Using AmCAT from R

Wouter van Atteveldt

May 25, 2016

This handout describes how to connect to AmCAT from R to conduct queries, download metadata, and upload articles and create new article sets. You will need an account on a working AmCAT server, you can create a free account at https://amcat.nl or install your own server, see https://github.com/amcat/amcat.

You will also need to install the `amcatr` package from github, for which you will need the devtools package:

```
install.packages("devtools")
devtools::install_github("amcat/amcat-r")
```

Connecting to AmCAT

On every computer you need to save your AmCAT password once::

```
library(amcatr)
amcat.save.password("https://amcat.nl", "your_username", "your_password")
```

Next, you can connect using the `amcat.connect` function, storing the connection details in an object `conn`. The token that this creates is valid for 24 hours, so you need to run this command every session:

```
conn = amcat.connect("https://amcat.nl")
```

Retrieving article (meta)data

The `amcat.getarticlemeta` command allows you to retrieve the metadata from an article set. Using the columns keyword, you can specify which columns to select, e.g. headline, medium, and author.

```
meta = amcat.getarticlemeta(conn, 41, 29454, dateparts = T, columns=c("medium", "date"))
head(meta)
```

id	medium	date	year	month	week
21816385	The Guardian	2010-07-01	2010-01-01	2010-07-01	2010-06-28
21804138	The Mirror	2009-10-23	2009-01-01	2009-10-01	2009-10-19
21796025	The Guardian	2009-11-26	2009-01-01	2009-11-01	2009-11-23
21851348	The Guardian	2010-03-06	2010-01-01	2010-03-01	2010-03-01
21830885	The Guardian	2011-09-20	2011-01-01	2011-09-01	2011-09-19
21834988	The Times	2008-06-19	2008-01-01	2008-06-01	2008-06-16

Querying AmCAT

You can use the `amcat.hits` function to run a keyword query which returns the number of hits per document for a query:

```
h = amcat.hits(conn, c("mortgage*", "greek* OR greece*"), labels=c("mortgage", "greece"), sets=29454)
head(h)
```

count	id	query
1	21794967	mortgage
1	21795537	mortgage
1	21795699	mortgage
1	21796592	mortgage
1	21798565	mortgage
1	21798673	mortgage

You can supply multiple queries, and can also supply labels for the queries with the labels argument.

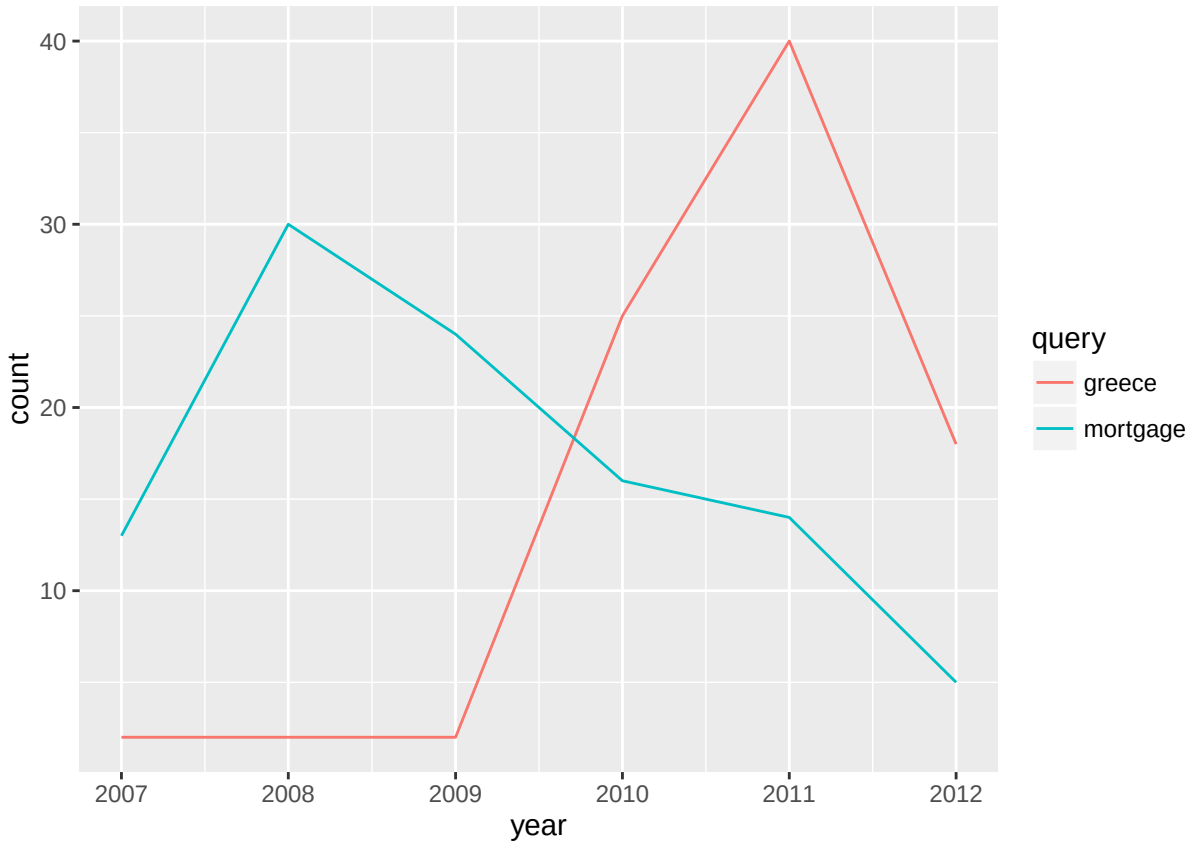
While the hits command returns a row per document per query, you can also get aggregate data directly with the `amcat.aggregate` command, specifying the aggregation axes such as medium, date, and keyword:

```
t = amcat.aggregate(conn, "mortgage*", sets=29454, axis1="year", axis2="medium")
head(t)
```

count	medium	year	query
1	The Times	2007-06-01	mortgage*
2	The Times	2007-09-01	mortgage*
3	The Times	2007-10-01	mortgage*
1	The Times	2007-12-01	mortgage*
1	The Times	2008-02-01	mortgage*
1	The Times	2008-03-01	mortgage*

We can also do the aggregation within R by merging the hits data with the meta data and using the `aggregate` command. For example, this code will create a plot of hits per newspaper per week:

```
h = merge(meta, h)
perweek = aggregate(h["count"], h[c("year", "query")], sum)
library(ggplot2)
ggplot(perweek, aes(x=year, y=count, color=query)) + geom_line()
```



Uploading articles

It is also possible to upload articles from R using the `amcat.upload.articles` command. This allows you to e.g. upload articles form a csv file or folder, or retrieve articles from an API and upload them.

```
d = data.frame(headline=c("A test", "Another test"), date=as.Date(c("2001-01-01", "2008-12-31")))
setid = amcat.upload.articles(conn, project=1, articleset = "Test set",
  medium="test", headline=d$headline, date=d$date, text=d$headline)
```

```
## Created articleset 29484: Test set in project 1
```

```
## Uploading 2 articles to set 29484
```

```
arts = amcat.getarticlemeta(conn, project = 1, articleset=setid, columns = c("date", "headline"))
```

```
## https://amcat.nl/api/v4/projects/1/articlesets/29484/meta?page_size=10000&format=rda&columns=date%2C
```

```
## Got 2 rows (total: 2 / 2)
```

```
head(arts)
```

id	date	headline
167443411	2001-01-01	A test
167443412	2008-12-31	Another test

Adding articles to article sets

Finally, you can add existing articles to a new or existing article set. For example, we could add the first articles from the set we used earlier to our new test set:

```
articles = meta$id[1:3]
amcat.add.articles.to.set(conn, project = 1, articleset = setid, articles=articles)
```

```
## [1] 29484
```

By specifying an articleset.name rather than an existing set id, we can also create a new articleset from a selection of articles:

```
articles = meta$id[1:3]
setid2 = amcat.add.articles.to.set(conn, project = 1, articleset.name="New test set", articles=articles)
```

```
## Created articleset 29485: New test set in project 1
```