

RA: a new engine for NooJ

max.silberztein@univ-fcomte.fr

The engine, v1.0

- Open source
- Developed with Swift, compatible with MacOS, Linux, Unix, Windows
- 19 modules (dictionary, orthographic and syntactic grammars...)
- Test-Driven Development (9,000 lines of test code)
- Managed by a version control system (git)
- Toolbox and examples downloadable at www.nooj4nlp.org
- Collaborative, source available at:
<https://gitlab.com/Silberz/ra-linguistic-engine>

The source

The screenshot shows the GitLab web interface for the repository `ra-linguistic-engine` by Max Silberstein. The interface includes a sidebar with navigation options, a header with the repository name and project ID, and a main content area showing the latest commit and a list of files.

Repository Information:

- Project ID: 31651161
- 48 Commits, 1 Branch, 0 Tags, 26 MB Project Storage
- Buttons: Find file, Web IDE, Clone

Latest Commit:

- Commit ID: 7ab1b40d
- Message: added concordance
- Author: Max Silberstein
- Time: 21 hours ago

Files and Commits:

Name	Last commit	Last update
LinguisticResources	added en-PrepNg.nog	2 days ago
RA.xcodeproj	added concordance	21 hours ago
RA	added concordance	21 hours ago
RATests	ALU: added various checking and parsing	2 days ago
concordance	added concordance	21 hours ago
dic2lst	Added lst2ra nog2ra nom2ra printra prints...	2 days ago
lexicalanalysis	Added lst2ra nog2ra nom2ra printra prints...	2 days ago
lst2ra	Added lst2ra nog2ra nom2ra printra prints...	2 days ago
nog2ra	Added lst2ra nog2ra nom2ra printra prints...	2 days ago
nom2ra	Added lst2ra nog2ra nom2ra printra prints...	2 days ago
printra	Added lst2ra nog2ra nom2ra printra prints...	2 days ago

Linguistic resources

.dic : dictionary, i.e. list of Atomic Linguistic Units (ALUs)

.idx : an index of matches with corresponding outputs

.lst : a list of forms linked to their ALUs

.nof : inflectional / derivational grammar

.nog : syntactic grammar

.nom : morphological / orthographic grammar

.not : a text + a Text Annotation Structure (TAS)

.ra : a Multiple Generalized Finite-State Automaton (MGFSA)

RA is 95% compatible with NooJ but...

- All automata are Multiple-Generalized-Finite-State-Automata (MGFSA)
- Dictionaries, orthographic, morphologic and syntactic grammars are compiled into MGFSA
- All MGFSA are stored in the same file format : **.ra**
- Only one format for all types of dictionary (DELAS, DELAF and DELAV):
 - NooJ: **eating,eat,V+G**
 - RA: **eating,V+G+_LEX=""**
- Text can be segmented into text units (e.g. sentences) with orthographic grammars
- A new collocation operator for grammars, e.g. “pizzeria” terms in the context of “Clinton”:
Pizzagate = <pizzeria> & Clinton ;
- All languages’ morphological operators available: Arabic **<M>** operator in NooJ → **<ARM>** in RA
- Special character “**#**” replaced with **<_>**; “**=**” replaced with **<_>**; “**_**” replaced with **<_ ->**
- ...

RA linguistic resources are easier to understand

RA's parsers are more efficient than NooJ's...

- Analysis of agglutination, e.g. a wordform contains 45 letters:

pneumonoultramicroscopicsilicovolcanoconiosis

needs to be analyzed as a sequence of 6 ALUs: <pneumono,PREFIX> <ultra,PREFIX>
<microscopic,PREFIX> <silico,PREFIX> <volcano,PREFIX> <coniosis,N>

- NooJ grammar:

(\$(P <L> <L>* \$)/<\$P=:PREFIX>)* \$(N <L> <L>* \$)/<\$N=:N>

Complex to understand, and not efficient: $O(2^n)$

=> NooJ checks $2^{45} = 35,184,372,088,832$ constraints <\$P=:PREFIX>

- RA grammar: **<PREFIX>* <N>**

Easier to understand, and very efficient: $O(n)$

=> RA performs 45 dictionary lookups in worst case

RA toolbox v0.9 contains 12 line commands

```
Command Prompt

C:\Users\Max\RA-Workbench\_windows>dir
Volume in drive C is BOOTCAMP
Volume Serial Number is 0246-9FF7

Directory of C:\Users\Max\RA-Workbench\_windows

26/06/2022  12:19    <DIR>          .
26/06/2022  12:19    <DIR>          ..
24/06/2022  09:26           300,544 concordance.exe
24/06/2022  09:30           868,352 dic2lst.exe
24/06/2022  09:37        1,596,416 lexicalanalysis.exe
24/06/2022  09:39           603,648 lst2ra.exe
25/06/2022  12:40           529,408 nog2ra.exe
24/06/2022  09:51           535,040 nom2ra.exe
24/06/2022  09:53           304,640 printra.exe
24/06/2022  09:55           219,136 printtas.exe
24/06/2022  09:57           332,288 ra2lst.exe
24/06/2022  10:02           839,168 segment.exe
24/06/2022  10:04        1,694,720 syntacticanalysis.exe
24/06/2022  10:06        1,649,664 syntacticlocate.exe
               12 File(s)          9,473,024 bytes
               2 Dir(s)  288,880,513,024 bytes free

C:\Users\Max\RA-Workbench\_windows>
```

Parameters to set before using RA tools

- In Windows 10's Start text field (bottom left of the screen), enter **edit the system environment variables**, click "Environment variables" at bottom of window, select "Path", Click "Edit", and then add directory **C:\Users\Joe\RA-Workbench_windows**.
- To use the Command Prompt: enter **cmd /K chcp 65001** in Windows Start text field.
- To use the Powershell: right-Click Windows Start icon, then click "Window PowerShell". Enter the command:
\$OutputEncoding = [console]::InputEncoding = [console]::OutputEncoding = New-Object System.Text.UTF8Encoding

dic2lst

inputs:

- dictionaries
- inflectional and derivational grammars

output:

- list of all generated inflected and derived forms

Generated forms are linked to the lexeme via the **LEX** special property

```
Command Prompt
C:\Users\Max\RA-Workbench>dic2lst
Usage: dic2lst dictionary.dic+ morphologicalGrammars.nof* propertiesDefinition.def
Note: some morphologicalGrammars might be referenced from inside dictionaries with the #use command
e.g.: dic2lst en-Nouns.dic en-Verbs.dic properties.def
e.g.: dic2lst en-dictionary.dic Nouns.nof Verbs.nof Derivations.nof properties.def
=> prints list of all generated forms

C:\Users\Max\RA-Workbench>dic2lst simplifiedictionary.dic properties.def
aberration constant,N+DOM="Phys"+_LEX="<E>" +FLX=APPLE+Number=s
aberration constants,N+DOM="Phys"+_LEX="<B>" +FLX=APPLE+Number=p
aberration,N+_LEX="<E>" +FLX=APPLE+Number=s
aberrations,N+_LEX="<B>" +FLX=APPLE+Number=p
aberrational,A
Abert's towhee,N+_LEX="<E>" +FLX=APPLE+Number=s
Abert's towhees,N+_LEX="<B>" +FLX=APPLE+Number=p
abesse,N+Distribution=Hum+_LEX="<E>" +FLX=APPLE+Number=s
abesses,N+Distribution=Hum+_LEX="<B>" +FLX=APPLE+Number=p
abets,V+Trans=T+_LEX="<B>" +FLX=BEG+Number=s+Pers=3+Tense=PR
abet,V+Trans=T+_LEX="<E>" +FLX=BEG+Number=p+Pers=123+Tense=PR
abetting,V+Trans=T+_LEX="<B4>" +FLX=BEG+Tense=G
abetted,V+Trans=T+_LEX="<B3>" +FLX=BEG+Number=p+Pers=123+Tense=PT
abetted,V+Trans=T+_LEX="<B3>" +FLX=BEG+Number=s+Pers=123+Tense=PT
abetted,V+Trans=T+_LEX="<B3>" +FLX=BEG+Tense=PP
abet,V+Trans=T+_LEX="<E>" +FLX=BEG+Number=s+Pers=12+Tense=PR
abet,V+Trans=T+_LEX="<E>" +FLX=BEG+Tense=INF
abetment,N+_LEX="<E>" +FLX=APPLE+Number=s
abetments,N+_LEX="<B>" +FLX=APPLE+Number=p
abettal,N+_LEX="<E>" +FLX=APPLE+Number=s
abettals,N+_LEX="<B>" +FLX=APPLE+Number=p
abetter,N+Distribution=Hum+_LEX="<E>" +FLX=APPLE+Number=s
abetters,N+Distribution=Hum+_LEX="<B>" +FLX=APPLE+Number=p
abetting,A
abetting,N+_LEX="<E>" +FLX=APPLE+Number=s
abettings,N+_LEX="<B>" +FLX=APPLE+Number=p
abettor,N+Distribution=Hum+_LEX="<E>" +FLX=APPLE+Number=s
abettors,N+Distribution=Hum+_LEX="<B>" +FLX=APPLE+Number=p
abeyance,N+_LEX="<E>" +FLX=APPLE+Number=s
abeyances,N+_LEX="<B>" +FLX=APPLE+Number=p
# Dictionary successfully generated 28 forms.

C:\Users\Max\RA-Workbench>
```

lst2ra

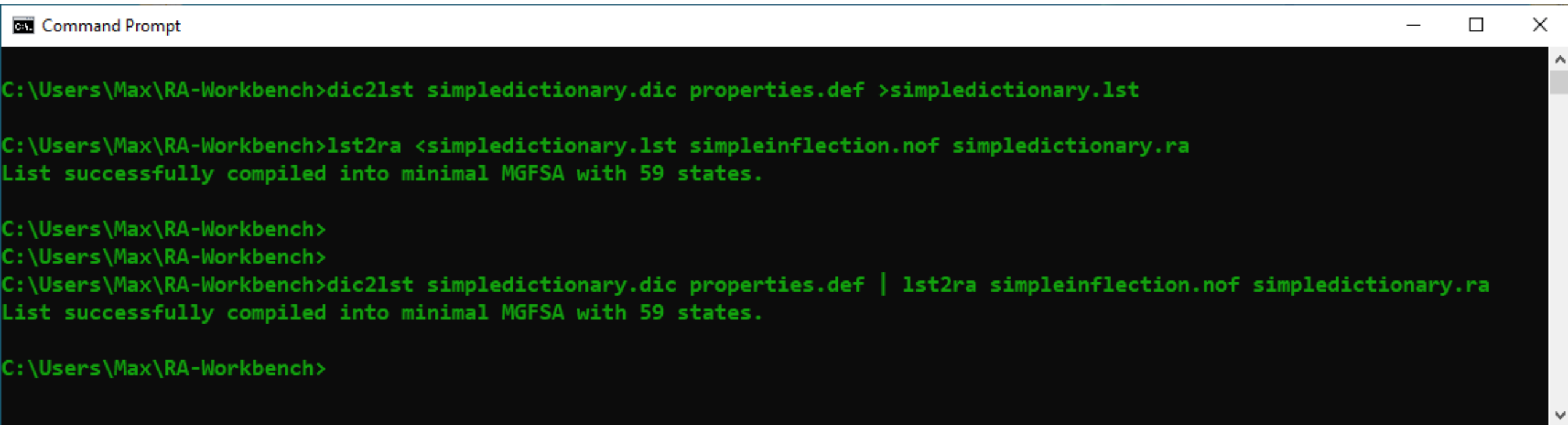
inputs:

- a list of all generated forms + inflectional and derivational grammars

outputs:

- a MGFSa

The MGFSa is reversible, i.e. can be used to parse or to generate texts



```
Command Prompt

C:\Users\Max\RA-Workbench>dic2lst simplifiedictionary.dic properties.def >simplifiedictionary.lst

C:\Users\Max\RA-Workbench>lst2ra <simplifiedictionary.lst simpleinflection.nof simplifiedictionary.ra
List successfully compiled into minimal MGFSa with 59 states.

C:\Users\Max\RA-Workbench>
C:\Users\Max\RA-Workbench>
C:\Users\Max\RA-Workbench>dic2lst simplifiedictionary.dic properties.def | lst2ra simpleinflection.nof simplifiedictionary.ra
List successfully compiled into minimal MGFSa with 59 states.

C:\Users\Max\RA-Workbench>
```

printra

displays a readable
version of a
MGFSA (.ra file)

```
C:\Windows\System32\cmd.exe

C:\Users\Max\RA-Workbench>printra simplifiedictionary.ra
Dictionary-level Recursive Automaton, v1.0
MGFSA contains 3 graphs; vocabulary contains 47 symbols.
Main:
  0: a→2 A→25
  1T
  2: b→3
  3: e→4
  4: r→5 t→43 s→39 y→56
  5: r→6
  6: a→7
  7: t→8
  8: i→9
  9: o→10
 10: n→11
 11: a→23 →12 <E>/N+_LEX="<E>"+FLX=APPLE+Number=s→1 s→22
 12: c→13
 13: o→14
 14: n→15
 15: s→16
 16: t→17
 17: a→18
 18: n→19
 19: t→20
 20: <E>/N+DOM="Phys"+_LEX="<E>"+FLX=APPLE+Number=s→1 s→21
 21: <E>/N+DOM="Phys"+_LEX="<B>"+FLX=APPLE+Number=p→1
 22: <E>/N+_LEX="<B>"+FLX=APPLE+Number=p→1
 23: l→24
 24: <E>/A→1
```

... (3 graphs; 47 symbols; 59 + 4 + 18 states).

nom2ra

input:

- a .nom graphical or textual grammar

output:

- the corresponding MGFSFA

options:

- determinize
- minimize

```
C:\Windows\System32\cmd.exe

C:\Users\Max\RA-Workbench>type simpleregexp.nom
Main = (a|b)*ab(a|b)* ;

C:\Users\Max\RA-Workbench>nom2ra simpleregexp.nom
Orthographic grammar simpleregexp.ra successfully compiled.
1 orthographic grammar successfully compiled.

C:\Users\Max\RA-Workbench>printra simpleregexp.ra
Orthographic-level Recursive Automaton, v1.0
MGFSFA contains 1 graph; alphabet contains 3 symbols.
Main:
  0: <E>->2
  1T
  2: <E>->3 <E>->4
  3: <E>->10
  4: <E>->6 <E>->8
  5: <E>->2
  6: a->7
  7: <E>->5
  8: b->9
  9: <E>->5
  10: <E>->12
  11: <E>->1
  12: a->13
  13: <E>->14
  14: <E>->16
  15: <E>->11
  16: b->17
  17: <E>->18
  18: <E>->19 <E>->20
  19: <E>->15
  20: <E>->22 <E>->24
  21: <E>->18
  22: a->23
  23: <E>->21
  24: b->25
  25: <E>->21

C:\Users\Max\RA-Workbench>

C:\Windows\System32\cmd.exe

C:\Users\Max\RA-Workbench>nom2ra simpleregexp.nom -determinize
Orthographic grammar simpleregexp.ra successfully compiled.
1 orthographic grammar successfully compiled.

C:\Users\Max\RA-Workbench>printra simpleregexp.ra
Orthographic-level Recursive Automaton, v1.0
MGFSFA contains 1 graph; alphabet contains 3 symbols.
Main:
  0: a->2 b->1
  1: a->2 b->1
  2: a->2 b->3
  3T a->5 b->4
  4T a->5 b->4
  5T a->5 b->6
  6T a->5 b->4

C:\Users\Max\RA-Workbench>nom2ra simpleregexp.nom -minimize
Orthographic grammar simpleregexp.ra successfully compiled.
1 orthographic grammar successfully compiled.

C:\Users\Max\RA-Workbench>printra simpleregexp.ra
Orthographic-level Recursive Automaton, v1.0
MGFSFA contains 1 graph; alphabet contains 3 symbols.
Main:
  0: a->2 b->0
  1T a->1 b->1
  2: a->2 b->1

C:\Users\Max\RA-Workbench>
```

nog2ra

input:

- a .nog graphical or textual grammar

output:

- the corresponding MGFSAs in the corresponding .ra file

options:

- determinize
- minimize

```
C:\Windows\System32\cmd.exe

C:\Users\Max\RA-Workbench>type np.nog
Main = <E>/<NP
      ( :PLURAL_DET (<A> | <E>) <N+p> |
        :SINGULAR_DET (<A> | <E>) <N+s> )
      <E>/> ;
PLURAL_DET = certain | few | many | the ;
SINGULAR_DET =
  a | an | some | the | this |
  part of the | such a | the flood of ;

C:\Users\Max\RA-Workbench>nog2ra np.nog -minimize
Syntactic grammar np.ra successfully compiled.

C:\Users\Max\RA-Workbench>printra np.ra
Syntactic-level Recursive Automaton, v1.0
MGFSAs contain 3 graphs; vocabulary contains 20 symbols.
Main:
  0: <E>/<NP→2
  1: <A>→7 <N+s>→3
  2: :PLURAL_DET→4 :SINGULAR_DET→1
  3: <E>/→5
  4: <A>→6 <N+p>→3
  5T
  6: <N+p>→3
  7: <N+s>→3
PLURAL_DET:
  0: many→1 the→1 certain→1 few→1
  1T
SINGULAR_DET:
  0: the→1 a→2 an→2 some→2 this→2 part→4 such→6
  1T flood→3
  2T
  3: of→2
  4: of→5
  5: the→2
  6: a→2

C:\Users\Max\RA-Workbench>
```

ra2lst

input:

- an MGFSFA

output:

- the list of generated sequences

options:

- limit number of loops

```
C:\Windows\System32\cmd.exe

C:\Users\Max\RA-Workbench>type csar.nom
Main = (c|t)/t (s|z)/s ar/ar,N+Hum (<E>/+m | ina/+f) (<E>/+s | s/+p) ;

C:\Users\Max\RA-Workbench>nom2ra csar.nom
Orthographic grammar csar.ra successfully compiled.
1 orthographic grammar successfully compiled.

C:\Users\Max\RA-Workbench>ra2lst csar.ra
csar,tsar,N+Hum+m+s
csars,tsar,N+Hum+m+p
csarina,tsar,N+Hum+f+s
csarinas,tsar,N+Hum+f+p
czar,tsar,N+Hum+m+s
czars,tsar,N+Hum+m+p
czarina,tsar,N+Hum+f+s
czarinas,tsar,N+Hum+f+p
tsar,tsar,N+Hum+m+s
tsars,tsar,N+Hum+m+p
tsarina,tsar,N+Hum+f+s
tsarinas,tsar,N+Hum+f+p
tzar,tsar,N+Hum+m+s
tzars,tsar,N+Hum+m+p
tzarina,tsar,N+Hum+f+s
tzarinas,tsar,N+Hum+f+p
# 16 generated sequences.

C:\Users\Max\RA-Workbench>
```

```
C:\Windows\System32\cmd.exe

C:\Users\Max\RA-Workbench>type np.nog
Main = <E>/<NP
      ( :PLURAL_DET (<A> | <E>) <N+p> |
        :SINGULAR_DET (<A> | <E>) <N+s> )
      <E>/> ;
PLURAL_DET = certain | few | many | the ;
SINGULAR_DET =
  a | an | some | the | this |
  part of the | such a | the flood of ;

C:\Users\Max\RA-Workbench>nog2ra np.nog
Syntactic grammar np.ra successfully compiled.

C:\Users\Max\RA-Workbench>ra2lst np.ra
certain <A> <N+p>,<NP>
certain <N+p>,<NP>
few <A> <N+p>,<NP>
few <N+p>,<NP>
many <A> <N+p>,<NP>
many <N+p>,<NP>
the <A> <N+p>,<NP>
the <N+p>,<NP>
a <A> <N+s>,<NP>
a <N+s>,<NP>
an <A> <N+s>,<NP>
an <N+s>,<NP>
some <A> <N+s>,<NP>
some <N+s>,<NP>
the <A> <N+s>,<NP>
the <N+s>,<NP>
this <A> <N+s>,<NP>
this <N+s>,<NP>
part of the <A> <N+s>,<NP>
part of the <N+s>,<NP>
such a <A> <N+s>,<NP>
such a <N+s>,<NP>
the flood of <A> <N+s>,<NP>
the flood of <N+s>,<NP>
# 24 generated sequences.

C:\Users\Max\RA-Workbench>
```

ra2lst

input:

- an MGFS

output:

- the list of generated sequences

options:

- limit number of loops

```
C:\Windows\System32\cmd.exe

C:\Users\Max\RA-Workbench>type simplifiedictionary2.dic
#use simpleinflection2.nof
the,DET
tsar,N+FLX=APPLE
mount,V+FLX=ASK
drink,N+FLX=APPLE
drink,V+FLX=DRINK
ultra,PREFIX
microscopic,PREFIX
tear,N+FLX=APPLE

C:\Users\Max\RA-Workbench>dic2lst simplifiedictionary2.dic properties.def |
lst2ra simpleinflection2.nof simplifiedictionary2.ra
List successfully compiled into minimal MGFS with 49 states.

C:\Users\Max\RA-Workbench>ra2lst simplifiedictionary2.ra
drinking,V+_LEX="<B3>" + FLX=DRINK + Tense=G
drink,N+_LEX="<E>" + FLX=APPLE + Number=s
drink,V+_LEX="<E>" + FLX=DRINK + Tense=INF
drink,V+_LEX="<E>" + FLX=DRINK + Number=p + Pers=123 + Tense=PR
drinks,N+_LEX="<B>" + FLX=APPLE + Number=p
drinks,V+_LEX="<B>" + FLX=DRINK + Number=s + Pers=3 + Tense=PR
drink,V+_LEX="<E>" + FLX=DRINK + Number=s + Pers=12 + Tense=PR
drank,V+_LEX="<L2><B>i<R2>" + FLX=DRINK + Number=s + Pers=123 + Tense=PT
drank,V+_LEX="<L2><B>i<R2>" + FLX=DRINK + Number=p + Pers=123 + Tense=PT
drunk,V+_LEX="<L2><B>i<R2>" + FLX=DRINK + Tense=PP
ultra,PREFIX
the,DET
tsar,N+_LEX="<E>" + FLX=APPLE + Number=s
tsars,N+_LEX="<B>" + FLX=APPLE + Number=p
tear,N+_LEX="<E>" + FLX=APPLE + Number=s
tears,N+_LEX="<B>" + FLX=APPLE + Number=p
microscopic,PREFIX
mounting,V+_LEX="<B3>" + FLX=ASK + Tense=G
mount,V+_LEX="<E>" + FLX=ASK + Number=s + Pers=12 + Tense=PR
mount,V+_LEX="<E>" + FLX=ASK + Number=p + Pers=123 + Tense=PR
mount,V+_LEX="<E>" + FLX=ASK + Tense=INF
mounts,V+_LEX="<B>" + FLX=ASK + Number=s + Pers=3 + Tense=PR
mounted,V+_LEX="<B2>" + FLX=ASK + Tense=PP
mounted,V+_LEX="<B2>" + FLX=ASK + Number=p + Pers=123 + Tense=PT
mounted,V+_LEX="<B2>" + FLX=ASK + Number=s + Pers=123 + Tense=PT
# 25 generated sequences.

C:\Users\Max\RA-Workbench>
```

segment

inputs:

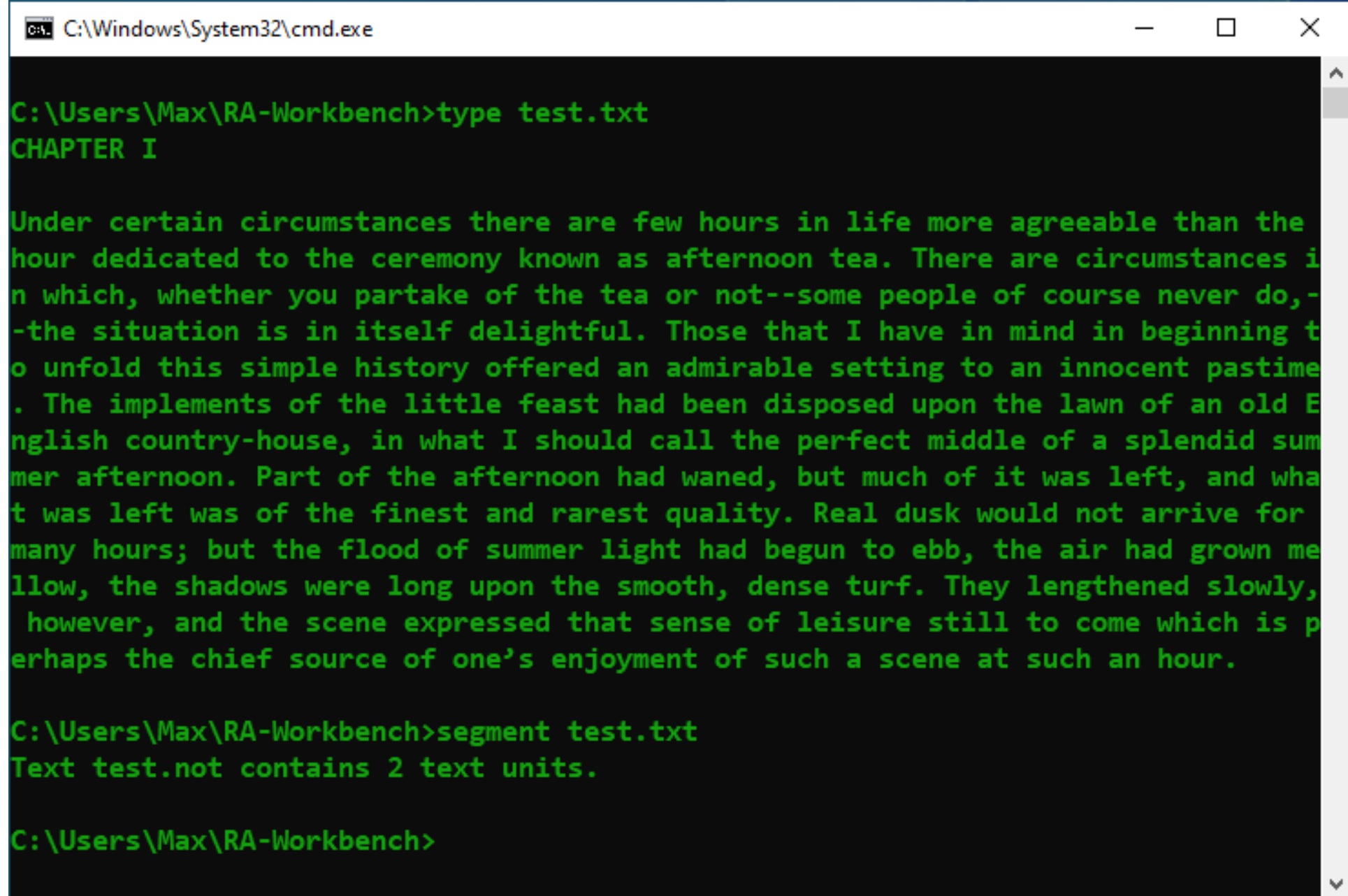
- texts in UTF8

outputs:

- annotated
segmented texts

option:

- a segmentation
orthographic
grammar



```
C:\Windows\System32\cmd.exe

C:\Users\Max\RA-Workbench>type test.txt
CHAPTER I

Under certain circumstances there are few hours in life more agreeable than the
hour dedicated to the ceremony known as afternoon tea. There are circumstances i
n which, whether you partake of the tea or not--some people of course never do,-
-the situation is in itself delightful. Those that I have in mind in beginning t
o unfold this simple history offered an admirable setting to an innocent pastime
. The implements of the little feast had been disposed upon the lawn of an old E
nglish country-house, in what I should call the perfect middle of a splendid sum
mer afternoon. Part of the afternoon had waned, but much of it was left, and wha
t was left was of the finest and rarest quality. Real dusk would not arrive for
many hours; but the flood of summer light had begun to ebb, the air had grown me
llow, the shadows were long upon the smooth, dense turf. They lengthened slowly,
however, and the scene expressed that sense of leisure still to come which is p
erhaps the chief source of one's enjoyment of such a scene at such an hour.

C:\Users\Max\RA-Workbench>segment test.txt
Text test.not contains 2 text units.

C:\Users\Max\RA-Workbench>
```

When no MGFSA is given, segments text paragraph per paragraph

segment

inputs:

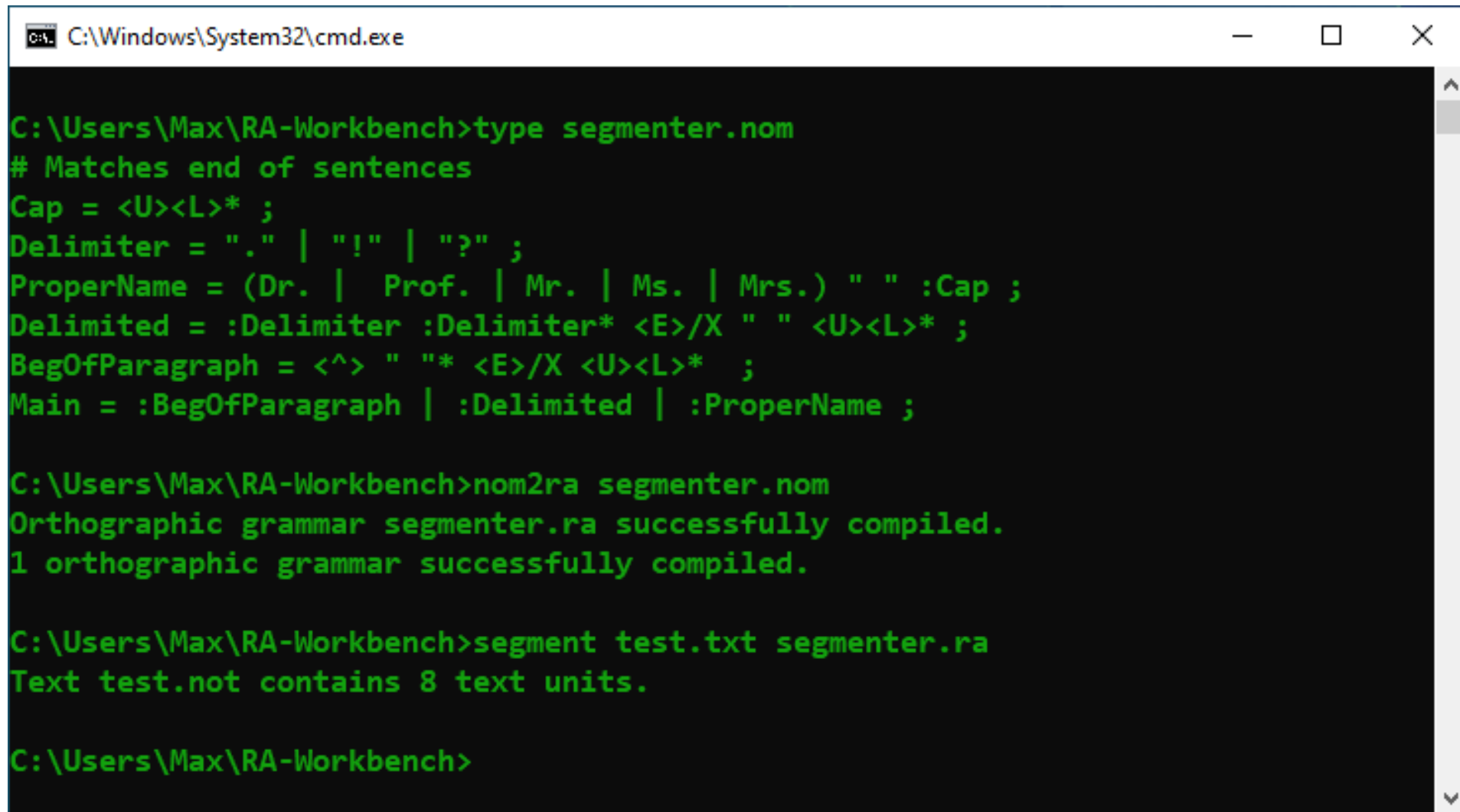
- texts in UTF8

outputs:

- annotated segmented texts

option:

- a segmentation orthographic grammar



```
C:\Windows\System32\cmd.exe

C:\Users\Max\RA-Workbench>type segmenter.nom
# Matches end of sentences
Cap = <U><L>* ;
Delimiter = "." | "!" | "?" ;
ProperName = (Dr. | Prof. | Mr. | Ms. | Mrs.) " " :Cap ;
Delimited = :Delimiter :Delimiter* <E>/X " " <U><L>* ;
BegOfParagraph = <^> " " * <E>/X <U><L>* ;
Main = :BegOfParagraph | :Delimited | :ProperName ;

C:\Users\Max\RA-Workbench>nom2ra segmenter.nom
Orthographic grammar segmenter.ra successfully compiled.
1 orthographic grammar successfully compiled.

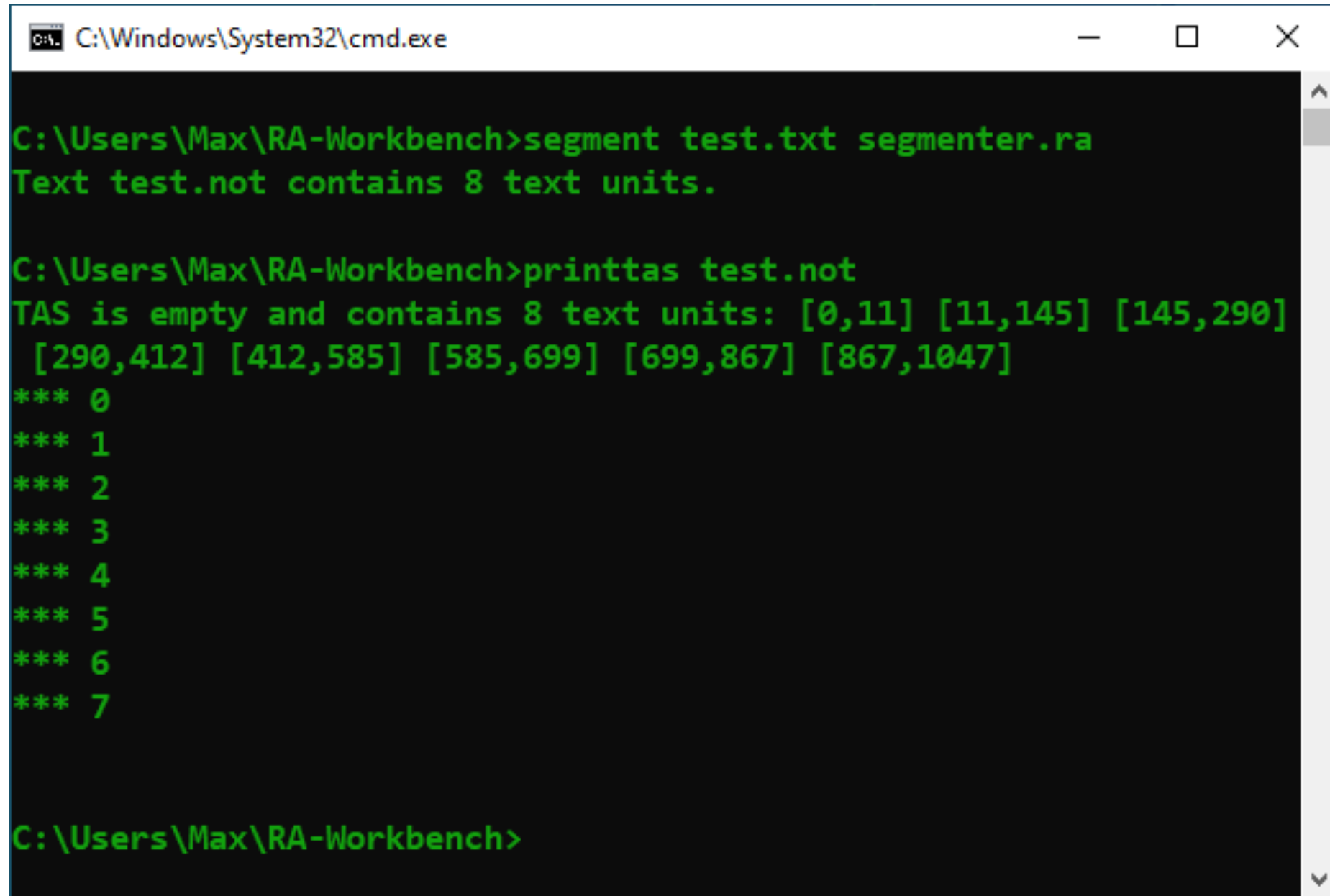
C:\Users\Max\RA-Workbench>segment test.txt segmenter.ra
Text test.not contains 8 text units.

C:\Users\Max\RA-Workbench>
```

Applies a MGFSFA to recognize text units (here: sentences)

printtas

prints a readable
version of a TAS



```
C:\Windows\System32\cmd.exe

C:\Users\Max\RA-Workbench>segment test.txt segmenter.ra
Text test.not contains 8 text units.

C:\Users\Max\RA-Workbench>printtas test.not
TAS is empty and contains 8 text units: [0,11] [11,145] [145,290]
[290,412] [412,585] [585,699] [699,867] [867,1047]
*** 0
*** 1
*** 2
*** 3
*** 4
*** 5
*** 6
*** 7

C:\Users\Max\RA-Workbench>
```

lexicalanalysis (1)

inputs:

- annotated texts
- MGFSAs
- a properties definitions file

outputs

- every text's TAS has been enriched

NOTE: only add annotations at empty positions

```
C:\Windows\System32\cmd.exe

C:\Users\Max\RA-Workbench>type test2.txt
The czar Zzzzist doesn't redismount drinkable ultramicroscopictears.
C:\Users\Max\RA-Workbench>segment test2.txt
Text test2.not contains 1 text unit.

C:\Users\Max\RA-Workbench>lexicalanalysis test2.not simpLEDictionary2.ra contractions.
ra csar.ra multipleprefixes.ra propernameism.ra reverb.ra verbable.ra properties.def
12 added annotations.

C:\Users\Max\RA-Workbench>printtas test2.not
TAS contains 12 annotations in 1 text unit: [0,68]
*** 0
000000 : <the,DET> → 000003
000004 : <tsar,N+Distribution=Hum+Gender=m+Number=s> → 000008
000009 : <Zzzz,N+Distribution=Hum+Number=s> → 000016
000017 : <do,V+Number=s+Pers=3+Tense=PR> → 000017.01
000017.01: <not,ADV> → 000024
000025 : <mount,V+Pref=dis+Pref=re+FLX=ASK+Number=p+Pers=123+Tense=PR> → 000035
000025 : <mount,V+Pref=dis+Pref=re+FLX=ASK+Number=s+Pers=12+Tense=PR> → 000035
000025 : <mount,V+Pref=dis+Pref=re+FLX=ASK+Tense=INF> → 000035
000036 : <drink,A+InitialCat=V+able> → 000045
000046 : <ultra,PREFIX> → 000046.01
000046.01: <microscopic,PREFIX> → 000046.02
000046.02: <tear,N+FLX=APPLE+Number=p> → 000067

C:\Users\Max\RA-Workbench>
```

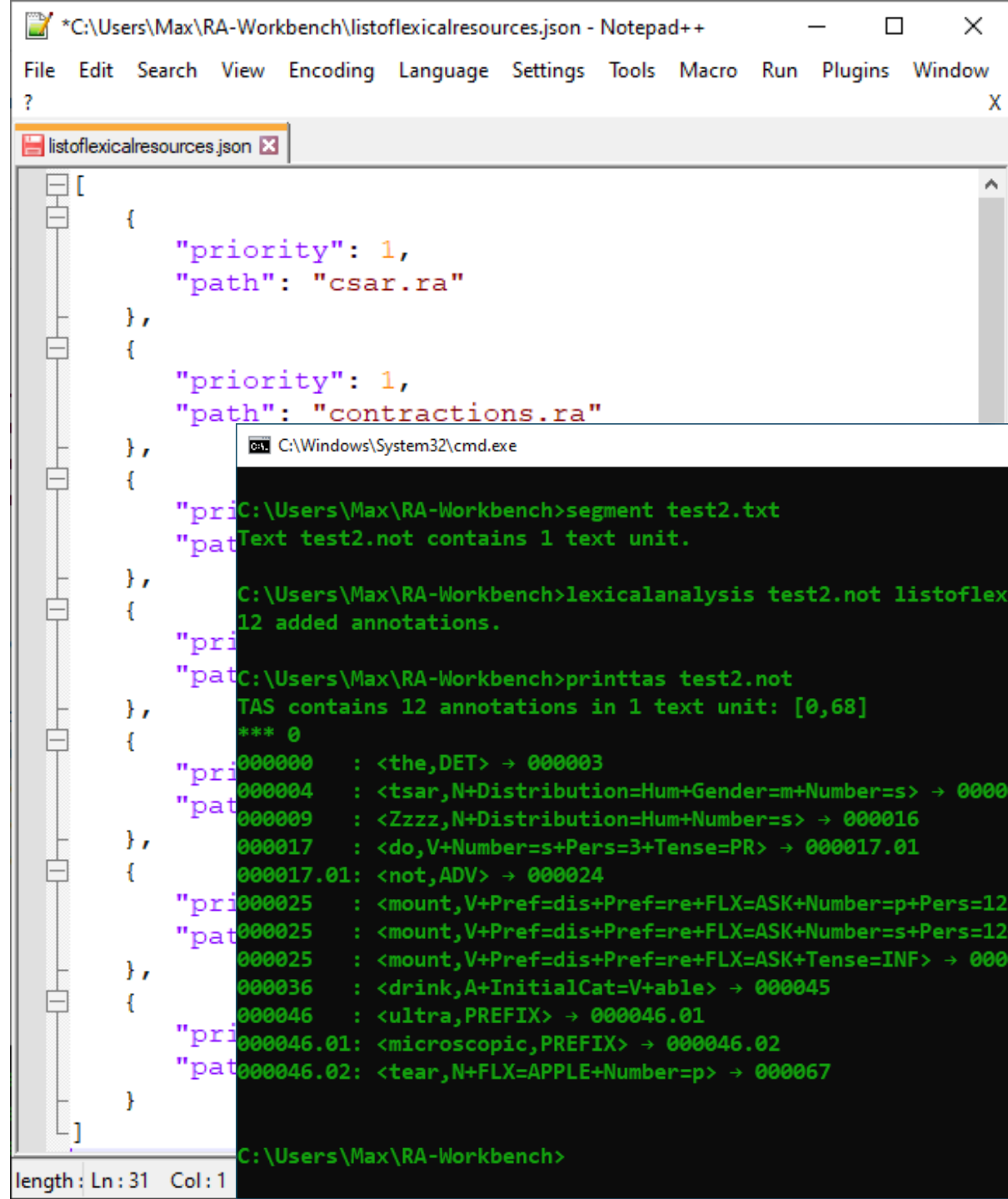
lexicalanalysis (2)

inputs:

- annotated texts
- **a list of resources**
- a properties definitions file

outputs

- the TAS has been enriched



The image shows a Notepad++ window titled '*C:\Users\Max\RA-Workbench\listoflexicalresources.json - Notepad++'. The window contains a JSON array with three objects, each having 'priority' and 'path' fields. The first object has 'priority': 1 and 'path': 'csar.ra'. The second object has 'priority': 1 and 'path': 'contractions.ra'. The third object has 'priority': 1 and 'path': '...'. The window also shows a tree view on the left side.

Overlaid on the bottom right is a Windows command prompt window titled 'C:\Windows\System32\cmd.exe'. It shows the following commands and their outputs:

```
C:\Users\Max\RA-Workbench>segment test2.txt
Text test2.not contains 1 text unit.

C:\Users\Max\RA-Workbench>lexicalanalysis test2.not listoflexicalresources.json properties.def
12 added annotations.

C:\Users\Max\RA-Workbench>printtas test2.not
TAS contains 12 annotations in 1 text unit: [0,68]
*** 0
000000 : <the,DET> → 000003
000004 : <tsar,N+Distribution=Hum+Gender=m+Number=s> → 000008
000009 : <Zzzz,N+Distribution=Hum+Number=s> → 000016
000017 : <do,V+Number=s+Pers=3+Tense=PR> → 000017.01
000017.01: <not,ADV> → 000024
000025 : <mount,V+Pref=dis+Pref=re+FLX=ASK+Number=p+Pers=123+Tense=PR> → 000035
000025 : <mount,V+Pref=dis+Pref=re+FLX=ASK+Number=s+Pers=12+Tense=PR> → 000035
000025 : <mount,V+Pref=dis+Pref=re+FLX=ASK+Tense=INF> → 000035
000036 : <drink,A+InitialCat=V+able> → 000045
000046 : <ultra,PREFIX> → 000046.01
000046.01: <microscopic,PREFIX> → 000046.02
000046.02: <tear,N+FLX=APPLE+Number=p> → 000067

C:\Users\Max\RA-Workbench>
```

syntacticanalysis

inputs:

- annotated texts
- syntactic MGFSAs
- properties definitions file

output

- add annotations to the TAS

```
C:\Windows\System32\cmd.exe

C:\Users\Max\RA-Workbench>type test3.txt
Under certain circumstances there are few hours in life more agreeable than the hour dedicated to the ceremony known as afternoon tea.

C:\Users\Max\RA-Workbench>segment test3.txt
Text test3.not contains 1 text unit.

C:\Users\Max\RA-Workbench>lexicalanalysis test3.not simpledictionary3.ra properties.def
46 added annotations.

C:\Users\Max\RA-Workbench>type np.nog
Main = <E>/<NP
      ( : PLURAL_DET (<A> | <E>) <N+p> |
        : SINGULAR_DET (<A> | <E>) <N+s> )
      <E>/> ;
PLURAL_DET = certain | few | many | the ;
SINGULAR_DET =
  a | an | some | the | this |
  part of the | such a | the flood of ;

C:\Users\Max\RA-Workbench>nog2ra np.nog -minimize
Syntactic grammar np.ra successfully compiled.

C:\Users\Max\RA-Workbench>syntacticanalysis test3.not np.ra properties.def
4 added annotations; 0 removed annotation.

C:\Users\Max\RA-Workbench>printtas test3.not
TAS contains 50 annotations in 1 text unit: [0,134]
*** 0
000000 : <under,A+Human> → 000005
000000 : <under,ADV> → 000005
000000 : <under,PREFIX> → 000005
000000 : <under,PREFIX> → 000005
000006 : <NP> → 000027
000006 : <under,A+Human> → 000013
000006 : <certain,DET> → 000013
000014 : <circumstance,N+Distribution=Abst+FLX=APPLE+Number=p> → 000027
000028 : <there,ADV> → 000033
000028 : <there,INTJ> → 000033
000028 : <there,PRO+Number=p+Pers=3> → 000033
000028 : <there,PRO+Number=s+Pers=3> → 000033
000034 : <be,V+Auxiliary+FLX=BE+Number=p+Pers=123+Tense=PR> → 000037
000034 : <be,V+Auxiliary+FLX=BE+Number=s+Pers=2+Tense=PR> → 000037
000038 : <NP> → 000047
```

syntacticanalysis

inputs:

- annotated texts
- syntactic MGFSAs
- properties definitions file

output

- remove annotations from the TAS

```
C:\Windows\System32\cmd.exe
C:\Users\Max\RA-Workbench>type test2.txt
The czar Zzzzist doesn't redismount drinkable ultramicroscopictears.
C:\Users\Max\RA-Workbench>segment test2.txt
Text test2.not contains 1 text unit.

C:\Users\Max\RA-Workbench>lexicalanalysis test2.not listoflexicalresources.json properties.def
12 added annotations.

C:\Users\Max\RA-Workbench>printtas test2.not
TAS contains 12 annotations in 1 text unit: [0,68]
*** 0
000000 : <the,DET> → 000003
000004 : <tsar,N+Distribution=Hum+Gender=m+Number=s> → 000008
000009 : <Zzzz,N+Distribution=Hum+Number=s> → 000016
000017 : <do,V+Number=s+Pers=3+Tense=PR> → 000017.01
000017.01 : <not,ADV> → 000024
000025 : <mount,V+Pref=dis+Pref=re+FLX=ASK+Number=p+Pers=123+Tense=PR> → 000035
000025 : <mount,V+Pref=dis+Pref=re+FLX=ASK+Number=s+Pers=12+Tense=PR> → 000035
000025 : <mount,V+Pref=dis+Pref=re+FLX=ASK+Tense=INF> → 000035
000036 : <link,A+InitialCat=V+able> → 000045
000046 : <ultra,PREFIX> → 000046.01
000046.01 : <microscopic,PREFIX> → 000046.02
000046.02 : <tear,N+FLX=APPLE+Number=p> → 000067

C:\Users\Max\RA-Workbench>type disamb.nog
Main = <do> <ADV> <V>/<V+INF> ;

C:\Users\Max\RA-Workbench>nog2ra disamb.nog -minimize
Syntactic grammar disamb.ra successfully compiled.

C:\Users\Max\RA-Workbench>syntacticanalysis test2.not disamb.ra properties.def
0 added annotation; 2 removed annotations.

C:\Users\Max\RA-Workbench>printtas test2.not
TAS contains 10 annotations in 1 text unit: [0,68]
*** 0
000000 : <the,DET> → 000003
000004 : <tsar,N+Distribution=Hum+Gender=m+Number=s> → 000008
000009 : <Zzzz,N+Distribution=Hum+Number=s> → 000016
000017 : <do,V+Number=s+Pers=3+Tense=PR> → 000017.01
000017.01 : <not,ADV> → 000024
000025 : <mount,V+Pref=dis+Pref=re+FLX=ASK+Tense=INF> → 000035
000036 : <link,A+InitialCat=V+able> → 000045
000046 : <ultra,PREFIX> → 000046.01
000046.01 : <microscopic,PREFIX> → 000046.02
000046.02 : <tear,N+FLX=APPLE+Number=p> → 000067

C:\Users\Max\RA-Workbench>
```

syntacticlocate

inputs:

- a list of texts
- one syntactic query'

MGFSA

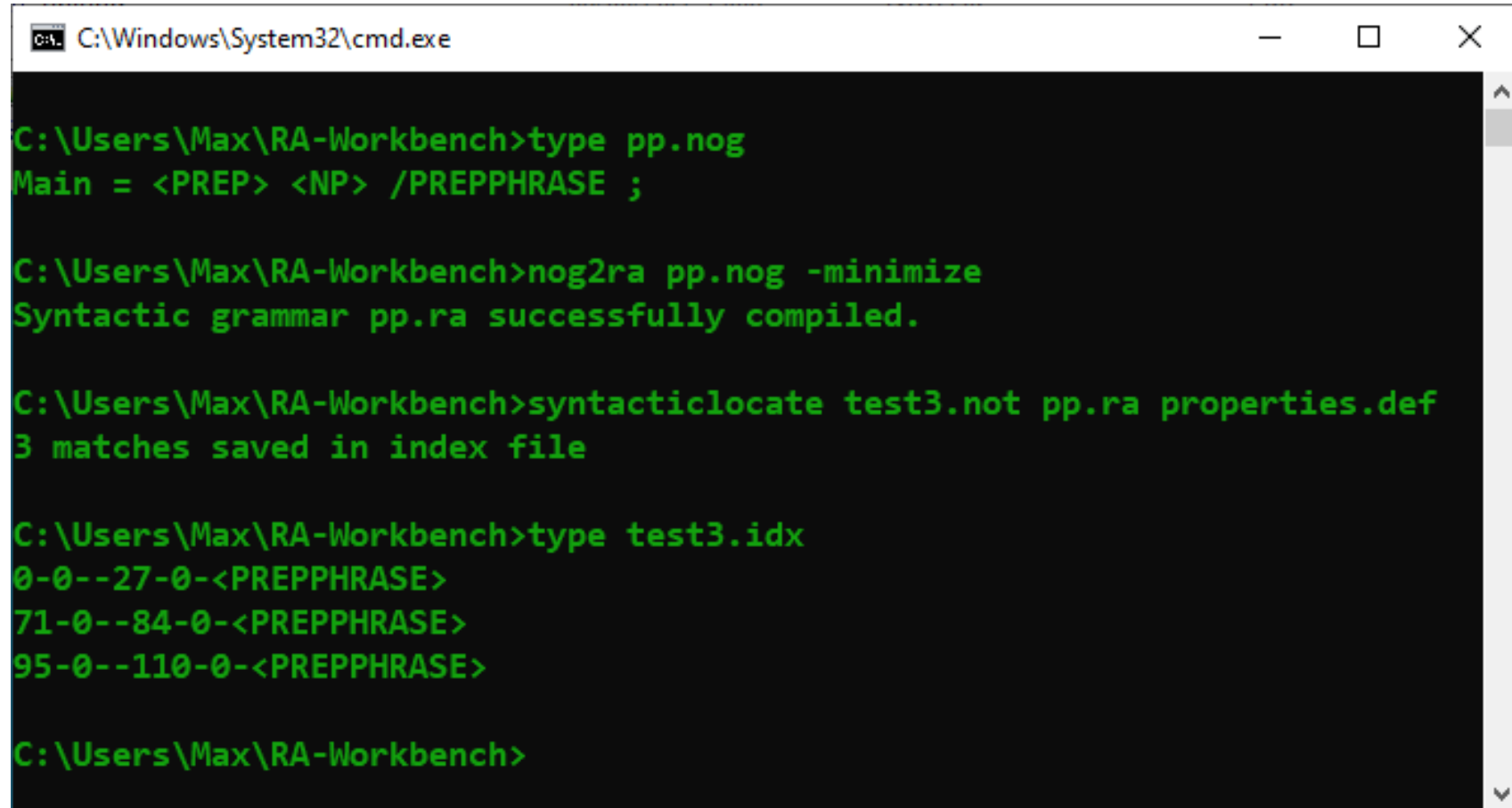
- a properties definitions file

option:

- shortest, longest, all

outputs:

- the index files that contains the matches in every text



```
C:\Windows\System32\cmd.exe

C:\Users\Max\RA-Workbench>type pp.nog
Main = <PREP> <NP> /PREPPHRASE ;

C:\Users\Max\RA-Workbench>nog2ra pp.nog -minimize
Syntactic grammar pp.ra successfully compiled.

C:\Users\Max\RA-Workbench>syntacticlocate test3.not pp.ra properties.def
3 matches saved in index file

C:\Users\Max\RA-Workbench>type test3.idx
0-0--27-0-<PREPPHRASE>
71-0--84-0-<PREPPHRASE>
95-0--110-0-<PREPPHRASE>

C:\Users\Max\RA-Workbench>
```

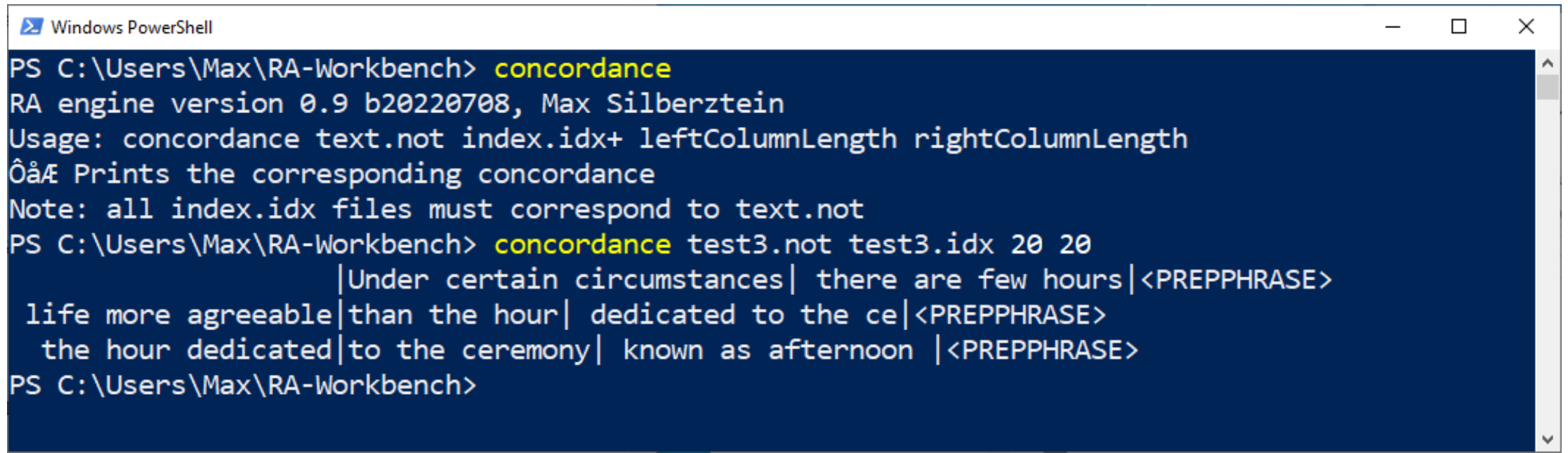
concordance

inputs:

- a text, e.g.: test.not
- a list of indices, e.g.: test1.idx, test2.idx, test3.idx
- left and right columns' lengths, e.g.: 20 20

output:

- the concordance



```
Windows PowerShell
PS C:\Users\Max\RA-Workbench> concordance
RA engine version 0.9 b20220708, Max Silberztein
Usage: concordance text.not index.idx+ leftColumnLength rightColumnLength
& Prints the corresponding concordance
Note: all index.idx files must correspond to text.not
PS C:\Users\Max\RA-Workbench> concordance test3.not test3.idx 20 20
      |Under certain circumstances| there are few hours|<PREPPHRASE>
life more agreeable|than the hour| dedicated to the ce|<PREPPHRASE>
the hour dedicated|to the ceremony| known as afternoon |<PREPPHRASE>
PS C:\Users\Max\RA-Workbench>
```


Perspective

- Aiming at v1.0: please send **feedback**!
- Formalize discontinuous expressions (e.g. phrasal verbs) in a easier way than NooJ's
- Implement a better transformational engine than NooJ's
- Adapt NooJ resources for the 30+ supported languages
- Add a graphical interface (WEB?)
- Add ATISHS' statistical functionalities, see <http://atish.univ-fcomte.fr>
- Collaborative... **Collaborations**, anyone?