Freie Universität Berlin

TEXTDATEN ROMANISTIK

Arbeiten mit Regulären Ausdrücken

Annette Gerstenberg

Würzburg, 16. März 2016
Workshop "Digitale Methoden"
beim Forum Junge Romanistik

Warum RegEx? Ziele!

- Beispiel 1: Bereinigen eines Transkripts, Entfernen von Annotationen
- Beispiel 2: Transformieren eines Transkripts, Wechsel des Programms

Beispiel 1: Vom Transkript zur Frequenzanalyse

Transcriber

Transcriber 1.5.1

File Edit Signal Segmentation Options Help

A01

[tss] [ins] avec une association euh depuis faire des voyages

donc comme je pars toute seule je me euh

INT

et vous allez où ?

AntConc

AntConc 3.4.3w (Windows) 2014

File Global Settings Tool Preferences Help

Corpus Files

A01a_nt.txt

Concordance | Concordance Plot | File View | Clusters/N-Grams | Collocates | Word List

Word Types: 411 **Word Tokens:** 1788 **Search Hits:** 0

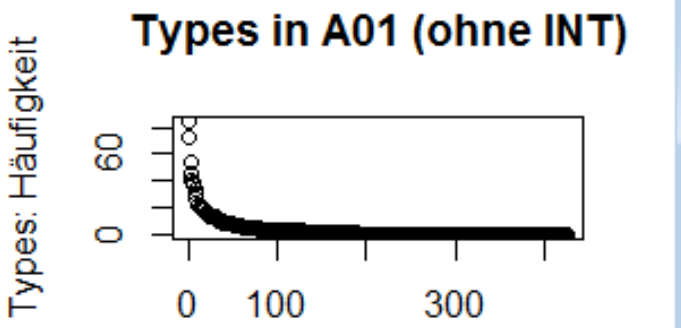
Rank	Freq	Word
1	84	euh
2	73	j
3	54	ai
4	45	de
5	41	et
6	41	oui
7	39	à

```
A01:  [tss] [ins] avec une association
INT:  et vous allez où ?
A01:  oh beh si vous voulez je vous dis
INT:  [rir]
A01:  ah oui je suis allée [ins] je sui
INT:  #2 en Chine ? [rir] #
A01:  oui je suis allée euh pff en Birm
INT:  m-hm
A01:  vous voyez oui vous connaissez la
INT:  #2 un peu oui [
A01:  ah et puis les
INT:  à Cologne vous
```

oui

TextPad

R Graphics: Device 2 (ACTIVE)



R

Beispiel 1: Bereinigen von Annotationen, z.B. für R

```
A01:  [tss] [ins] avec une association
INT:  et vous allez où ?
A01:  oh beh si vous voulez je vous dis
INT:  [rir]
A01:  ah oui je suis allée [ins] je sui
INT:  #2 en Chine ? [rir] #
A01:  oui je suis allée euh pff en Birm
INT:  m-hm
A01:  vous voyez oui vous connaissez la
INT:  #2 un peu oui [rir] #
A01:  ah et puis les pays d' Europe oui
INT:  à Cologne vous avez ?
```

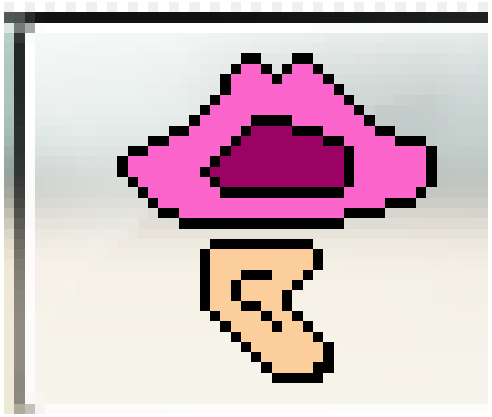
Beispiel 2: Transformation C-ORAL-ROM-Transkript

XML (WinPitch)

```
<?xml version="1.0" encoding="iso-8859-1"?>

<!DOCTYPE Alignment SYSTEM "coralrom.dtd">

<Alignment>

<TimeStamp Value="Monday, October 07, 2002 time 10h 32m 11s"/>

<WinPitch Program="Aligner"  Version="1.0"/>
<Trans version="1.0" creationDate="Monday, October 07, 2002
time 10h 32m 11s"   audioFilename="EFAMMN01.WAV"
textFilename="efammn01.txt"/>
<Layer1 Name="HIS" ID="HIS" Short="HIS" Color="RGB
(145,213,110)"/>
<Layer2 Name="Layer 2" ID="layer2" Short="L 2" Color="RGB
(213,145,220)"/>
<Layer3 Name="Layer 3" ID="layer3" Short="L 3" Color="RGB
(145,145,213)"/>
<Layer4 Name="Layer 4" ID="layer4" Short="L 4" Color="RGB
(228,228,228)"/>
<Layer5 Name="Layer 5" ID="layer5" Short="L 5" Color="RGB
(200,213,180)"/>
<Layer6 Name="Layer 6" ID="layer6" Short="L 6" Color="RGB
(213,222,140)"/>
<Layer7 Name="Layer 7" ID="layer7" Short="L 7" Color="RGB
(120,195,200)"/>
<Layer8 Name="Layer 8" ID="layer8" Short="L 8" Color="RGB
(255,228,255)"/>

<UNIT speaker="HIS" startTime="0.000" endTime="3.288"
Channel="M">que / éramos / cuatro mujeres / como te digo y
cuatro hombres //</UNIT>
```

TEXTGRID (Praat)

```
File type = "ooTextFile"
Object class = "TextGrid"

xmin = 0.0
xmax = 996.593
tiers? <exists>
size = 1
item []:
    item [1]:
        class = "IntervalTier"
        name = "HIS"
        xmin = 0.000
        xmax = 2006.459
        intervals: size = 490
        intervals [1]:
            xmin = 0.000
            xmax = 3.288
            text = "que / éramos / cuatro mujeres / como te digo
        intervals [2]:
            xmin = 3.288
            xmax = 4.592
            text = "ya yo tenía a mi hermano //"
        intervals [3]:
            xmin = 4.592
            xmax = 5.732
            text = "para mí //"
```



Beispiel 2: Transformation Transkript Kodierung Turnlänge

Ausgangsformat: XML, Kodierung für WinPitch (DVD C-ORAL-ROM)

```
<UNIT speaker="HIS" startTime="0.000" endTime="3.288"
Channel="M">que / éramos / cuatro mujeres / como te
digo y cuatro hombres //</UNIT>
```

Zielformat: TextGrid, kann in Praat bearbeitet werden

```
intervals [1]:
            xmin = 0.000
            xmax = 3.288
            text = "que / éramos / cuatro mujeres /
como te digo y cuatro hombres //"
```

Ziel erreicht: C-ORAL-ROM-Dateien in Praat öffnen

Übersicht: Reguläre Ausdrücke
Regular Expressions / RegEx

- Definition: **Die Kunst zu finden, was man nicht gesucht hat**
- Grundlagen: Zeichenklassen, Gruppieren, Ersetzen
- Beispiele
  - Fragestellungen
  - Anwendungen
  - Auswertungen ... R

„Ein *regulärer Ausdruck*, oft auch als *Suchmuster* bezeichnet, ist eine Schablone, die auf einen gegebenen String passt oder nicht passt. Haben wir also eine unendlich große Anzahl von Strings, so teilt das gegebene Muster diese in zwei Gruppen: diejenigen, die passen, und diejenigen, die nicht passen.

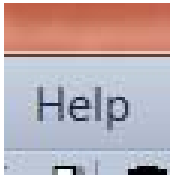
Ein Muster kann hierbei auf einen möglichen String passen oder auf zwei oder hundert oder sogar auf eine unendliche Anzahl. Oder ein Muster passt auf alle möglichen Strings, *bis auf* einen. Oder auch auf eine unendliche Anzahl.“

Schwartz / Phoenix / Lang 2005: 112s.

Reguläre Ausdrücke: Muster statt Inhalt

- Komplexe Suche- + Ersetze-Prozesse
- Suche nach Mustern statt nach buchstäblich bekannten Okkurrenzen
- Definieren von Gruppen und Umstellen solcher Gruppen
- Eingebunden in viele Programme
  (Text- und XML-Editoren, Python, Perl)

Reguläre Ausdrücke: Vorsichtsmaßnahmen & Nebenwirkungen



- Trial and Error: Erhöhte Frustrationstoleranz
- Zahlreiche Varianten, "Hilfe-Fenster" gleich offen lassen
- Nebenwirkung: Abstraktionsprozesse wirken schon bei der Datenhaltung
  - Wie kann ich die Daten so strukturieren, dass ich klare Such- und Austauschanweisungen formulieren kann?

Zeichen und Zeichenklassen

- Literale Strings (Zeichenfolgen)

`Suchzeichenfolge`

- Klassenausdrücke
- Metazeichen
- Logische Operatoren

Klassenausdrücke

- Auswahl von Zeichengruppen: in eckigen Klammern
  - Lettern des Alphabets, Zahlen, Diakritika, ...

```
[aeiou][13579][èéêèë] [ìíîï]
```

  - Bereiche

```
[A-Z]    [a-z]  [0-9]
```

- Ausschließen von Zeichengruppen: eckige Klammern und ^

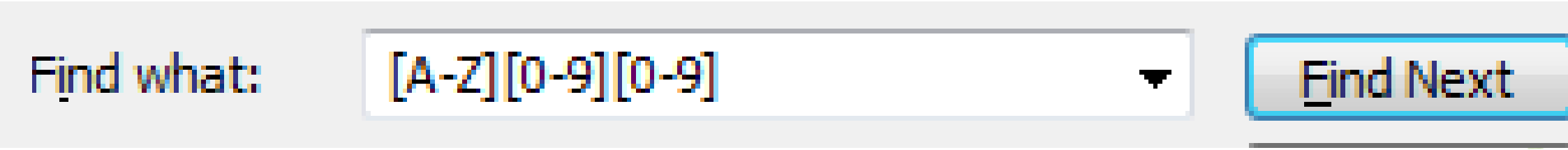
```
[^ë,ï]
```

Übung Zeichenklassen: Entferne Sprecherkodierung A01

```
A01:   [tss] [ins] avec une association
INT:   et vous allez où ?
A01:   oh beh si vous voulez je vous dis
INT:   [rir]
A01:   ah oui je suis allée [ins] je sui
INT:   #2 en Chine ? [rir] #
A01:   oui je suis allée euh pff en Birm
INT:   m-hm
A01:   vous voyez oui vous connaissez la
INT:   #2 un peu oui [rir] #
A01:   ah et puis les pays d' Europe oui
INT:   à Cologne vous avez ?
```

Datei: A01a.txt

Funktionstaste (TextPad) Suchen & Ersetzen: F8
Haken bei "Regular Expressions" (Mac: Grep)



Zeichen: Metazeichen, Überblick

- Außer in Klassenausdrücken: Punkt, eckige Klammer, Backslash
  `.  [  \` ... hängt von der Programmversion ab!
- Außer in Klassenausdrücken, 1. Position RegEx oder Muster: Stern, Fragezeichen, Plus
  `*  ?  +`
- Nur in Klassenausdruck (außer 1. oder letzte Position): Minus
  `-`
- Erste Position RegEx oder Klassenausdruck: Caret ("Dach")
  `^`
- Letzte Position RegEx: Dollar
  `$`

Zeichen: Metazeichen und literale Zeichen

- Klassenausdrücke in eckigen Klammern

```
[a-z]
```

- Suche nach Text, der in eckigen Klammern steht: Backslash

```
\[ins\]
```

- Kombination von Klassenausdrücken und literalen Zeichen?

Übung Metazeichen: Entferne "events"

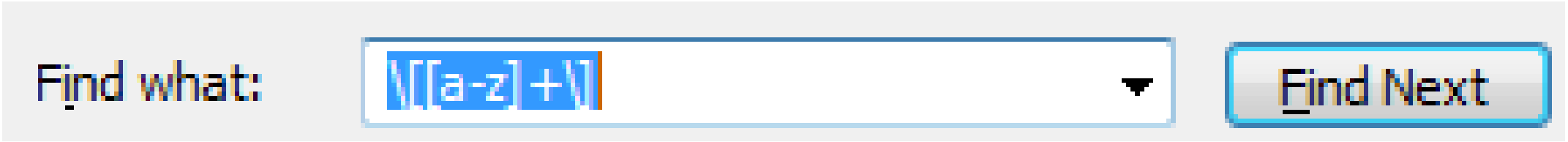
```
A01:  [tss] [ins] avec une association
INT:  et vous allez où ?
A01:  oh beh si vous voulez je vous dis
INT:  [rir]
A01:  an oui je suis allée [ins] je sui
INT:  #2 en Chine ? [rir] #
A01:  oui je suis allée euh pff en Birm
INT:  m-hm
A01:  vous voyez oui vous connaissez la
INT:  #2 un peu oui [rir] #
A01:  ah et puis les pays d' Europe oui
INT:  à Cologne vous avez ?
```

Zeichen: Metazeichen, Quantifikatoren
Bezug auf kleinsten vorausgehenden RegEx

- Asterisk *
  - Kein Auftreten, einmaliges oder mehrmaliges Auftreten
  - `toto*` findet `tot, toto, totoo, totooo`, etc.
- Fragezeichen ?
  - Kein- oder einmaliges Auftreten
  - `toto?` findet `tot, toto`, etc.
- Plus +
  - Ein- oder mehrmaliges Auftreten
  - `toto+` findet `toto, totoo, totooo`, etc.

Übung Quantifikatoren 1: Entferne "events"

```
A01:  [tss] [ins] avec une association
INT:  et vous allez où ?
A01:  oh beh si vous voulez je vous dis
INT:  [rir]
A01:  an oui je suis allée [ins] je sui
INT:  #2 en Chine ? [rir] #
A01:  oui je suis allée euh pff en Birm
INT:  m-hm
A01:  vous voyez oui vous connaissez la
INT:  #2 un peu oui [rir] #
A01:  ah et puis les pays d' Europe oui
INT:  à Cologne vous avez ?
```



Zeichen: Metazeichen, Quantifikatoren
Bezug auf kleinsten vorausgehenden RegEx

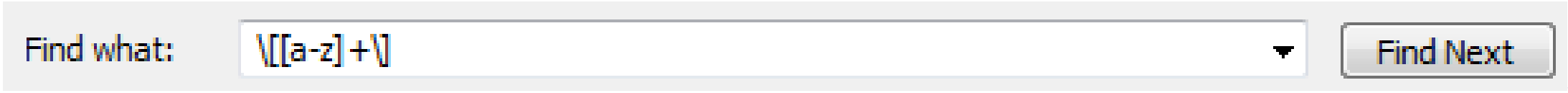
- Alternativ-Operator |
  - Zutreffen auf linke oder rechte Seite, niedrige Priorität (Gruppierung)

  `Suchmuster1|Suchmuster2`
- Bestimmte Wiederholungszeichen
  - Anzahl, Mindesthäufigkeit, Mindest- und Maximalhäufigkeit

  `\{Anzahl\} \{min,\}  \{min,max\}`

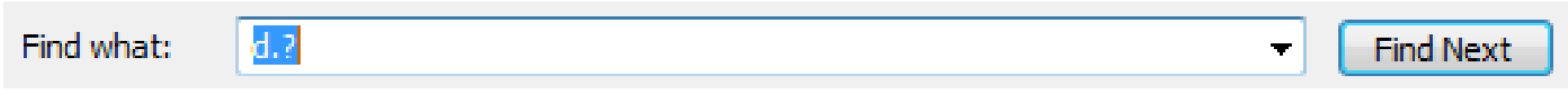
Übung Quantifikatoren 2: Entferne "events"

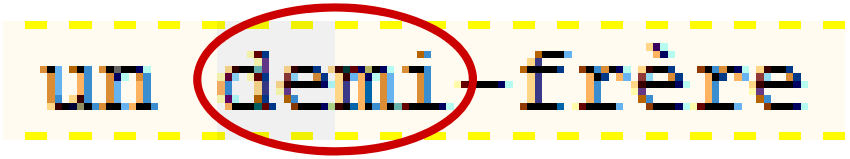
```
A01:  [tss] [ins] avec une association
INT:  et vous allez où ?
A01:  oh beh si vous voulez je vous dis
INT:  [rir]
A01:  ah oui je suis allée [ins] je sui
INT:  #2 en Chine ? [rir] #
A01:  oui je suis allée euh pff en Birm
INT:  m-hm
A01:  vous voyez oui vous connaissez la
INT:  #2 un peu oui [rir] #
A01:  ah et puis les pays d' Europe oui
INT:  à Cologne vous avez ?
```



Übung Quantifikator 3: Gesucht wird frz. *de* oder *des*

```
A01:   [tss] [ins] avec une association
INT:   et vous allez où ?
A01:   oh beh si vous voulez je vous dis
INT:   [rir]
A01:   ah oui je suis allée [ins] je sui
INT:   #2 en Chine ? [rir] #
A01:   oui je suis allée euh pff en Birm
INT:   m-hm
A01:   vous voyez oui vous connaissez la
INT:   #2 un peu oui [rir] #
A01:   ah et puis les pays d' Europe oui
INT:   à Cologne vous avez ?
```





Problem... das war nicht gesucht!

Lösung: Positionsangaben

- Zeilenanfang, Anfang RegEx: Caret ("Dach")

  ^
- Zeilenende, Ende RegEx: Dollar

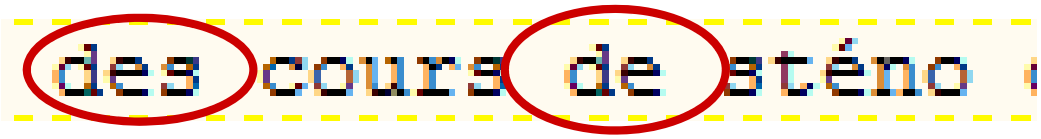
  $
- Wortanfang:

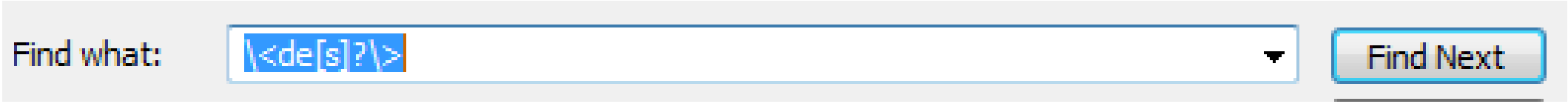
  \<
- Wortende

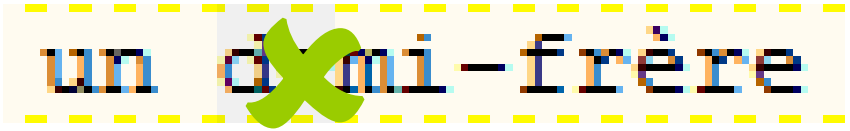
  \>

Übung Quantifikator 3: Gesucht wird frz. *de* oder *des*

```
A01:   [tss] [ins] avec une association
INT:   et vous allez où ?
A01:   oh beh si vous voulez je vous dis
INT:   [rir]
A01:   ah oui je suis allée [ins] je sui
INT:   #2 en Chine ? [rir] #
A01:   oui je suis allée euh pff en Birm
INT:   m-hm
A01:   vous voyez oui vous connaissez la
INT:   #2 un peu oui [rir] #
A01:   ah et puis les pays d' Europe oui
INT:   à Cologne vous avez ?
```







Quantifikator mit Positionsangabe: Beispiel 2

- Gesucht werden Sätze, die mit einem Punkt enden.

```
.$      service en milieu rural.
on de la source : Est Républicain/CNRTL
```

Punkt ist Metazeichen!

- Metazeichen aufheben mit Backslash: Punkt als literales Zeichen interpretieren

```
\.$     on de la source : Est Républicain/CNRTL
```

Gruppieren: Muster speichern
... und ersetzen

- Zeichenfolgen zusammenfassen

  `\([a-z]*\)` oder (je nach Version) `([a-z]*)`

- Zeichenfolgen ersetzen, umordnen, dabei auf Nr. verweisen
  - Suchen nach *Wort* gefolgt von *Zahl*

  `\([a-z]*\) \([0-9]*\)` oder `([a-z]*) ([0-9]*)`

  - Ersetzen durch *Zahl* gefolgt von *Wort*

  `\2 \1`

Weitere Zeichen

- Tabulator

  `\t`

- Zeilenwechsel

  `\n`

Exkurs XML
Extensible Markup Language

- Element: **Öffnendes** und **schließendes** Tag ("Markup", "balise")

```
<Turn speaker="spk2">  TEXT </Turn>
```

- Leeres Element: **Eingliedriges Tag mit Abschluss**

```
<Event desc="ins" type="noise" extent="instantaneous"/>
```

- **Paare: Attribut=** und **"Wert"**

```
<Turn speaker="spk2">  TEXT </Turn>
```

Übung: Nurtext in XML umformatieren

```
A01:   [tss] [ins] avec une association
INI:   et vous allez où ?
A01:   oh beh      vous voulez je vous dis
INI:   [rir]
A      ah oui      suis allée [ins] je sui
I      #2 en C     e ? [rir] #
A      oui je      s allée euh pff en Birm
I      m-hm
A      vous vo     oui vous connaissez la
I      #2 un p     oui [rir] #
A      ah et p     les pays d' Europe oui
I      à Colo      vous avez ?
```

<Event desc="ins" type="noise" extent="instantaneous"/>

<Turn speaker="spk2"> TEXT </Turn>

Übung: Nurtext in XML umformen

- Was steht da? (Suche "event")

```
\[[a-z]{3}\]
```

- Was soll da stehen? (Ersetze), verwende Ausgangstext
  - Ausgangstext benennen, durch runde Klammer

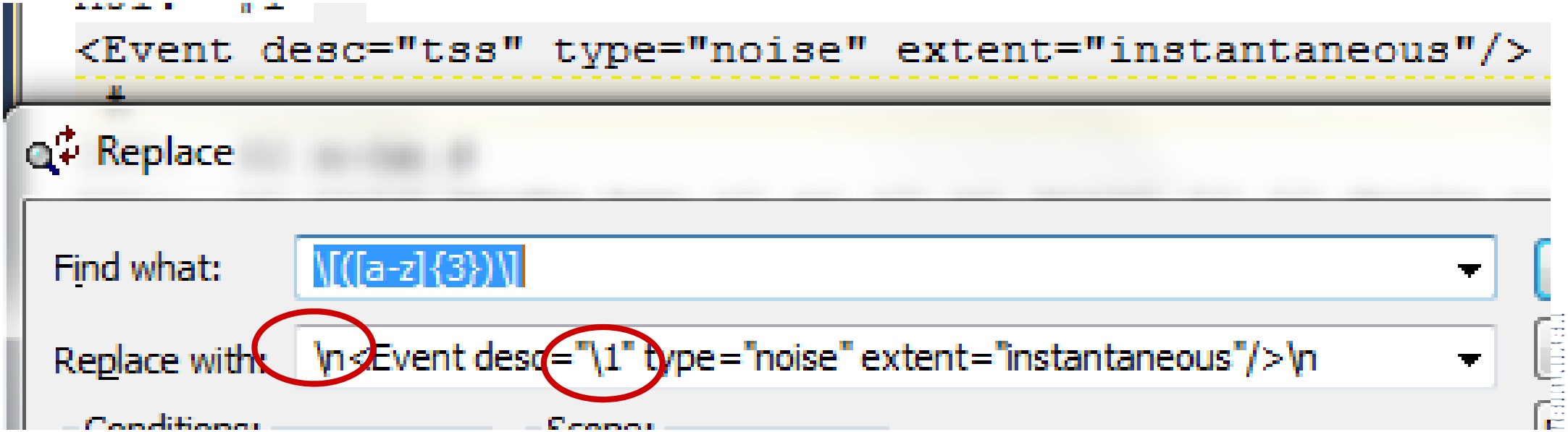
```
\[([a-z]{3})\]
```

  - Ersetzen durch

```
\n<Event desc="\1" type="noise" extent="instantaneous"/>\n
```

Ersetzen von "event" durch ergänzten Ausgangstext

[tss]



Übung: Nurtext in XML umformen

- Was steht da? (Suche Sprecherkodierung)

```
[A-Z][0-9]{2}:
```

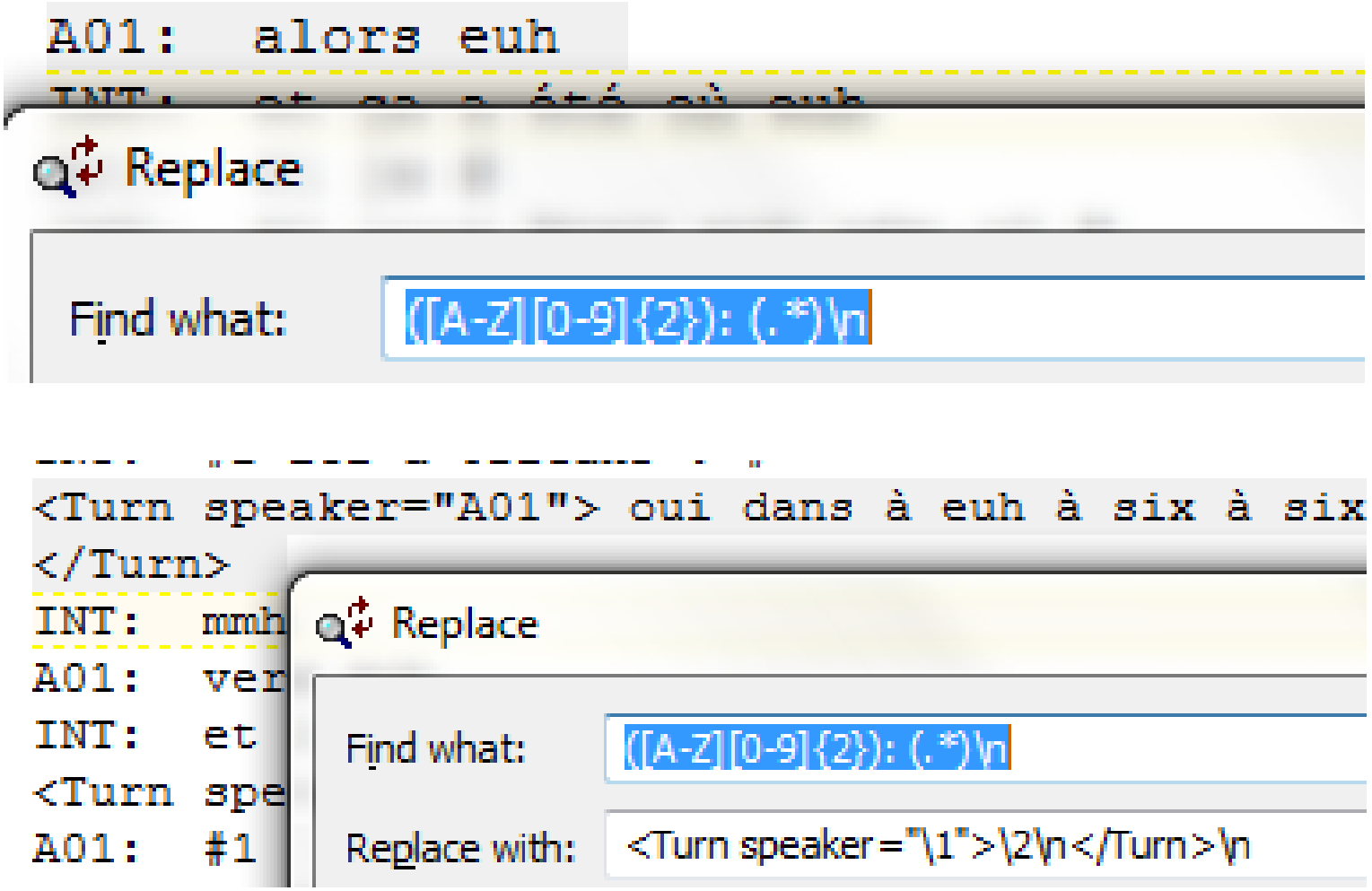
- Was soll da stehen? (Ersetze), verwende Ausgangstext
  - Ausgangstext benennen, durch runde Klammer

```
Replace what: ([A-Z][0-9]{2}): (.*)\n
```

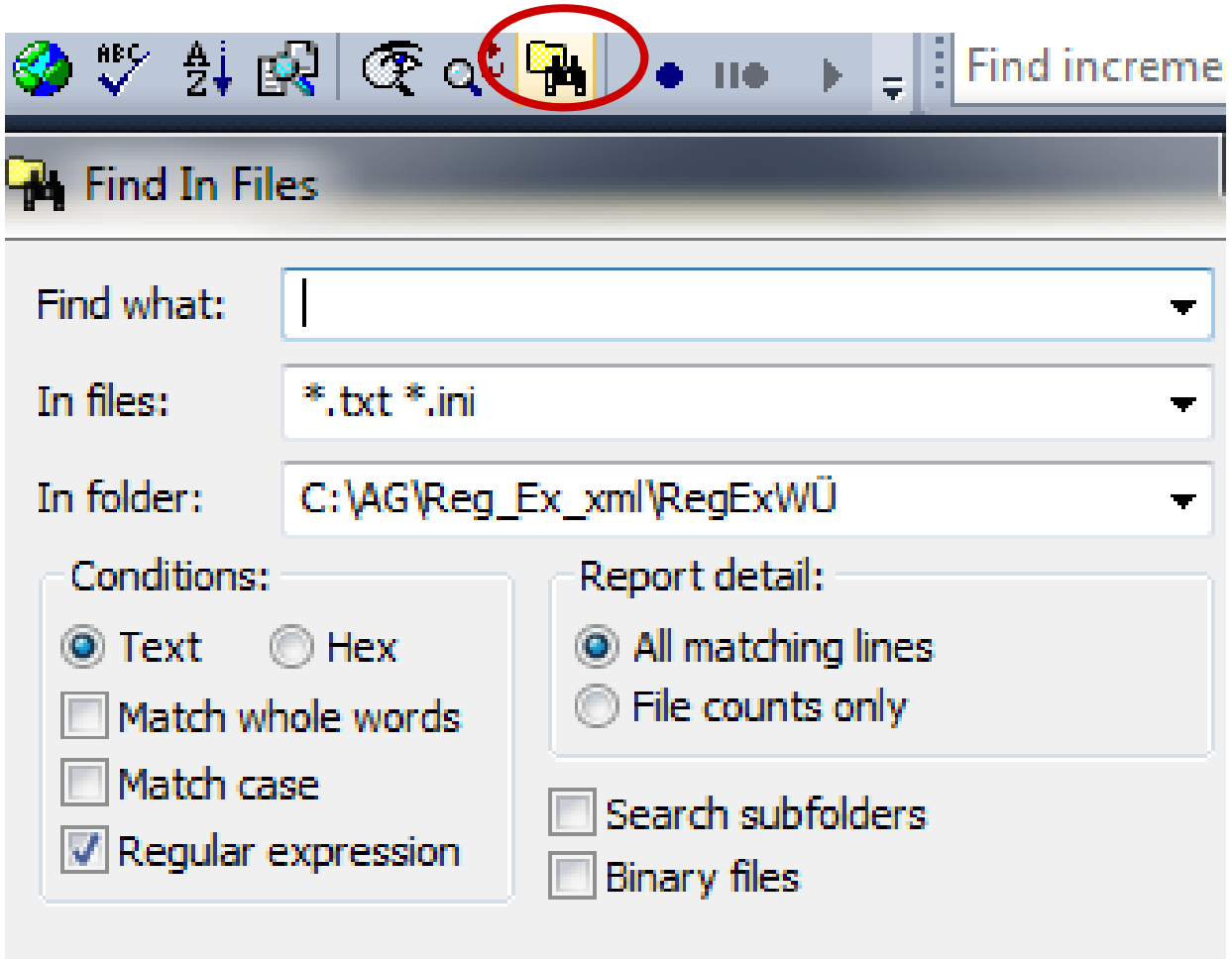
  - Ausgangstext referenzieren, durch Zahl

```
Replace with: <Turn speaker="\1">\2\n</Turn>\n
```

Übung: Sprecherkodierung in XML-Tag



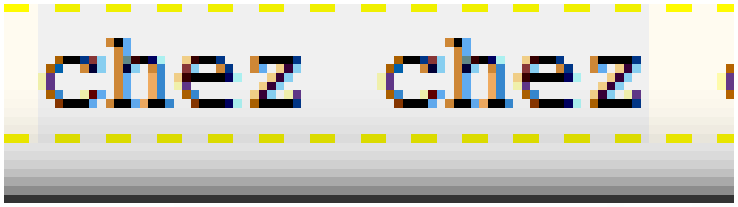
Suchen und Ersetzen in mehreren Dateien

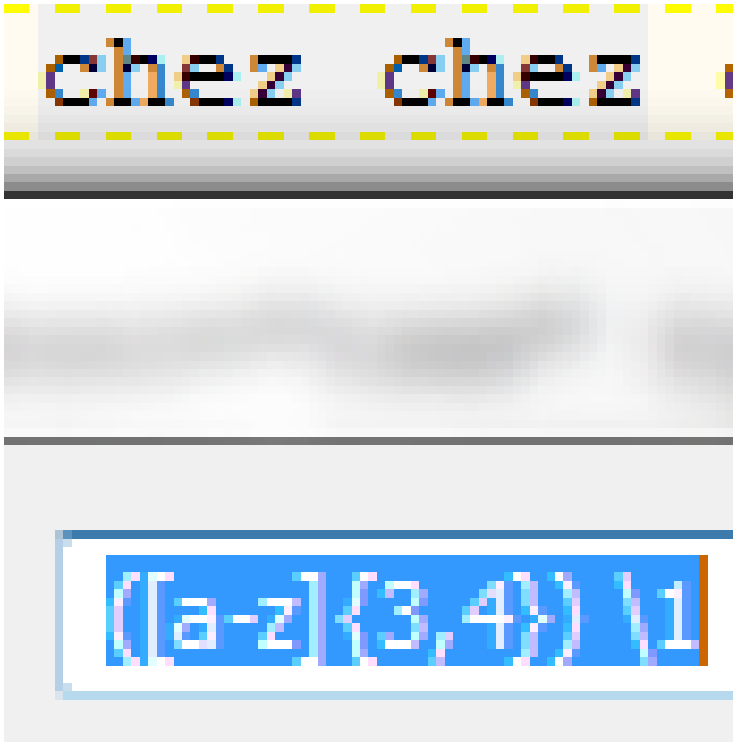


Übungsfragen

- Wortwiederholungen finden
  - *chez chez*
- Dateien bereinigen
  - Annotationen entfernen
  - Ziel: "nur Text"

Übung: Wiederholte Wörter mit 3-4 Buchstaben finden

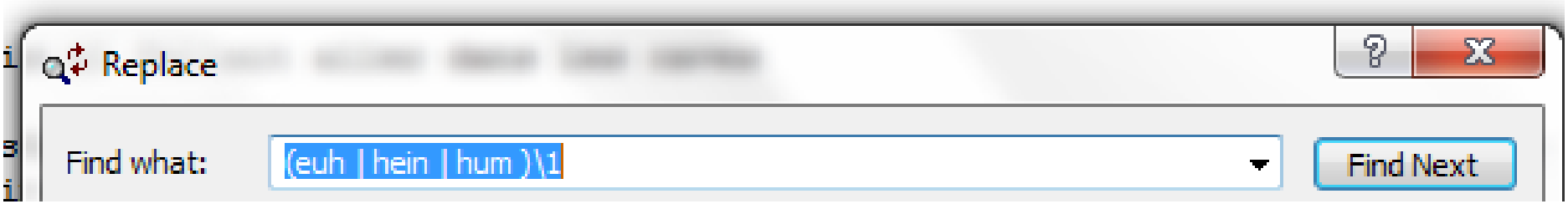




Übung: Wiederholte Interjektionen finden

euh euh

travailler euh euh c' était la guerre en mille-neuf-cent-quarante-deu



Übung: Sprachliche Phänomene annotieren

- Annotationen können in geeigneten Programmen ausgewertete werden
  - Konkordanzprogramme
  - Editor
  - R
- Beispiel Negation

```
<neg ne="0" adv="pas" lemma="savoir">je sais pas</neg>
<neg ne="1" adv="rien" lem="savoir">je n'en sais rien</neg>
```

Auswertung: Frequenzanalysen

- In Editor mit Kodierung UTF-8 speichern: Weiter in AntConc
- In Editor mit Kodierung ANSI speichern: Weiter mit