ex8

May 16, 2020

```
[1]: #!pip install eniam
     from eniam import *

     # Ex. 1
     result = Eniam(
     # Custom atomic symbols in lexicon
     [],
     # Custom lexicon rules
     {
         rule(lemma='kot',pos=subst,case=nom):       nom,
         rule(lemma='gonić',pos=fin,person=ter):     (ip<nom)>acc,
         rule(lemma='mysz',pos=subst,case=acc):      acc,
         root_rule():                                s % ip,
     },
     # Custom valence phrases
     ['KOT', 'MYSZ', 'ZDARZENIE'],
     # Valence rules
     {
         valence_rule('kot', 'noun'): 'KOT',
         valence_rule('mysz', 'noun'): 'MYSZ',
         valence_rule('gonić', 'verb'): 'ZDARZENIE',
     }).dom("Kot goni mysz.")

     #result.show()
     result.save_html('ex1')
```

```
[1]: <eniam.eniam_core.DomparserResult at 0x10c4fafd0>
```

```
[2]: # Ex. 2
     from eniam import *
     result = Eniam(
         [],
         {
             rule(lemma='kot',pos=subst):       np*nom,
             rule(lemma='mysz',pos=subst):      np*acc,
             rule(lemma='gonić',pos=fin):       (ip<(np*nom))>(np*acc),
             root_rule():                       s % ip,
         },
         ['KOT', 'MYSZ', 'ZDARZENIE'],
         {
             valence_rule('kot', 'noun'): 'KOT',
             valence_rule('mysz', 'noun'): 'MYSZ',
             valence_rule('gonić', 'verb'): 'ZDARZENIE',
         }
     ).dom("Kot goni mysz.")

     #result.show()
     result.save_html('ex2')
```

```
[2]: <eniam.eniam_core.DomparserResult at 0x10c532e10>
```

[3]:

```
# Ex. 3
from eniam import *
result = Eniam(
    [],
    {
        rule(lemma='kot',pos=subst):       np*nom,
        rule(lemma='mysz',pos=subst):      quant(np*case, case=[nom, acc]),
        rule(lemma='gonić',pos=fin):       (ip<(np*nom))>(np*acc),
        root_rule():                       s % ip,
    },
    ['KOT', 'MYSZ', 'ZDARZENIE'],
    {
        valence_rule('kot', 'noun'): 'KOT',
        valence_rule('mysz', 'noun'): 'MYSZ',
        valence_rule('gonić', 'verb'): 'ZDARZENIE',
    }
).dom("Kot goni mysz.")

#result.show()
result.save_html('ex3')
```

```
[3]: <eniam.eniam_core.DomparserResult at 0x10ca9bfd0>
```

[4]:

```
# Ex. 5
from eniam import *
result = Eniam(
    [],
    {
        rule(lemma='kot',pos=subst,case=nom):       np*sg*nom,
        rule(lemma='mysz',pos=subst,case=acc):      np*sg*acc,
        rule(lemma='gonić',pos=fin,person=ter):     (ip<(np*sg*nom))>(np*T*acc),
        root_rule():                                s % ip,
    },
    ['KOT', 'MYSZ', 'ZDARZENIE'],
    {
        valence_rule('kot', 'noun'): 'KOT',
        valence_rule('mysz', 'noun'): 'MYSZ',
        valence_rule('gonić', 'verb'): 'ZDARZENIE',
    }
).dom("Kot goni mysz.")

#result.show()
result.save_html('ex5')
```

[4]: <eniam.eniam_core.DomparserResult at 0x10c4eaa50>

[5]:
```
# Ex. 6
from eniam import *
result = Eniam(
    [],
    {
        rule(lemma='kot',pos=subst,case=nom):       nom,
        rule(lemma='mysz',pos=subst,case=acc):      acc,
        rule(lemma='gonić',pos=fin,person=ter):     ip[nom | acc],
        root_rule():                                s % ip,
    },
    ['KOT', 'MYSZ', 'ZDARZENIE'],
    {
        valence_rule('kot', 'noun'): 'KOT',
        valence_rule('mysz', 'noun'): 'MYSZ',
        valence_rule('gonić', 'verb'): 'ZDARZENIE',
    }
).dom("Kot goni mysz.")

#result.show()
result.save_html('ex6')
```

[5]: <eniam.eniam_core.DomparserResult at 0x10caa8f90>

[6]:
```
# Ex. 7
from eniam import *
result = Eniam(
    [],
    {
        rule(lemma='łódź',pos=subst):        quant(case, case=[nom, acc]),
        rule(lemma='statek',pos=subst):      quant(case, case=[nom, acc]),
        rule(lemma='wyprzedzać',pos=fin):    ip[nom | acc],
        root_rule():                         s % ip,
    },
    ['L', 'S', 'W'],
    {
        valence_rule('Łódź', 'noun'): 'L',
        valence_rule('statek', 'noun'): 'S',
        valence_rule('wyprzedzać', 'verb'): 'W',
    }
).dom("Łódź wyprzedza statek.")

#result.show()
result.save_html('ex7')
```

```
[6]: <eniam.eniam_core.DomparserResult at 0x10c536ed0>
```

[7]:

```
# Ex. 10
from eniam import *
result = Eniam(
    [],
    {
        rule(lemma='kot', pos=subst):               nom,
        rule(lemma='który',pos=apron):              (cp>(ip<nom))>nom,
        rule(lemma='gonić', pos=fin):               (ip<nom)>acc,
        rule(lemma='mysz', pos=subst, case=acc):    acc,
        root_rule():                                s < cp,
    },
    ['KOT', 'MYSZ', 'ZDARZENIE'],
    {
        valence_rule('kot', 'noun'): 'KOT',
        valence_rule('mysz', 'noun'): 'MYSZ',
        valence_rule('gonić', 'verb'): 'ZDARZENIE',
    }
).dom("Który kot goni mysz.")

#result.show()
result.save_html('ex10')
```

```
[7]: <eniam.eniam_core.DomparserResult at 0x10cad2f90>
```

[8]:

```
# Ex. 11
from eniam import *
result = Eniam(
    [],
    {
        rule(lemma='kot', pos=subst):               np*sg*nom,
        rule(lemma='gonić', pos=fin):              ␣
 ↪((ip*sg)<(np*T*acc))<(np*sg*nom),
        rule(lemma='co', pos=comp):                ␣
 ↪quant((cp>((ip*T)<(np*sg*case))), case=[nom, acc]),
        root_rule():                                s % (ip+cp),
    },
    ['KOT', 'ZDARZENIE'],
    {
        valence_rule('kot', 'noun'): 'KOT',
        valence_rule('gonić', 'verb'): 'ZDARZENIE',
    }
).dom("Co kot goni.")

#result.show()
result.save_html('ex11')
```

[8]: <eniam.eniam_core.DomparserResult at 0x10c536450>

[9]:

```
# Ex. 12
from eniam import *

result = Eniam(
    [],
    {
        rule(lemma='każdy', pos=subst, case=nom):    (np*nom) > (n*nom),
        rule(lemma='słoń', pos=subst):               n*nom,
        rule(lemma='trąbić', pos=fin):               ip < (np*nom),
        root_rule():                                 s < ip
    },
    ['SŁOŃ', 'ZDARZENIE', 'KWANTYFIKATOR'],
    {
        valence_rule('każdy', 'noun'):  'KWANTYFIKATOR',
        valence_rule('słoń', 'noun'):   'SŁOŃ',
        valence_rule('trąbić', 'verb'): 'ZDARZENIE',
    }
).dom("Każdy słoń trąbi.")

#result.show()
result.save_html('ex12')
```

[9]: <eniam.eniam_core.DomparserResult at 0x10caa0fd0>

[10]:

```
# Ex. 13
from eniam import *

result = Eniam(
    [],
    {
        rule(lemma='każdy',pos=subst,case=nom):     (np*nom) > (n*nom),
        rule(lemma='chłopak', pos=subst):           n*nom,
        rule(lemma='lubić', pos=fin):               ip[(np*nom) | (np*acc)],
        rule(lemma='dziewczyna', pos=subst):        np*acc,
        root_rule():                                        s < ip
    },
    ['OSOBA', 'ZDARZENIE', 'KWANTYFIKATOR'],
    {
        valence_rule('każdy', 'noun'):       'KWANTYFIKATOR',
        valence_rule('chłopak', 'noun'):     'OSOBA',
        valence_rule('dziewczyna', 'noun'):  'OSOBA',
        valence_rule('lubić', 'verb'):       'ZDARZENIE',
    }
).dom("Każdy chłopak lubi dziewczynę.")

#result.show()
result.save_html('ex13')
```

```
[10]: <eniam.eniam_core.DomparserResult at 0x109396f90>
```

[11]:

```
# Ex. 14
from eniam import *
result = Eniam(
    [],
    {
        'lemma=nie,pos=ppron3':                            ip>ip,
        rule(lemma='przerwać', pos=praet, person=pri):   ip>acc,
        rule(lemma='praca', pos=subst, case=acc):        acc,
        rule(lemma='praca', pos=subst, case=gen):        gen,
        root_rule():                                       s%ip,
    },
    ['CZYNNOŚĆ', 'ZDARZENIE'],
    {
        valence_rule('praca', 'noun'):      'CZYNNOŚĆ',
        valence_rule('przerwać', 'verb'):   'ZDARZENIE',
    }
).dom("Przerwał pracę.")

#result.show()
result.save_html('ex14')
```

```
[11]: <eniam.eniam_core.DomparserResult at 0x10ca9b610>
```

[12]:

```
# Ex. 15
from eniam import *
result = Eniam(
    [],
    {
        'lemma=nie,pos=ppron3':                            ip>ip,
        rule(lemma='przerwać', pos=praet, person=pri):   ip>acc,
        rule(lemma='jednorożec', pos=subst, case=gen):   gen,
        root_rule():                                     s%ip,
    },
    ['CZYNNOŚĆ', 'ZDARZENIE'],
    {
        valence_rule('praca', 'noun'):      'CZYNNOŚĆ',
        valence_rule('przerwać', 'verb'):   'ZDARZENIE',
    }
).dom("Nie przerwał pracy.")

#result.show()
result.save_html('ex15')
```

[12]:

```
<eniam.eniam_core.DomparserResult at 0x10caa1610>
```

[ ]: