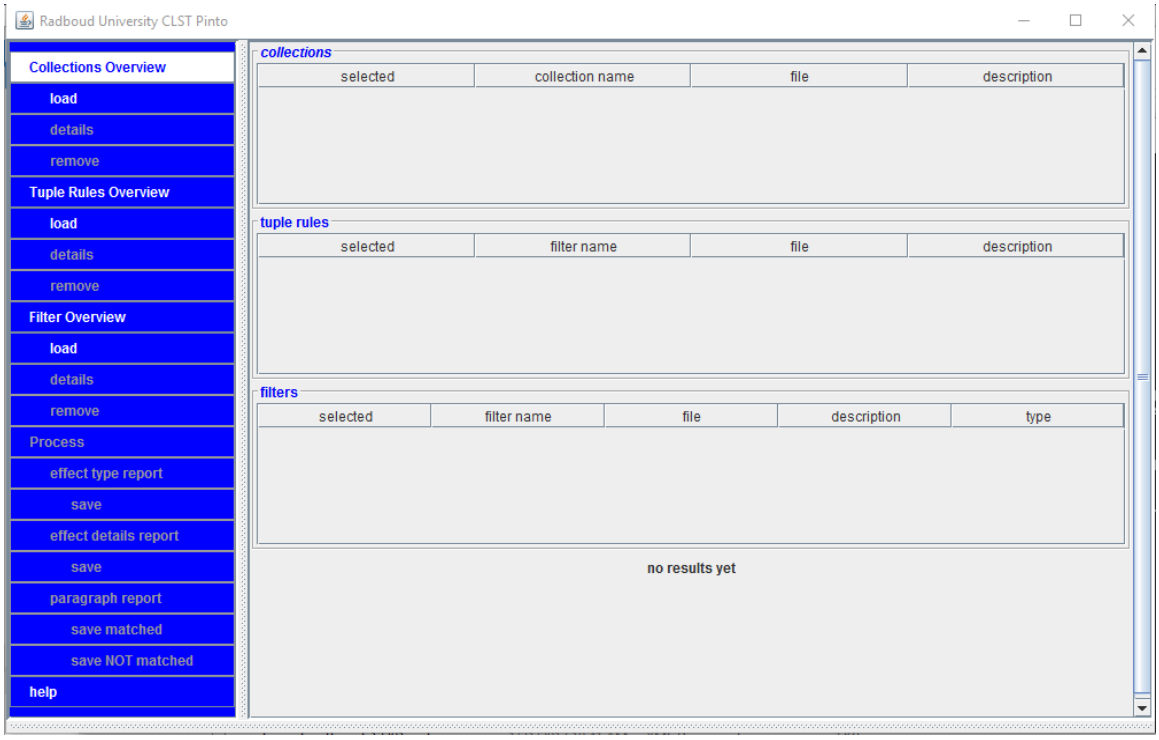
Pinto 1.5.1 – User Manual



2018

P. Beinema, Polderlink bv

N. Oostdijk, Radboud University / CLST

Contents

1. Introduction

This is the user manual for Pinto, an application developed by Polderlink bv, commissioned by Radboud University / CLST. Pinto supports users in effectively and efficiently searching large text collections, using a method developed by CLST. A sample application would be: searching internet forum posts to find possible side effects of dietary supplements. All examples in this manual focus on this application.

Starting from version 1.5 the data of several reports can be stored as tsv files: files containing tab-separated fields that can be imported easily in spreadsheet programs, such as MS Office Excel.

2. Hardware / software requirements

In order to use Pinto, the computer used should meet the following requirements:.

Operating System

Microsoft Windows, Linux, Apple OS X. (the software has been developed and tested on Windows 7 and 10).

Hardware

The computer should have at least 4GB of memory and a reasonably fast processor (e.g., Intel i5 or i7). At least 5GB of disk space is required.

Software

Pinto is a Java application. This implies that a Java Runtime Environment (JRE) for a recent version of Java should be installed. During development, version 8, update 121, dd. 2017-01-17 was used.

Available Java versions can be inspected in Windows → settings → Apps and features.

The Java Runtime Engine can be downloaded from the Oracle Corporation Java website: https://www.java.com/nl/download/

3. Installation

Pinto comes as zip file. Its constituent files must be extracted in a hard disk directory. To prepare Pinto for use, the following steps must be taken:

- Create a directory named “Pinto” in a convenient location on the hard disk.
- Place the Pinto zip file in this directory.
- Unzip the zip file, using a utility like 7zip or Winzip.

Once extracted, the directory will contain the following file structure:

Pinto	
\bin	Executable files required to run Pinto.
\data	Data files that can be used as input.
\collections	Files containing searchable input texts.
\filters	Files containing straightforward search terms definitions.
\reports	Files containing generated search results.
\rules	Files containing complex search term definitions.
\doc	Documentation (this file).
\src	Files containing source code. (This folder is not required to run Pinto. It is not supplied by default. It can be supplied by CLST).

After extraction Pinto is ready for use.

4. Pinto use

Start Pinto

(MS Windows:) Pinto is started from a Command prompt, that can be activated in two ways:

- Using the "search" button in the task bar, by entering *cmd*;
- By entering the key combination *Windows key* + *R*, and then entering *cmd* in the command line of the pop-up screen.

When the Command Window is available, it is required to navigate to Pinto's *bin* directory. This can be done by entering the *cd* (change directory) command, followed by a space character, and then the location of the Pinto *bin* directory.

Example

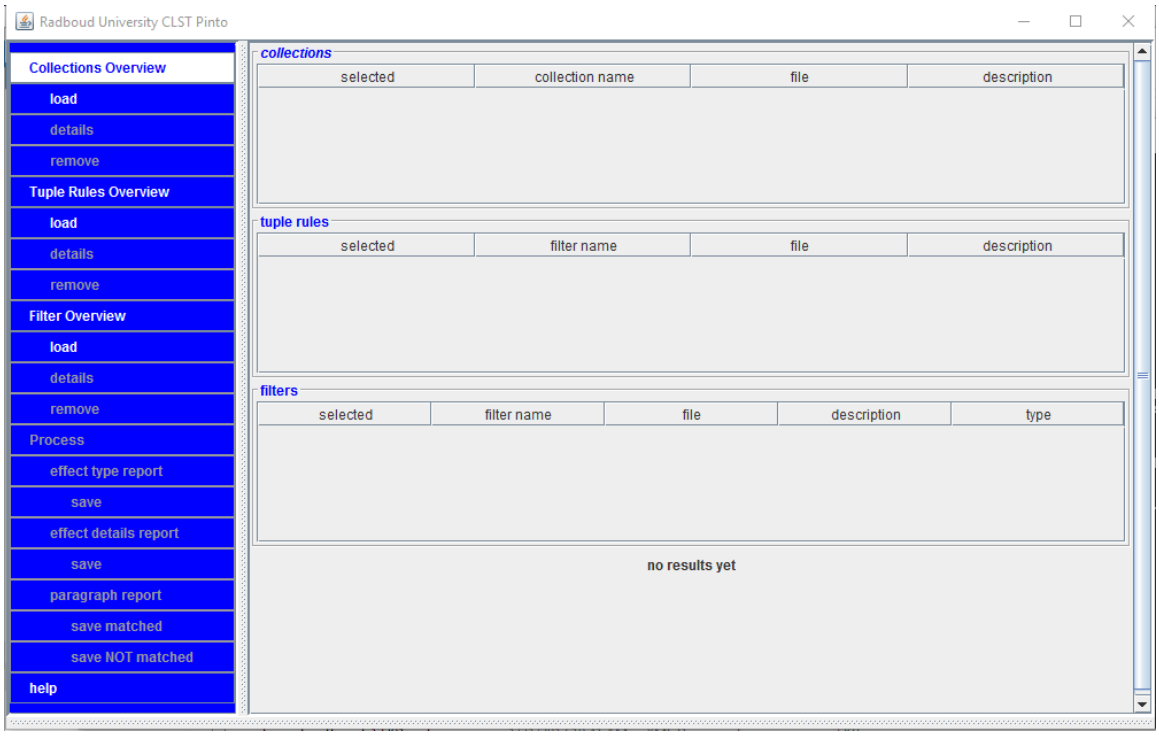
The Pinto directory is located on the C-disk in the User-directory of user 'UserX'. The location of the *bin* directory is:

C:\Users\UserX\Pinto\bin

The command to navigate to this directory is:

cd C:\Users\UserrX\Pinto\bin

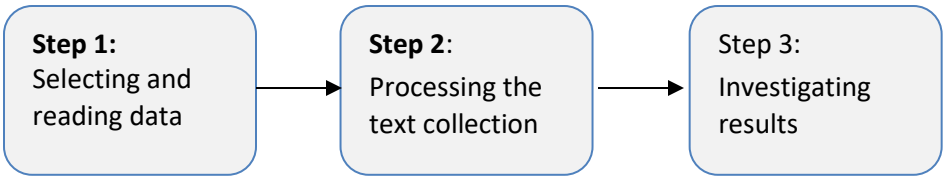
In the *bin* directory the command *run_pinto* is available. This command starts Pinto. Running this command results in displaying the following screen:



Picture showing a newly started Pinto application

Use

Searching a text collection with Pinto requires three steps:



Step 1: Selectering and Reading Data

Three components are required to search a text collection:

- A text collection to be searched.
- One set of rules that define compound search terms.
- One or more sets of simple search terms (filters).

Example

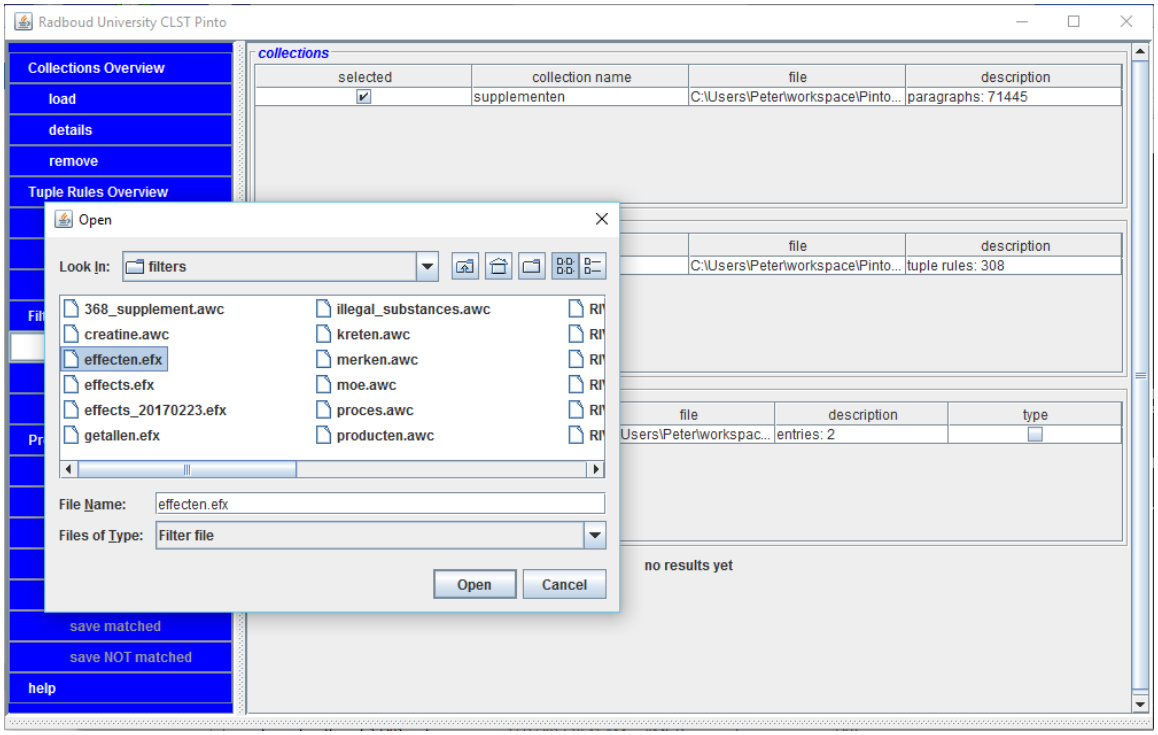
The text collection is a large set of forum messages. If one were to search for the possible effects of the supplement *creatine*, then the following lists and rules could be loaded in Pinto (Dutch terminology based):

Text collection: LorumIpsum.xml (a generated XML-text, created to demonstrate the use of Pinto. In actuality, a large collection of forum texts transformed to the same XML-layout using XSLT was used).

Rules:	effectensyntax.rls
Filters:	creatine.awc
	effecten.efx

The ***load*** button is used to select and read files. For text collections, rules and filters respectively, these are the load buttons in the ***Collections overview***, ***Tuple rules overview*** and ***Filter overview*** menu sections. When a ***load*** button is clicked, a pop-up window appears that shows the files that can be selected.

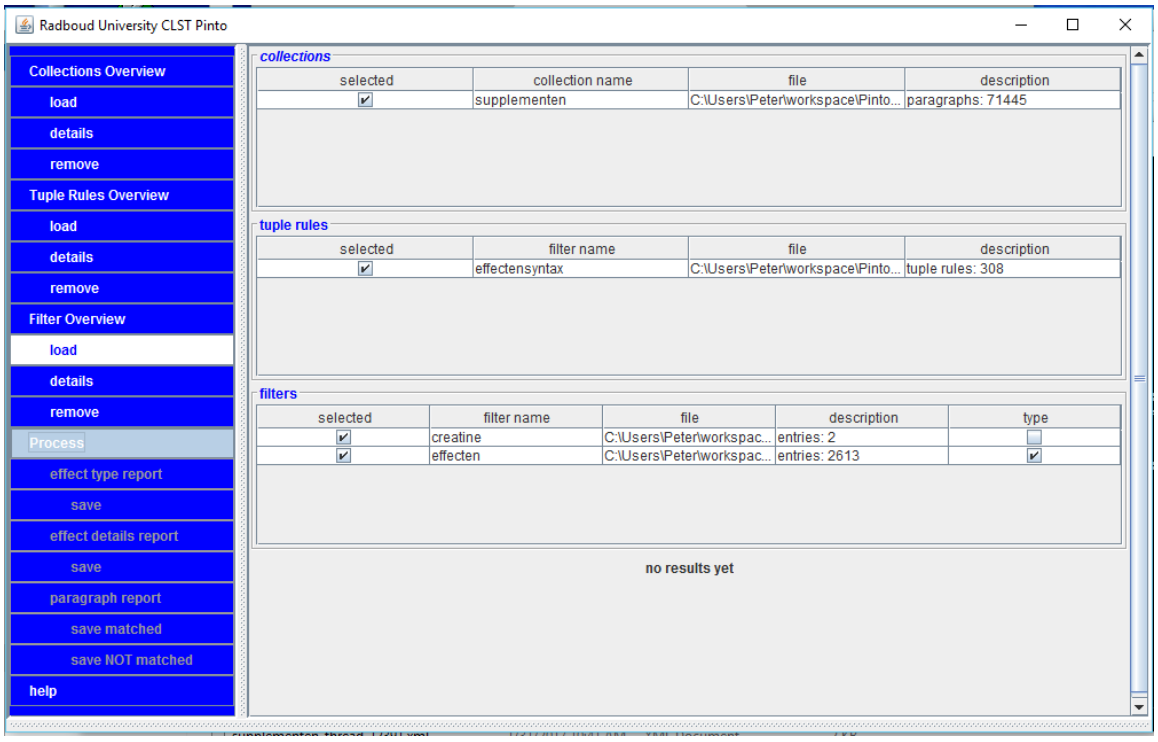
Pinto was developed to find relevant paragraphs in the text collections offered to it. A relevant paragraph is defined as a paragraph in which at least one search term was found for each of the selected filter lists. For the example used here this means that paragraphs are only relevant if they contain at least one item from the *creatine.awc* filter as well as an item from the effecten.efx rules set.



Picture showing the pop-up screen for loading filters.

After a file is loaded it appears in the corresponding overview section. In the overview section files can be (de) selected for processing, by checking the check box in the first column. When the ***Remove*** menu item is clicked, the checked files in the corresponding overview segments are removed. When the ***Details*** menu item is clicked detailed information on the checked files is displayed.

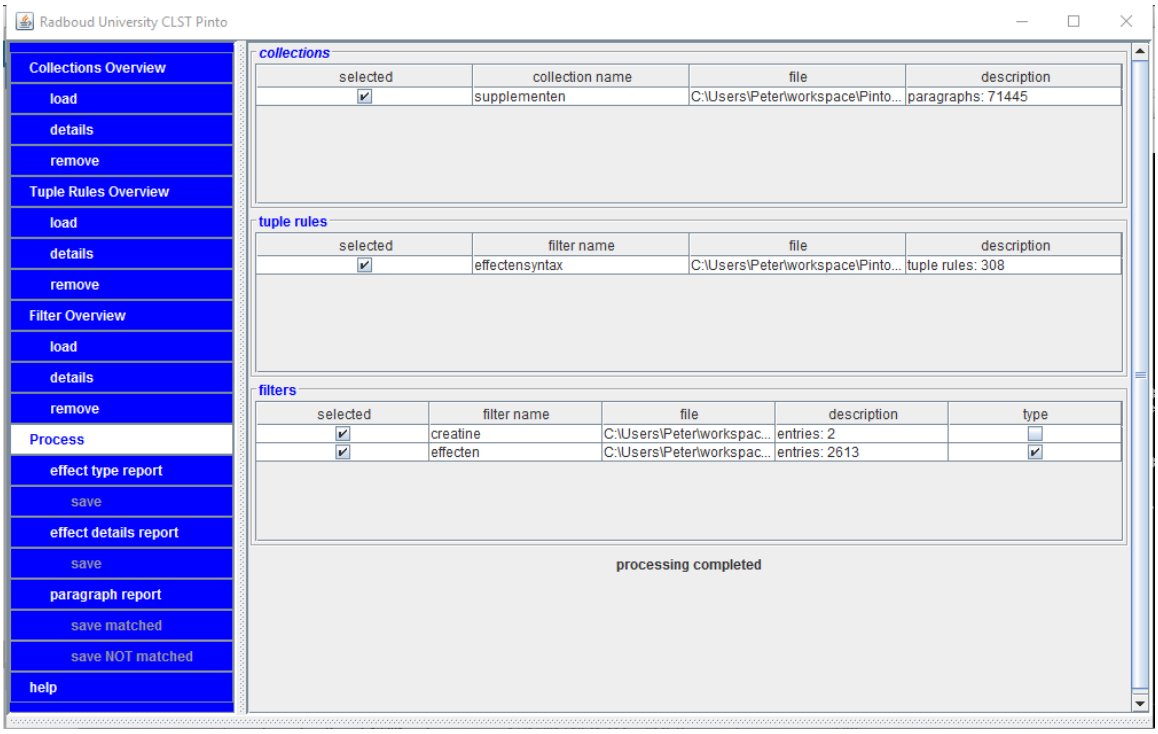
As soon as the minimally required set of files is loaded and selected, the ***Process*** menu item is activated: its text colour turns white and the item is now selectable. It is now possible to load and select additional files. Next, clicking the ***Process*** menu item will start searching for relevant paragraphs.



Files have been selected and processing has started

Step 2: Processing Text Collections

Click the ***Process*** menu item in the left column to start processing the text collections. The ***Process*** menu item button will be coloured light blue while ***Pinto*** has not finished processing. When processing is finished, the ***Process*** menu item turns white. The menu items to select the method of result representation now are activated: either aggregated at effect level (***Effect Report***), at effect detail level (**Effect Detail Report**), or as a list of relevant paragraphs (***Paragraph Report***).



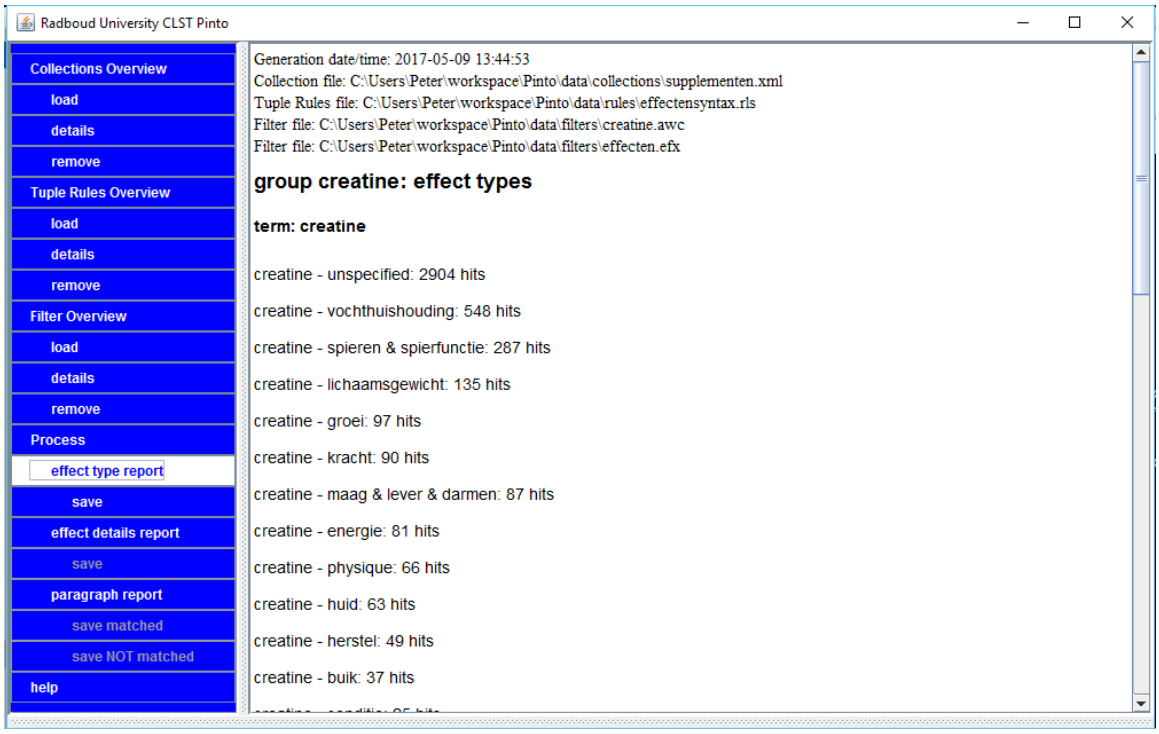
Processing has completed, and the reports can be inspected

Step 3: Inspecting Results

As specified in the preceding paragraph, results can be viewed in three possible ways: aggregated on effect-type level, as a list of detailed effects, or as a list of matching relevant paragraphs. For each of these a report menu item is available below the ***Process***-menu item.

Results aggregated at Effect Type level

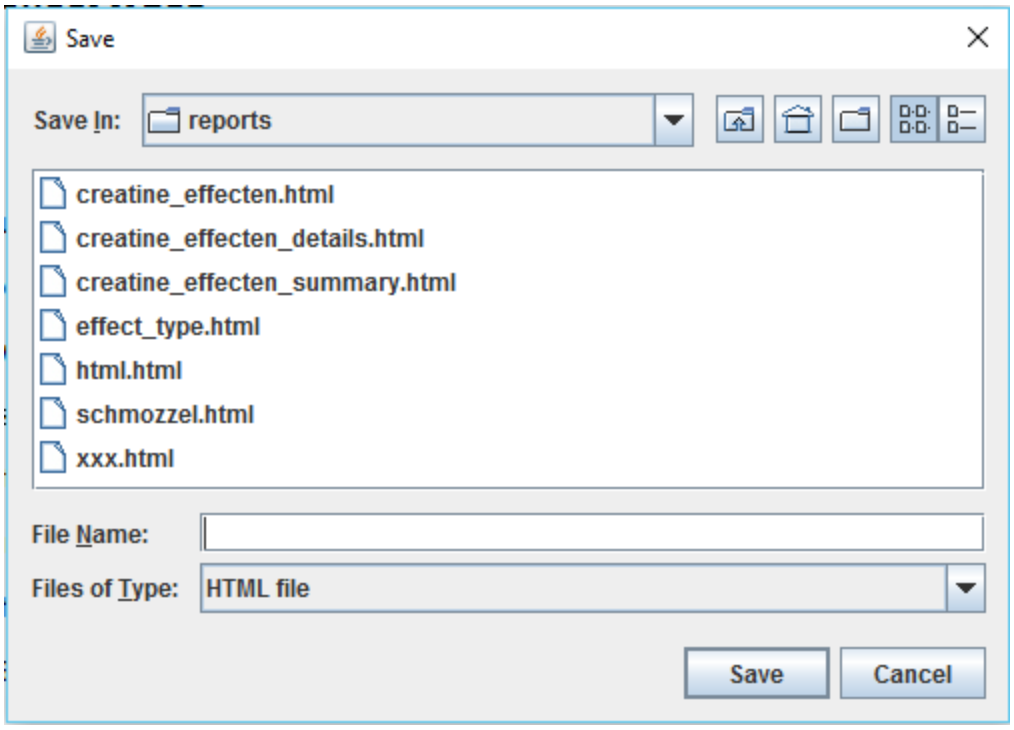
Selecting the ***Effect Type Report***-menu item will make the results available as an overview report. In our *creatine* and *effect* example the report will resemble the picture given below. Results are sorted by recognised product (number of occurrences) and on effect-type (number of occurrences).



Example: Report on aggregated effect type

When a report type is selected, the associated ***Save*** menu item is activated. It can be used to save the report in the form of an HTML file.

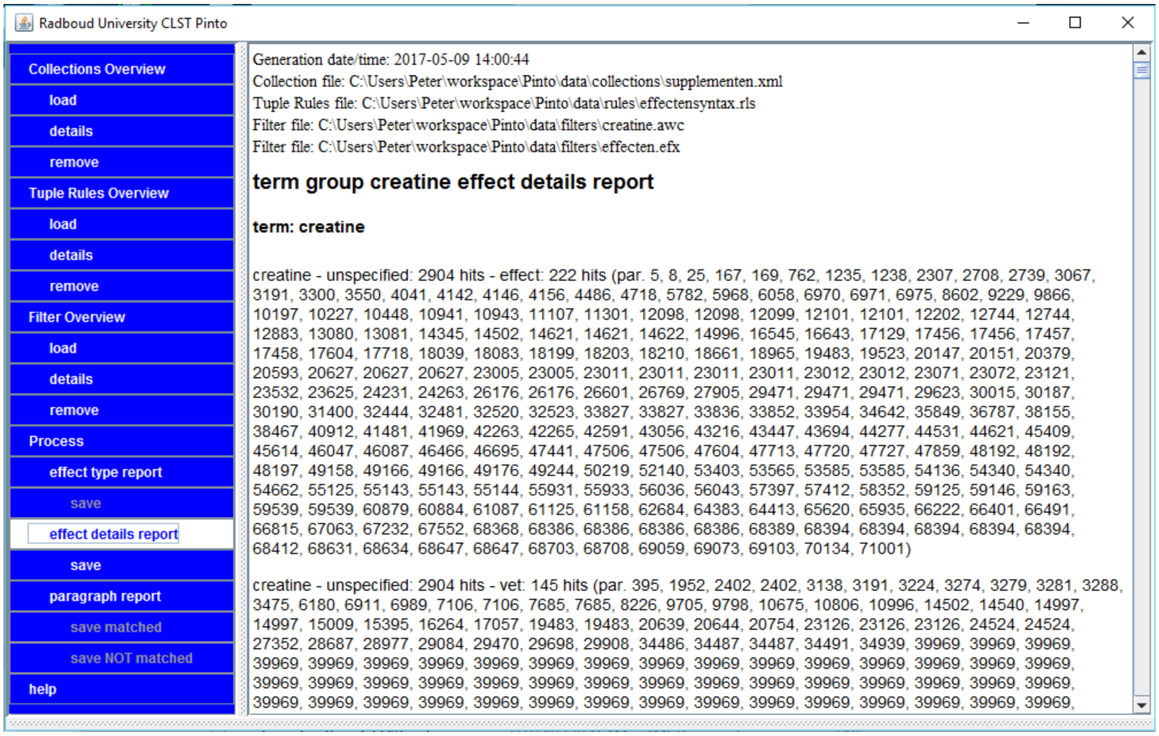
Pinto versions 1.5 and higher can also save the report data in the form of a .tsv file, i.e. a file containing tab-separated fields that can be opened in spreadsheet applications, such as MS Office Excel.



Pop-up screen to store the report as a file in directory "data/reports"

Results sorted by Detailed Effects

Selecting the ***Effect Details Report*** menu item will result in representing the report in the corresponding report format. In our *creatine* and *effects* example the report will resemble the picture below. Results are sorted by *product* (number of occurrences), and then by *effect-type*, and then by *effect*. For each report line, a list with corresponding paragraph numbers is given.



Example: Detailed report of effects

Here, too, the corresponding ***Save*** menu item is activated. It can be used to save the report in the form of an HTML file.

Pinto versions 1.5 and higher can also save the report data in the form of a .tsv file, i.e. a file containing tab-separated fields that can be opened in spreadsheet applications, such as MS Office Excel.

Report on matching paragraphs

This representation of results will show relevant paragraphs only. Search terms that were recognised are marked: those specified in efx-files are shown in **blue, whereas** those specified in awc-files are shown in **red**.

Example: Report on relevant paragraphs

Following the selecting of this report type, <u>two</u> ***Save*** menu items become available:

- ***Save matched***
- ***Save NOT matched***

The ***save matched*** menu item activates a pop-up window with which all *matched* paragraphs can be stored as a Pinto collection.

The ***save NOT matched*** menu item activates a pop-up window to save all <u>unmatched</u> paragraphs as a Pinto collection. This set is the complement of the matched set.

The reported paragraphs will have the exact same form as the original paragraphs in the input collection(s).

5. Input Data Files

Text collections

In the above examples, Pinto has been run on a set of text collections. The collections used are not included in the zip-file because of rights issues; a placeholder text ("LorumIpsum.xml") has been provided instead.

Collections offered to Pinto must have the following structure:

Example: XML format used.

```
<?xml version='1.0' encoding='iso-8859-1'?>
  <fora>
  <forum type='forum' name='NAME'>
    <thread id="nnnnn">
    <category>-</category>
    <title>TITLE</title>
    <posts>
      <post id="nnnnnn">
        <author>NAME</author>
        <timestamp>DATE-TIME</timestamp>
        <postindex>nn</postindex>
        <parentid>-</parentid>
        <body>
          <paragraph>TEXT</paragraph>
        </body>
        <upvotes>-</upvotes>
        <downvotes>-</downvotes>
      </post>
    </posts>
    </thread>
  </forum>
  </fora>
```

The blue fields define the required structure. The black fields can take any textual value.

The ‘tags’

<fora>…</fora>
<forum>…</forum>
<thread>…</thread>
<posts>…</posts>
<post>…</post>
<body>…</body>
<paragraph>…</paragraph>

can be repeated (i.e. can occur more than once).

Filters

Sample files containing search terms are provided with Pinto. There are two types: .awc files and.efx files.

Files of the .awc type contain at least one search term. If the user wants to search for a specific product (such as ‘creatine’ or ‘rpm’), then a file can be used dan that specifies the search term ‘creatine’ (or. ‘rpm’) only. If, on the other hand, the user wants to retrieve data on several products, then a file containing a list of search terms can be used.

The example lists that are supplied in file form are:

creatine.awc	list consisting of ‘creatine’ only
rpm.awc	list consisting of ‘rpm’ only
merken.awc	list of brands
producten.awc	list of products and product names
verbodenstoffen.awc	list of prohibited substances
effecten.efx	list of terms used to detect substance effects

.awc files

The .awc files list individual search terms, one search term per line. When matching the specified search terms, no distinction is made between uppercase and lowercase characters (words containing upper case characters can be specified, but case characters are treated as lowercase characters).

For the majority of the terms it was decided to require an exact match. The terms are immediately preceded and followed by the marker *b* . This marker matches a word boundary in the text. Blanks in search terms are allowed in combination with ‘*b*’.

Example

\banimal stack\b

\bcrazy\b

\bhalodrol\b

It is possible to skip the *b* marker. When the marker *b* is omitted at the beginning of the term, then Pinto will try to find all words in the text that end in the specified term. A similar effect occurs when *b* is left out at the end of the term: in that case all words starting with the specified term will be found. If no marker is used (neither at the beginning nor at the end of the term), all words in which the search term occurs will be found.

Example

\bamino Matches e.g. *amino, aminocell, aminolean, aminox*

The user can adapt the existing lists if so desired, and can add new lists. To edit a list any text editor can be used (such as Notepad or Textpad). Lists cannot be edited inside Pinto. Files containing search terms should be stored as *.awc* files in directory *Pinto**data**filters*.

The .efx file

Apart from *.awc* files, Pinto uses an *.efx* file. Although both file types specify search terms, there is a significant difference: the *.efx* file cannot be used stand-alone, but can only be used in combination with rules that are specified in the *.rls* file. Furthermore, the format of .efx entries differs from the format in .awc entries. In the current configuration it has been decided to specify *effects* in an .efx file. The reason for this is that it gives us the possibility to define large numbers of multiword n-grams without summing them up exhaustively, a quality that has proven to be very effective for defining *effects*. The *.efx* file describes the building blocks for effect n-grams: individual words/terms. The rules component (.rls file) describes how the building blocks can be combined.

As in the *.awc* files, the individual items of the *.efx* file are on separate lines. Every word/term in the 'effecten' list (file effecten.efx) is followed by a label that defines its part of speech (adjective, noun, etc.) and also gives an indication of its semantic value. Words/terms that clearly imply an effect should have an effect type classification (e.g. LONGEN & LUCHTWEGEN (Lungs and respiratory system), HART & VAATSTELSEL(Heart and vascular system)). These effect types are optional. For those cases where a search term does not unambiguously lead to one specific effect type, the effect type *UNSPECIFIED* can be used. When no effect type is specified, the default value UNSPECIFIED is implied.

Terms in the file *effecten.efx* are preceded and followed by the marker '*b*'. This forces an exact word boundary match. Just as is the case with .awc entries it is possible to leave out the *b* markers.

Example

\bademb	N4	LONGEN & LUCHTWEGEN
\bbloedvatverwijdende\b	ADJ2	HART & VAATSTELSEL
\bchronisch\b	ADJ13	UNSPECIFIED
\bleverkwaal\b	N60	MAAG & LEVER & DARMEN

The Appendix specifies the word classes defined in *effecten.efx* and used in *effectensyntax.rls*.

Rules

To search texts Pinto uses filters, i.e. lists with search terms. In most cases these can be summed up easily. Where this is not possible, one can make use of the possibility to recognise all combinations of words/terms by applying *rules*. Individual items are defined in an *.efx* file (see above). This file is the lexicon that is used by the rules in the *.rls* file.

The rules are so-called Finite State Rules. They describe which (types of) lexical items can occur by themselves and/or (and if so, how) they can be combined. For the current application the file *effectensyntax.rls* is used. The rules not only describe which combinations are possible, they also specify which of the lexicon items define the rule effect type.

Example

*N81
*ADJ2 N2

Example realizations of N81 are e.g. *energieverlies, krachttoename* and *celbalansvermindering*. These words by themselves indicate a specific effect. N2 words (e.g. *werking, effecten, gevolgen*) on the other hand are by themselves too vague or too general to speficy meaningful search terms. In combination with a preceding adjective such as an adjective of type 2 (ADJ2) these words are of interest. Examples of ADJ 2 type adjectives are e.g. *bloedverdunnende, herstellende* and *misselijkmakende*. The second rule in the above example describes combinations such as *bloedverdunnende werking, herstellende werking* and *misselijkmakende effecten.*

It is in principle possible for a user to modify the rules. However, because of the complex interaction between the rules and the words/terms this is discouraged.

6. Help

The '*Help'*-button activates a pop-up screen containing concise help information.

Appendix

This Appendix describes the word classes that are specified in *effecten.efx* and used in *effensyntax.rls*.

ADJ10	attributive (+e) adjectives that can be used in wider contexts and that not necessarily have to indicate effects,; e.g. *belangrijkste, acute*
ADJ11	attributive (deverbal, ingp +e) adjectives that can be used in a much wider sense and that not necessarily have to indicate effects; e.g. *aanhoudende, toenemende*
ADJ12	attributive (deverbal, ingp) adjectives that can be used in a wider sense that do not necessarily have to indicate effects; e.g. *diepgaand, stillend*
ADJ13	adjectives that can be used attributively as well as predicatively in a wider context; e.g. *duidelijk, breder*
ADJ2	attributive (deverbal, ingp +e) adjectives that denote a specific effect, e.g. *diuriserende, bloedverdunnende*
ADJ21	attributive (deverbal, ingp +e) adjectives that denote an underspecified effect; e.g. *afnemende, bevorderende*
ADJ3	(deverbal ingp) adjectives that can be used attributively and predicatively and that denote a specific effect; e.g. *bloedvatverwijdend, stressverhogend*
ADJ31	(deverbal ingp) adjectives that can be used attributively and predicatively and that denote an underspecified effect; e.g. *beschermend, bindend*
ADJ4	(almost all)* adjectives that are used predominantly predicatively and that indicate how someone is feelings or wat he/she looks like; e.g. *kotsmisselijk, kortademig* * possibly risky: *breed, dik, onprettig, slap, sterk, stevig, super, zwak*
ADJ5	attributive (+e) adjectieves that denote a specific effect; e.g. *allergische, hormonale*
ADJ51	adjectives that can be used attributively as well as predicatively, which by themselves indicate a specific effect; e.g. *psychologisch, fysiek*
ADJ6	attributive (deverbal, edp +e) adjectives that denote the specific nature of an effect; e.g. *belemmerde, gejaagde*
ADJ67	deverbal (edp) adjectives that can be used attributively as well as predicatively, and can be used in a wider context; e.g. *opgenomen, afgescheiden*
ADJ7	deverbal (edp) adjectieves that can be used attributively as well as predicatively, and can be used in a wider context; e.g. *verminderd, uitgerekt*
ADV1	intensifying adverbs that can be used in wider contexts; e.g. *enorm, extreem*
ADV2	general adverbs that can be used in wider contexts; e.g. *langzaam, ongunstig*
N1	generic nouns indicating a change (but not in relation to a specific body part, organ, or the like); e.g. *afname, verhoging*
N2	generieke nouns that are more or less synonymous to 'effect'; e.g. *bijwerking, symptoom*

N3	a limited set of generic nouns that indicate an effect (in contrast to N1 these express a result or state); e.g. *probleem*, *klachten*
N4	nouns naming body parts, organs, and the like that can be used in wider contexts; e.g. *hoofd, benen*
N5	the nouns *gevoel* en *gevoelens* that both can be used in a wider context
N60	nouns indicating a specific effect; e.g. *diarree, haaruitval*
N61	nouns indicating a more generic effect; e.g. *ontsteking, zwelling*
N7	nouns that typically are a part of a compound noun (but can occur in isolation); e.g. *hersen, ademhalings*
N8	nouns that describe body-related things that require additional information to express an effect, possibly by means of a modifying adjective; e.g. *buikvet, botmassa*
N81	nouns expressing a specific change (in contrast to N1 generic nouns); e.g. *krachttoename, energieverlies*
N9	generic nouns that can occur a more general context; e.g. *geleiding, ervaring*
N10	nouns expressing a unit of weight or size; e.g. *kilo, centimeter*
V1	Vinfin (separable or not separable), e.g. *aankomen, bevorderen*
V11	verb part of separable Vinfin, e.g. *vallen, tasten* (as verb parts of *aan-/afvallen, aantasten*)
V2	Vtense pres (separable or not separable), e.g. *opbouwt, vermindert*
V21	verb part of separable Vtense pres, e.g. *bouwt, breekt* (as verb parts of *bouwt op*, *breekt af*)
V3	Vtense past (separable or not separable), e.g. *opbouwde, verminderde*
V31	verb part of separable Vtense past, e.g. *bouwde, brak* (as verb parts of *bouwde op*, *brak af*)
AU	auxiliary verbs, e.g. *kunnen, laten*
DET	determiners, e.g. *de, je, mijn*
P1	phrases, variations on ‘last hebben van’
PR	particles (parts of a separable verb), e.g. *aan, af* (as in *valt af, komt aan*)
Q	quantifier, e.g. *bakken, veel*