	<input type="text"/>	Filter by Author(s)	query	string
so_j	<input type="text"/>	Filter by Journal(s)	query	string
so_d	<input type="text"/>	Filter by date. Examples:	query	string

- categoryTitle field

Populate this field with the taxonomy major category branch name(s) that you want to get data for. To get data from multiple categories, use a new line for each branch name. The major branch name options for the standard GeneTree taxonomy are listed below:

- Genes, Proteins, and RNA

- Small Molecules
- Diseases
- Biological Functions
- Organisms
- Anatomical Concepts
- Cell Lines
- Cell Components
- External Factors
- Clinical Parameters
- Medical and Research Procedures
- Search Enhancement Terms
- Measurement Values
- **columnIds field**

Populate this field with a decoded string that provides a JSON array of ids for the desired columns and specified order for the exported report. This field is expecting the data to be in the following format - **[“0”, “x”, “y”, “z”]** - where x, y, z are string values from the table below.

The table below lists and describes all possible value options for columnIds field:

String Value	Category	Description
compositeld	document/bibliographic	Elsevier Text Mining unique identifier that is a compilation of the various source document identifiers
doi	document/bibliographic	Digital object identifier
pmid	document/bibliographic	PubMed document identifier
pmcid	document/bibliographic	PubMed Central document identifier
rePorter	document/bibliographic	NIH RePORTER identifier
pii	document/bibliographic	Publisher item identifier
openUrl	document/bibliographic	OpenURL link for a document
localDocs	document/bibliographic	Local document identifier
snippet	document/bibliographic	Relevant text snippet
reference	document/bibliographic	Reference information – consolidated citation information; Elsevier Text Mining standard
title	document/bibliographic	Title information of a document
author	document/bibliographic	Author information of a document
institution	document/bibliographic	Institution information of a document
journal	document/bibliographic	Journal information of a document
date	document/bibliographic	Date information of a document
year	document/bibliographic	Year information of a document
abstractText	document/bibliographic	Document abstract
keyword	document/bibliographic	Keyword information of a document
meSh	document/bibliographic	MeSH information of a document
chemical	document/bibliographic	Chemical information of a document
URN	Diseases	
MeSH Heading	Diseases	

Helpful Hint – Using the API code link information from the User Interface (Export Wizard)

The user interface export wizard can be helpful in setting up the desired report - selecting categories, annotations, document metadata, and the ordering of items in the report - and quickly generating a link to the specific API call.

The screenshot displays the Elsevier Text Mining API user interface. At the top, there's a navigation bar with 'Text Mining' and tabs for 'Basic', 'Advanced', 'Multiple', 'Taxonomy', and 'Tools'. Below this, the 'Basic Search' section shows a search for 'asthma: drugs'. A 'Relevant Terms Export' wizard is open, showing a list of relevant terms and a modal dialog box. The dialog box contains the following text:

Your data is available via API

To access the data, select 'Copy API Code' button or select the string below, copy and paste in your app.

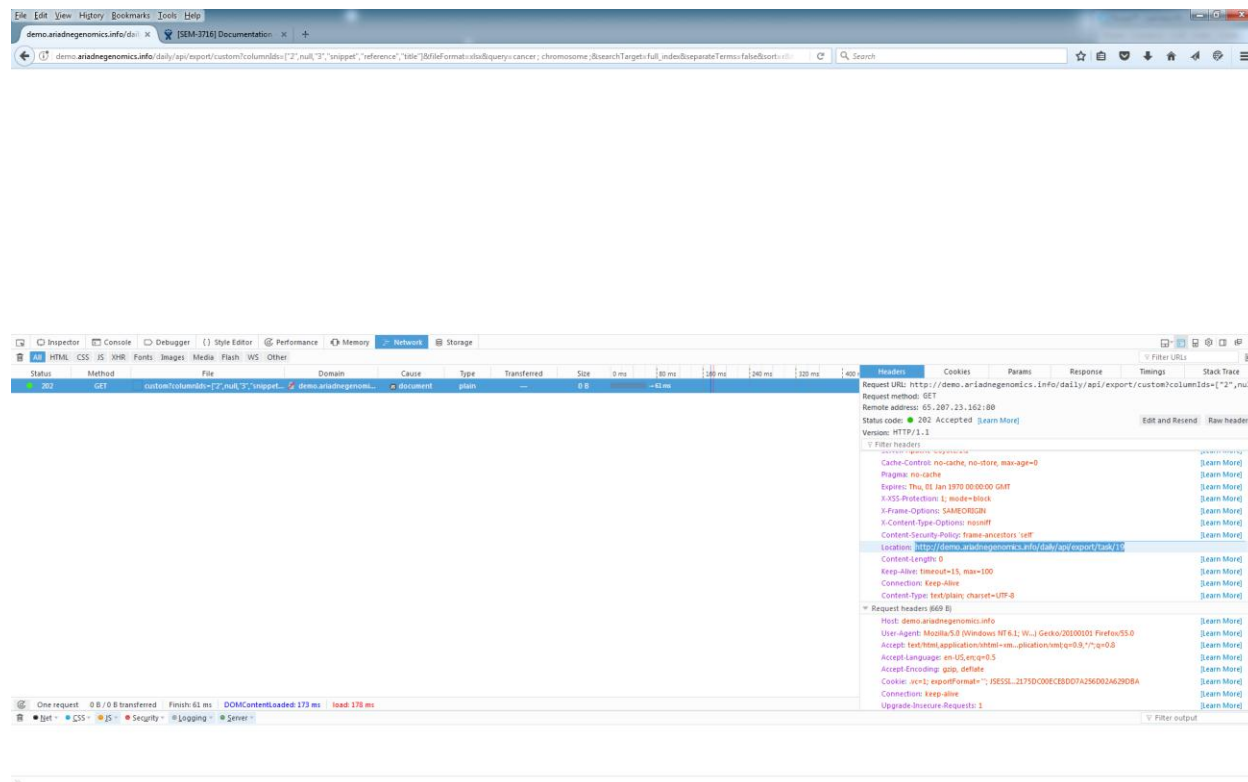
http://dev.elseviertextmining.com/daily/api/export/keyfacts?categoryTitle=Diseases&chooseDocumentCount=1&chooseTerms=all&columnId=5B%20%22,%22composited%22,%22doi%22,%22pmid%22,%22pmcid%22,%22rePorter%22,%22pii%22,%22openUrl%22,%22localDocs%22,%22snippet%22,%22reference%22,%22title%22,%22author%22,%22institution%22,%22journal%22,%22date%22,%22year%22,%22abstractText%22,%22keyword%22,%22mesh%22,%22chemical%22,%22MeSH%20Heading%22,%22URN%22%5D&exportFormat=xlsx&query=asthma;%20drugs&searchTarget=full_index&sort=r&apikey=YOUR_KEY&instoken=YOUR_TOKEN

Buttons: Copy API Code, Cancel

The background shows a list of relevant terms with columns for 'Column Order' and 'Column'. The terms include 'eye disease', 'adverse drug re...', 'allergic contact ...', 'arthritis', 'autoimmune di...', 'Biological Functions', 'Organisms', and 'Anatomical Concepts'. The URL at the bottom of the page is:

dev.elseviertextmining.com/daily/search?searchTarget=full_index&mode=BasicSearch&sort=r&query=asthma;%20drugs&so_a=8&so_j=8&so_d=8&so_s=8

Using this API code within a browser will appear to return an empty page but that is because the needed information for subsequent API calls (task ID number) is returned in the response header.



Please note that the API code from the User Interface (Export Wizard) button is equivalent to what one would get in the Request URL box in Export KeyFacts Swagger UI endpoint.

[Try it out!](#)
[Hide Response](#)

Curl

```
curl -X GET --header 'Accept: text/plain' 'http://dev.elseviertextmining.com/daily/api/export/keyfacts?categoryTitle=Diseases&quer
```

Request URL

```
http://dev.elseviertextmining.com/daily/api/export/keyfacts?categoryTitle=Diseases&query=asthma%20%3B%20drugs&sort=r&searchTarget=
```

Response Body

```
no content
```

Response Code

```
202
```

Response Headers

```
{
  "pragma": "no-cache",
  "date": "Tue, 26 Sep 2017 15:06:25 GMT",
  "x-content-type-options": "nosniff",
  "server": "Apache-Coyote/1.1",
  "x-frame-options": "SAMEORIGIN",
  "content-type": "text/plain; charset=UTF-8",
  "location": "http://dev.elseviertextmining.com/daily/api/export/task/26",
  "cache-control": "no-cache, no-store, max-age=0",
  "content-security-policy": "frame-ancestors 'self'",
  "connection": "Keep-Alive",
  "keep-alive": "timeout=15, max=100",
  "content-length": "0",
  "x-xss-protection": "1; mode=block",
  "expires": "Thu, 01 Jan 1970 00:00:00 GMT"
}
```

Helpful Hint –Translating the API code link information from the User Interface (Export Wizard) for use in Swagger UI (Decode the list of columnIds)

The API code from the Relevant Terms export (API link in the export wizard user interface) to the specific API call contains coded information for the columnIds parameter. To get the JSON array of ids for the desired columns and specified order for the exported report to use in Swagger UI, you must first decode the columnIds values:

1. Copy values from columnIds parameter within API code
2. Access a URL decoder e.g. <https://www.urldecoder.org/>
3. Paste information from API code, and decode
4. Copy decoded JSON array information
5. Paste into columnIds field in the Export Keyfacts Swagger UI endpoint

URL
Decode and Encode

Have to deal with URL encoded format? Then this site is made for you! Use the super simple online form below to decode or encode your data. If you're interested about the inner workings of URL encoding, just read the detailed description at the bottom of the page. Welcome!

Decode Encode Other tools

Decode from URL encoded format
Simply use the form below

`%5B%22%22,%22compositeld%22,%22doi%22,%22pmid%22,%22pmcid%22,%22rePorter%22,%22pii%22,%22openUrl%22,%22localDocs%22,%22snippet%22,%22reference%22,%22title%22,%22author%22,%22institution%22,%22journal%22,%22date%22,%22year%22,%22abstractText%22,%22keyword%22,%22meSh%22,%22chemical%22,%22MeSH%20Heading%22,%22URN%22%5D`

Decode **Live mode OFF** Decodes while you type or paste.
UPLOAD FILE Decodes an entire file (max. 10MB).

NADEX **Nadex Binary Options** [Learn More](#)

`["0","compositeld","doi","pmid","pmcid","rePorter","pii","openUrl","localDocs","snippet","reference","title","author","institution","journal","date","year","abstractText","keyword","meSh","chemical","MeSH Heading","URN"]`

Untitled - Notepad
File Edit Format View Help
`http://dev.elseviertextmining.com/daily/api/export/keyfacts?categoryTitle=Diseases&chooseDocumentCount=1&chooseTerms=all&columnIds=%5B%22%22,%22compositeld%22,%22doi%22,%22pmid%22,%22pmcid%22,%22rePorter%22,%22pii%22,%22openUrl%22,%22localDocs%22,%22snippet%22,%22reference%22,%22title%22,%22author%22,%22institution%22,%22journal%22,%22date%22,%22year%22,%22abstractText%22,%22keyword%22,%22meSh%22,%22chemical%22,%22MeSH%20Heading%22,%22URN%22%5D&exportFormat=xlsx&query=asthma;%20drugs&searchTarget=full_index&sort=r&apikey=YOUR_KEY&insttoken=YOUR_TOKEN`

Come back for great savings
Budget

Example of URL decoding transformation:

Information from API link	Decoded information as JSON array
<code>%5B%22%22,%22compositeld%22,%22doi%22,%22pmid%22,%22pmcid%22,%22rePorter%22,%22pii%22,%22openUrl%22,%22localDocs%22,%22snippet%22,%22reference%22,%22title%22,%22author%22,%22institution%22,%22journal%22,%22date%22,%22year%22,%22abstractText%22,%22keyword%22,%22meSh%22,%22chemical%22,%22MeSH%20Heading%22,%22URN%22%5D</code>	<code>["0","compositeld","doi","pmid","pmcid","rePorter","pii","openUrl","localDocs","snippet","reference","title","author","institution","journal","date","year","abstractText","keyword","meSh","chemical","MeSH Heading","URN"]</code>

Helpful Hint –Create KeyFacts API call templates to easily replicate desired reports

To simplify recreation of similar reports, set up the optimal report structure and then you can reuse many parts of the API code (e.g. categoryTitle and columnIds fields). Recall that the columnIds field accepts an ordered list so if you do keep the order the same each time, you can have your system easily parse through reports of the same structure each time. For example, one can set up a template for each main taxonomy branch category to simplify getting such data as individual reports, otherwise querying for multiple categories will return a report with multiple sheets.

How to use the Export Task/TaskID endpoint in Swagger UI

This API endpoint returns the status information for a specific export task ID. The response body will indicate the type of export for a particular task ID, and its status.

Try it out! [Hide Response](#)

Curl

```
curl -X GET --header 'Accept: application/json' 'http://dev.elseviertextmining.com/daily/api/export/task/26?apikey=apoptosis1'
```

Request URL

```
http://dev.elseviertextmining.com/daily/api/export/task/26?apikey=apoptosis1
```

Response Body

```
{
  "label": "Relevant Terms Export",
  "description": "",
  "taskId": 26,
  "state": "COMPLETED",
  "percent": -1
}
```

Response Code

```
200
```

Response Headers

```
{
  "pragma": "no-cache",
  "date": "Tue, 26 Sep 2017 15:07:34 GMT",
  "content-encoding": "gzip",
  "x-content-type-options": "nosniff",
  "server": "Apache-Coyote/1.1",
  "x-frame-options": "SAMEORIGIN",
  "content-type": "application/json",
  "cache-control": "no-cache, no-store, max-age=0",
  "transfer-encoding": "chunked",
  "content-security-policy": "frame-ancestors 'self'",
  "connection": "Keep-Alive",
  "vary": "Accept-Encoding",
  "x-xss-protection": "1; mode=block",
  "keep-alive": "timeout=15, max=100",
  "expires": "Thu, 01 Jan 1970 00:00:00 GMT"
}
```

How to use the Export Task/TaskID/Result endpoint in Swagger UI

This API endpoint returns the actual export report link for a specific export task ID. Use the Request URL information to retrieve the exported report e.g. copy and paste the URL into a browser window; this will begin the download of the report and open the report in Microsoft Excel (depending on the browser settings).

Curl

```
curl -X GET --header 'Accept: application/vnd.openxmlformats-officedocument.spreadsheetml.sheet' 'http://dev.elseviertextmining.co
```

Request URL

```
http://dev.elseviertextmining.com/daily/api/export/task/26/result?apikey=apoptosis1
```

Response Body

[Download result?apikey=apoptosis1](#)

Response Code

```
200
```

Response Headers

```
{
  "pragma": "no-cache",
  "date": "Tue, 26 Sep 2017 15:08:04 GMT",
  "x-content-type-options": "nosniff",
  "server": "Apache-Coyote/1.1",
  "x-frame-options": "SAMEORIGIN",
  "content-type": "application/vnd.openxmlformats-officedocument.spreadsheetml.sheet",
  "content-disposition": "attachment; filename=relevant_terms.xlsx",
  "cache-control": "no-cache, no-store, max-age=0",
  "content-security-policy": "frame-ancestors 'self'",
  "connection": "Keep-Alive",
  "keep-alive": "timeout=15, max=100",
  "content-length": "241541",
  "x-xss-protection": "1; mode=block",
  "expires": "Thu, 01 Jan 1970 00:00:00 GMT"
}
```