Quicktext Query Language

QQL USER'SMANUAL

nanjing kuaiwen information technology co., LTD.

Catalogue

_	
	_
	2 8 4
	₩

Cata	alogue		1
1.	User N	Manual	3
2.	SELEC	CT Query Statement	3
	2.1	Statement Example	3
	2.2	Console Log.	4
	2.3	Query Result	4
3.	LIMIT	Query Statement	5
	3.1	Statement Example	5
	3.2	Console Log.	5
	3.3	Query Result	6
4.	CORP	US Query Statement	7
	4.1	Statement Example	7
	4.2	Console Log.	7
	4.3	Query Result	8
5.	COLU	MN Query Statement	9
	5.1	Statement Example	9
	5.2	Console Log	9
	5.3	Query Result1	0
6.	SORT	Query Statement	0
	6.1	Console Log1	1
	6.2	Query Result1	2
7.	FILTE	R Query Statement1	2
	7.1	Statement Example	3
	7.2	Console Log1	3
	7.3	Query Result1	4
8.	ACTIO	ON Query Statement1	4
	8.1	Statement Example	5

8.2	Console Log	15
8.3	Query Result	16

1.User Manual

The Query Statement must contain keywords. This manual is for Quicktext Query Language users. Quikctext Query Language is a scripting Language designed and implemented by nanjing kuaiwen information technology co., LTD.

In this manual, we take the Chinese keyword "人工智能(Artificial Intelligence)" for example

2.SELECT Query Statement

select Column Name from Corpus Name where keywords=[keywords]

2.1 Statement Example

Query all the reports related to "人工智能" under the CSSCI corpus, showing the publication time, journal name, author, title and link address:

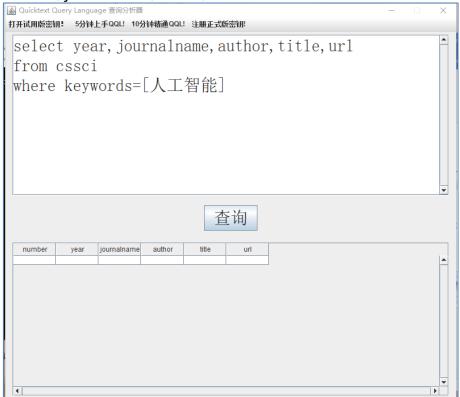


Figure 1

2.2 Console Log

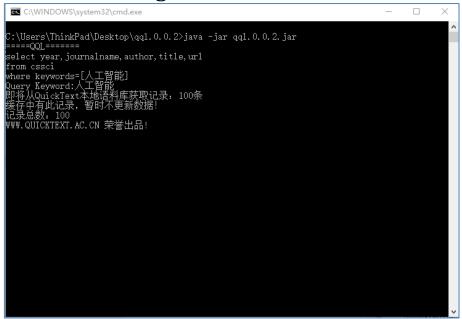


Figure 2



Figure 3

3.LIMIT Query Statement

select year, journalname, author, title, url from cssci where keywords = [人工智能]

limit Start position(default 0 starts from the first data), Gets the number of pieces of data (default is 100 pieces of data)

3.1 Statement Example

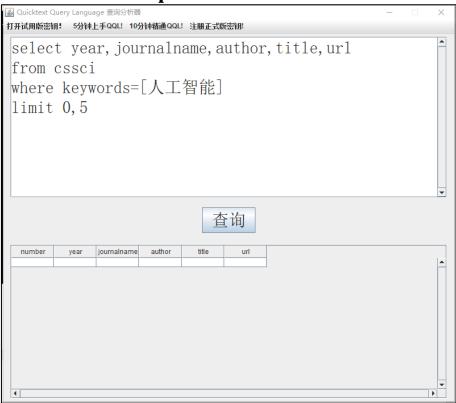


Figure 4

```
■ C:\Users\ThinkPad\Desktop\qq1.0.0.2>java -jar qq1.0.0.2.jar ======0QL=======
select year, journalname, author, title, url from cssci where keywords=[人工智能] limit 0,5 Query Keyword:人工智能即将人如ickText本地语判库获取记录。4990条接下中和记录,智时不更新数据!记录总数、4990
```

Figure 5

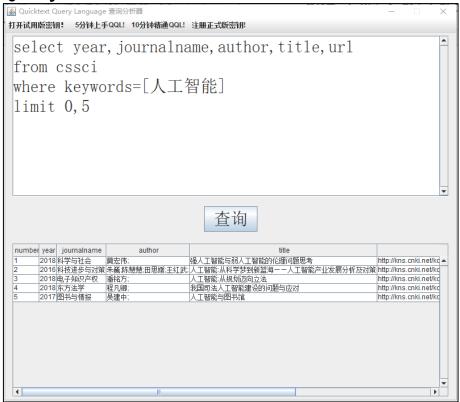


Figure 6

4.CORPUS Query Statement

select year,journalname,author,title,url from Corpus Name where keywords=[人工智能]

4.1 Statement Example

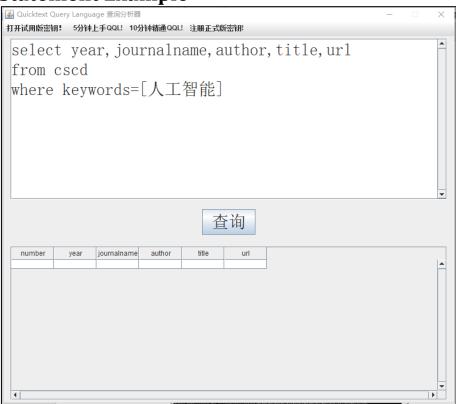


Figure 7

```
C:\WINDOWS\system32\cmd.exe

C:\Users\ThinkPad\Desktop\qq1.0.0.2>java -jar qq1.0.0.2.jar

-----QQL=------
select year, journalname, author, title, url
from cscd
where keywords=[人工智能]
Query Keyword:人工智能
即将从QuickText本地语料库获取记录: 0条
缓存中无此记录,即将强制更新数据!
即将从QuickText语料库获取记录: 4500条
开始从QuickText语料库下载语料
```

Figure 8

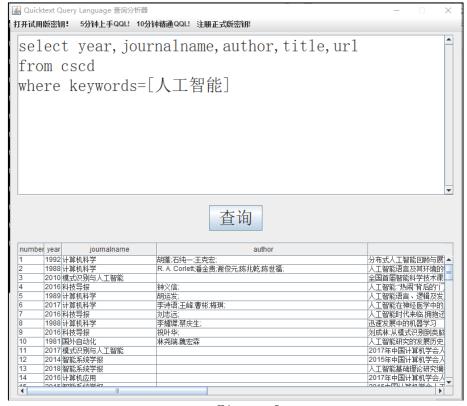


Figure 9

5.COLUMN Query Statement

select Column Name from cssci where keywords=[人工智能]

5.1 Statement Example



Figure 10

```
ThinkPad\Desktop\qq1.0.0.2\java -jar qq1.0.0.2.jar select year, title, url from casci where keywords=[人工智能]
UNEYN Keyword-【工智能]
UNFAULICHTEXT * Multiple Think Th
```

Figure 11

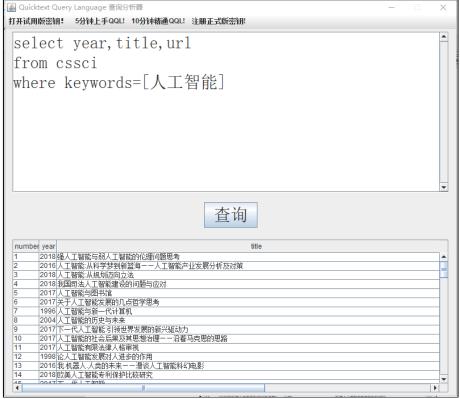


Figure 12

6.SORT Query Statement

select year,journalname,author,title,url from cssci where keywords=[人工智能] and sort by Column Name Ascending(Descending)

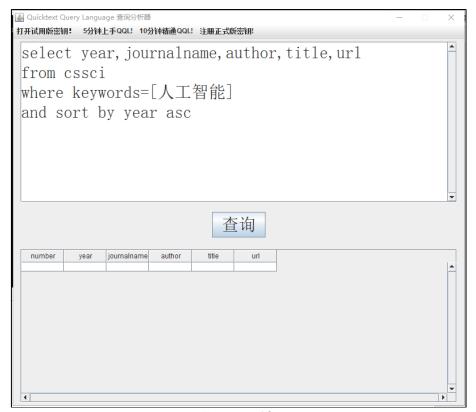


Figure 13

```
EX C:\Users\ThinkPad\Desktop\qq1.0.0.2>java -jar qq1.0.0.2.jar
====QQ1======
select year, journalname, author, title, url
from cssci
where keywords=[人工智能]
and sort by year ass
Query Keyword:人工智能
即將人如氏形主水社地语判库获取记录: 100条
该序中有此记录: 曾时不更新数据:
记录总数: 100
WWW.QUICKTEXT.AC.CN 荣誉出品:
```

Figure 14

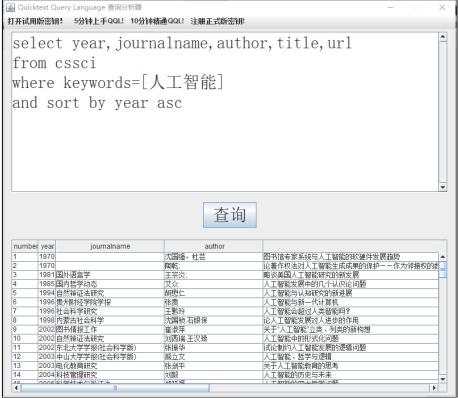


Figure 15

7.FILTER Query Statement

select year,journalname,author,title,url from cssci where keywords=[人工智能] and sort by year asc and filter by multiple conditions

7.1 Statement Example

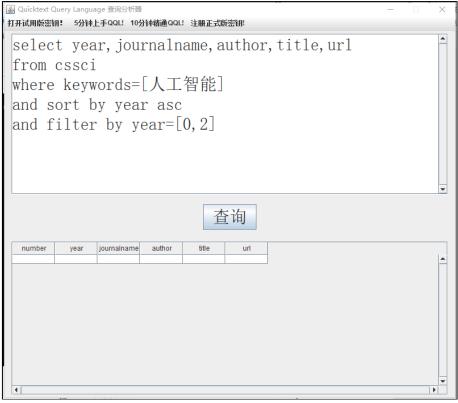


Figure 16

```
C:\Users\ThinkPad\Desktop\qq1.0.0.2\java -jar qq1.0.0.2.jar
====QQ[=======
select year, journalname, author, title, url
from cssci
where keywords=[人工智能]
and sort by year asc
and filter by year=[0,2]
Query Keyword:人工智能
即将人包试cHzxt本地语料库获取记录。100条
缓行中有比记录。暂时不更新数据!
记录总数,100
\text{VVV.QUICKTEXT.AC.CN 荣誉出品!}
```

Figure 17

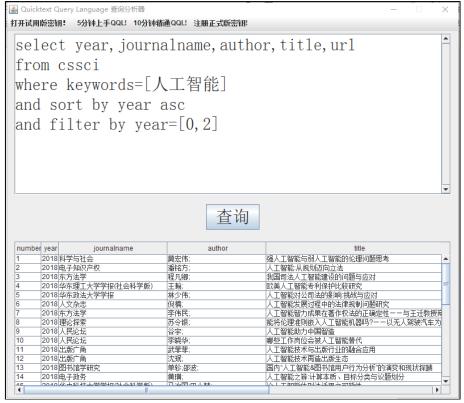


Figure 18

8.ACTION Query Statement

select year,journalname,author,title,url from cssci where keywords=[人工智能] then action=[bib=1.ris,bib=2.csv]

8.1 Statement Example

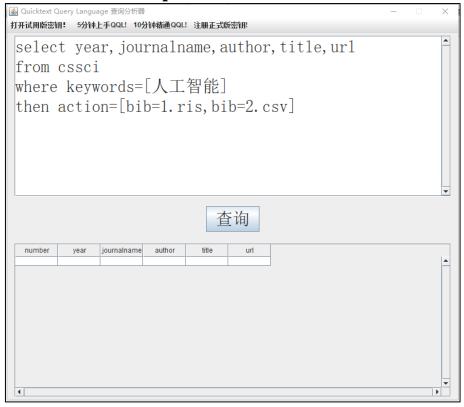


Figure 19

Figure $\overline{20}$

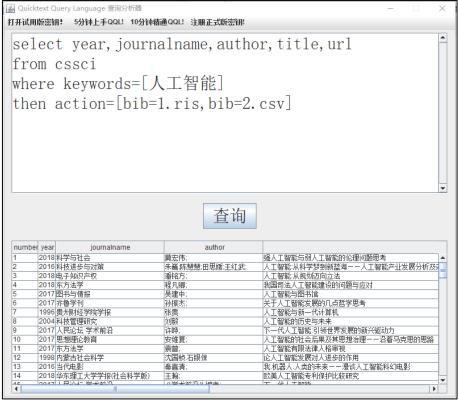


Figure 21