STYLOMETRY WITH R DETAILED INSTRUCTIONS

JOANNA BYSZUK IJP PAN, JAN RYBICKI, UJ

2018

STYLO
MAIN FUNCTIONS

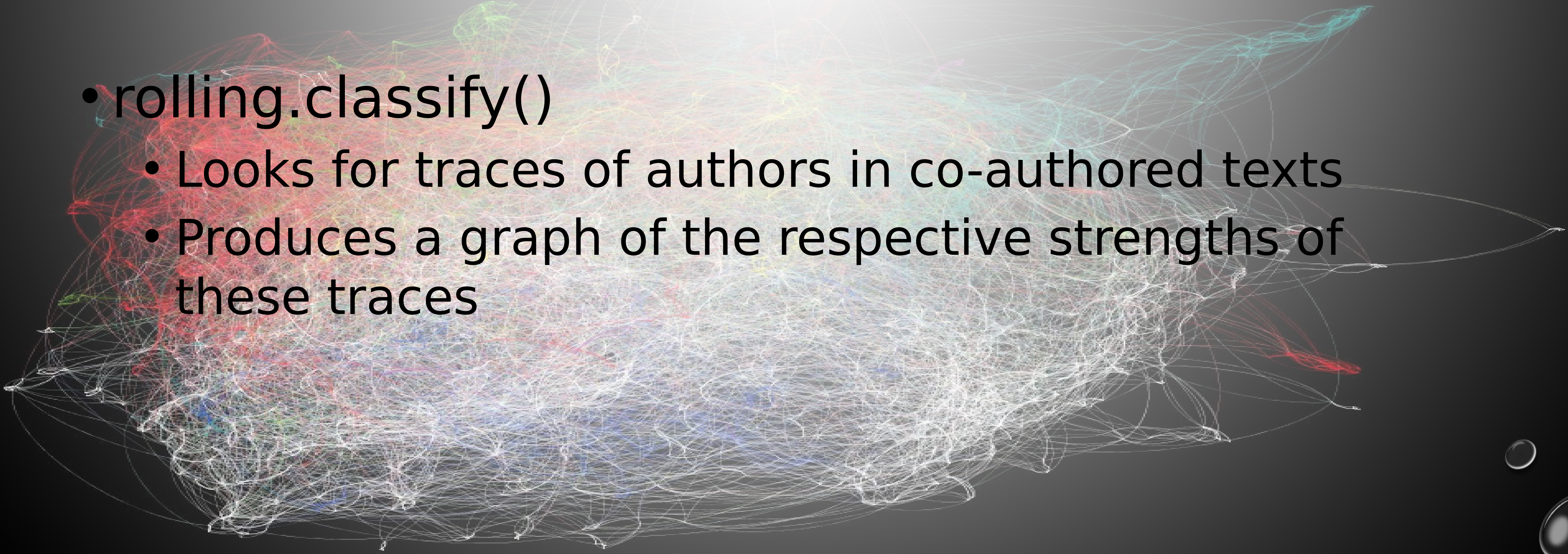
- stylo()
  - Calculates distances (differences) between series of most frequent words and draws graphs of those distances
    - CLUSTER ANALYSIS trees (for a single set of parameters)
    - BOOTSTRAP CONSENSUS trees (for multiple parameter settings)
    - MULTIDIMENSIONAL SCALING maps
    - PRINCIPAL COMPONENTS ANALYSIS maps

STYLO
MAIN FUNCTIONS

- oppose()
  - Cuts texts into equal-sized samples
  - Finds words characteristic for two (groups) of texts
    - These can be reused with stylo()
  - Produces a diagram of the use of each group’s words

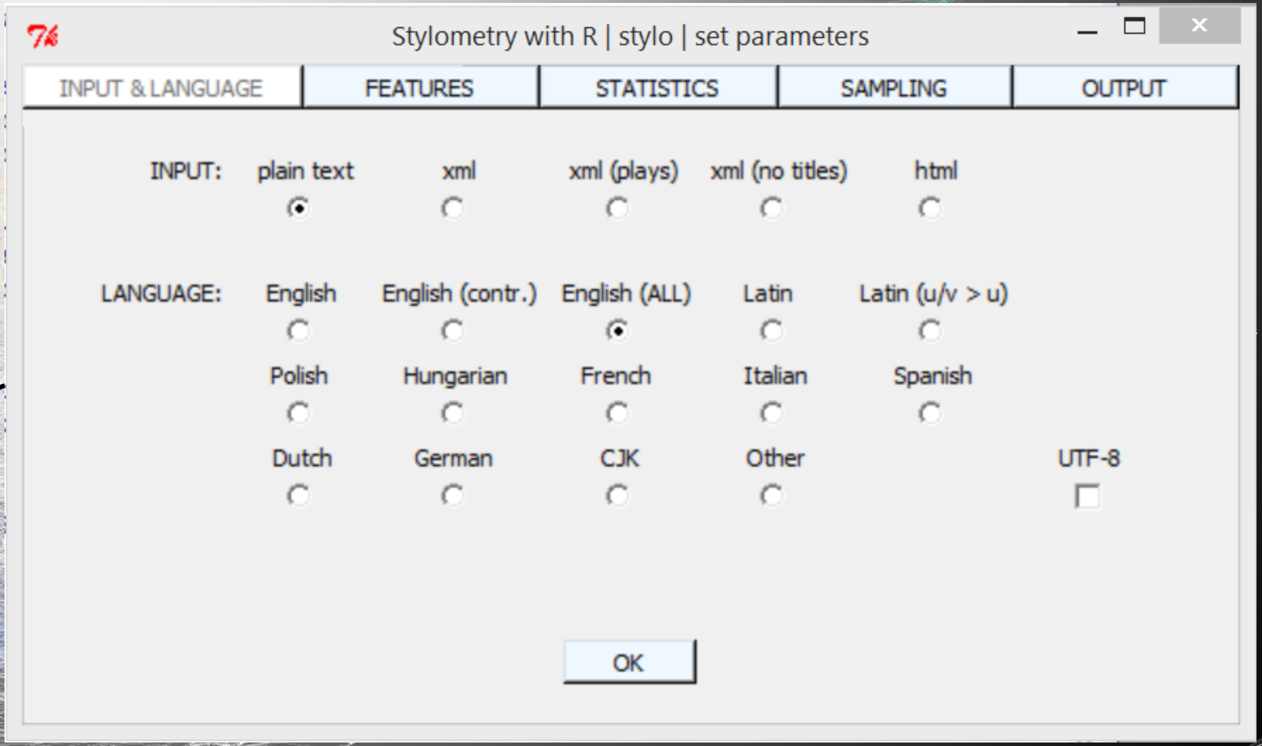
STYLO
MAIN FUNCTIONS

- rolling.classify()
  - Looks for traces of authors in co-authored texts
  - Produces a graph of the respective strengths of these traces



RUNNING STYLO()

- Where are my texts?:
  - MENU:
  - FILE > CHANGE DIRECTORY >
  - E.G. English Benchmark etc
- (it contains the subfolder „cor
- but don’t go there!)
- library(stylo) <ENTER>
- stylo() <ENTER>



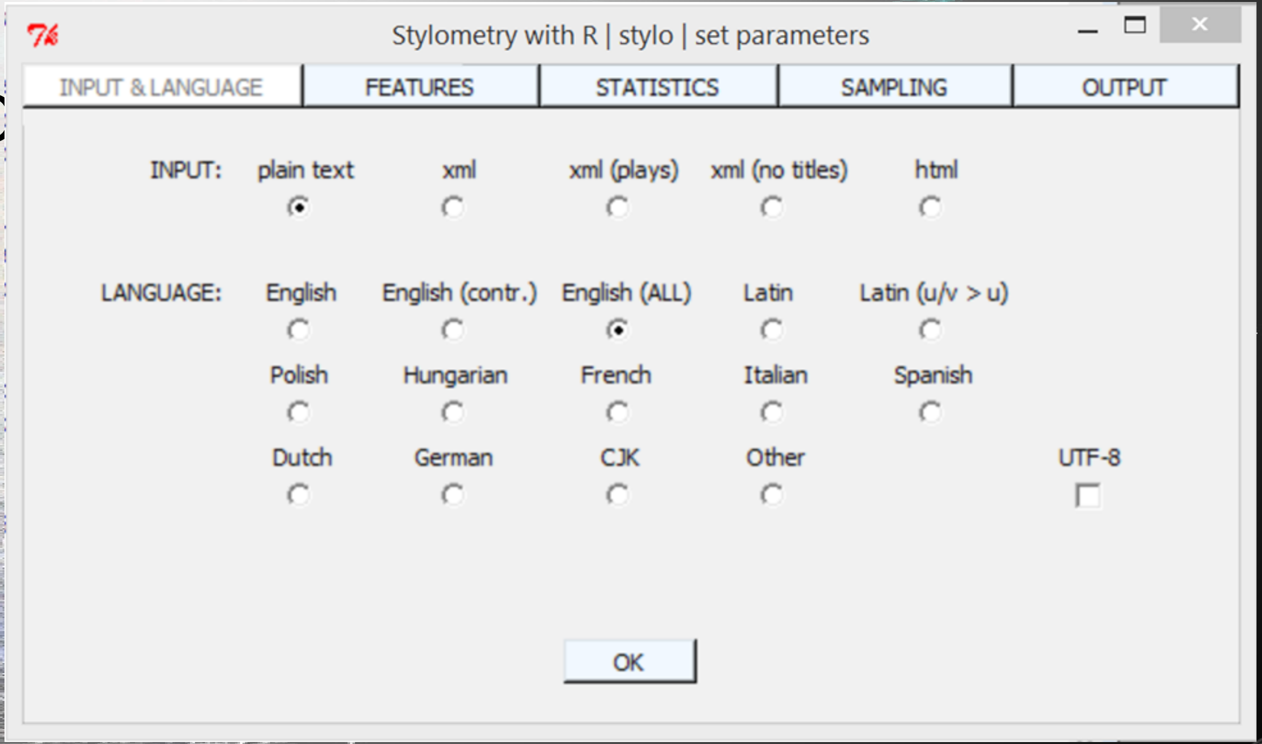
INPUT: ● plain text ○ xml ○ xml (plays) ○ xml (no titles) ○ html

LANGUAGE: ○ English ○ English (contr.) ● English (ALL) ○ Latin ○ Latin (u/v > u) ○ Polish ○ Hungarian ○ French ○ Italian ○ Spanish ○ Dutch ○ German ○ CJK ○ Other

UTF-8 □

STYLO() PARAMETERS

- INPUT: STATE YOUR TEXTS’ FC
- LANGUAGE
- DON’T PRESS „OK” YET!!!



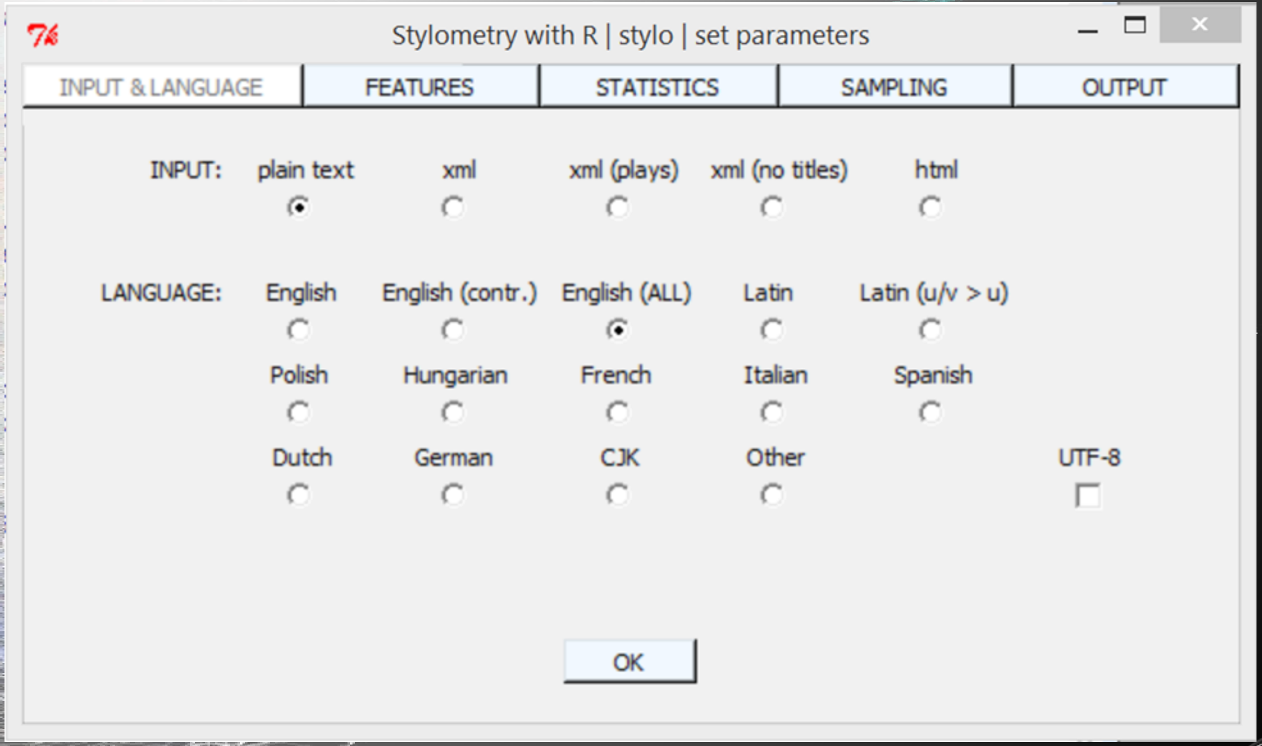
INPUT: ● plain text ○ xml ○ xml (plays) ○ xml (no titles) ○ html

LANGUAGE: ○ English ○ English (contr.) ● English (ALL) ○ Latin ○ Latin (u/v > u)
○ Polish ○ Hungarian ○ French ○ Italian ○ Spanish
○ Dutch ○ German ○ CJK ○ Other □ UTF-8

STYLO() PARAMETERS

- FEATURES: THINGS TO COUNT: WORDS OR CHARACTERS
  - ngram size: COUNT SINGLE FEATURES (1) OR THEIR CLUSTERS (>1)
- MFW SETTINGS: HOW MANY MOST FREQUENT WORDS TO USE

  UNLESS WE USE „bootstrap consensus tree” IN STATISTICS, Minimum=Maximum



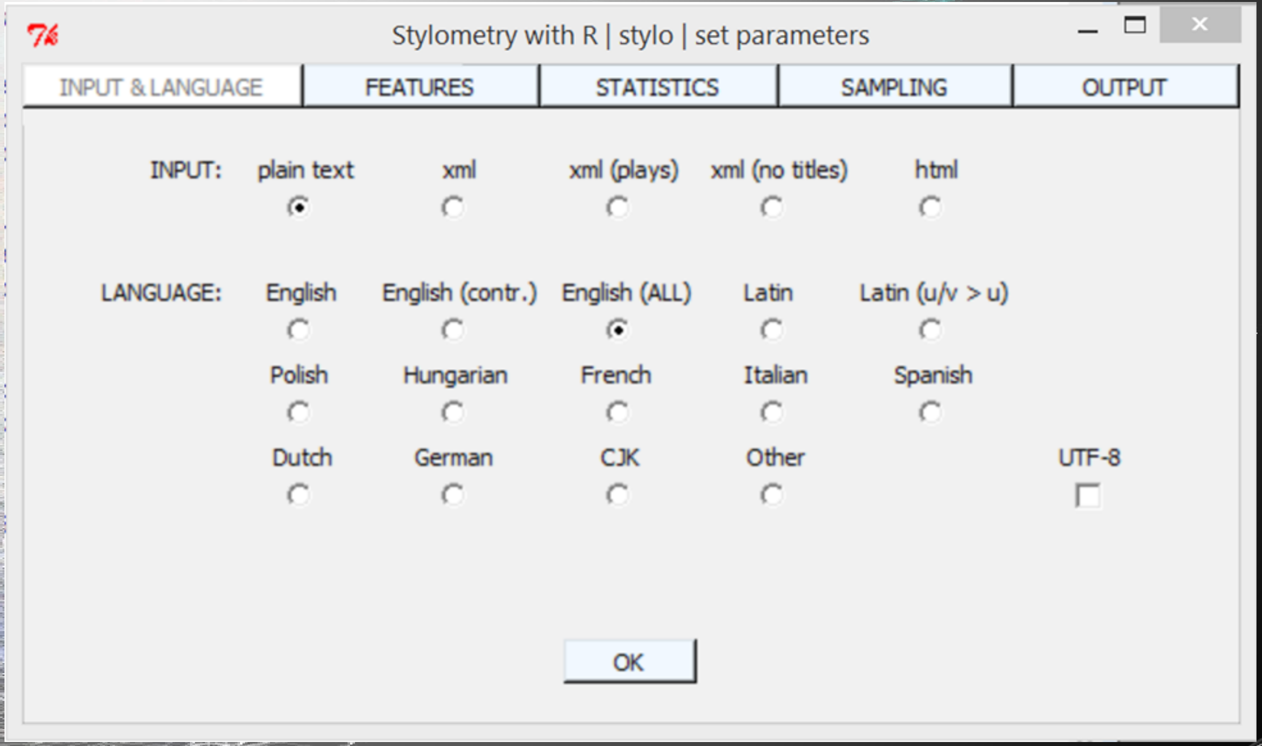
INPUT: ● plain text ○ xml ○ xml (plays) ○ xml (no titles) ○ html

LANGUAGE: ○ English ○ English (contr.) ● English (ALL) ○ Latin ○ Latin (u/v > u) ○ Polish ○ Hungarian ○ French ○ Italian ○ Spanish ○ Dutch ○ German ○ CJK ○ Other

□ UTF-8

STYLO() PARAMETERS

- CULLING: MANIPULATING THE WORDLIST (0)
  - 0%: NO WORDS ARE REMOVED
  - 100%: ALL WORDS ARE REMOVED THAT DO NOT OCCUR IN ALL THE TEXTS
- DELETE PRONOUNS?
- DON’T PRESS „OK” YET!!!



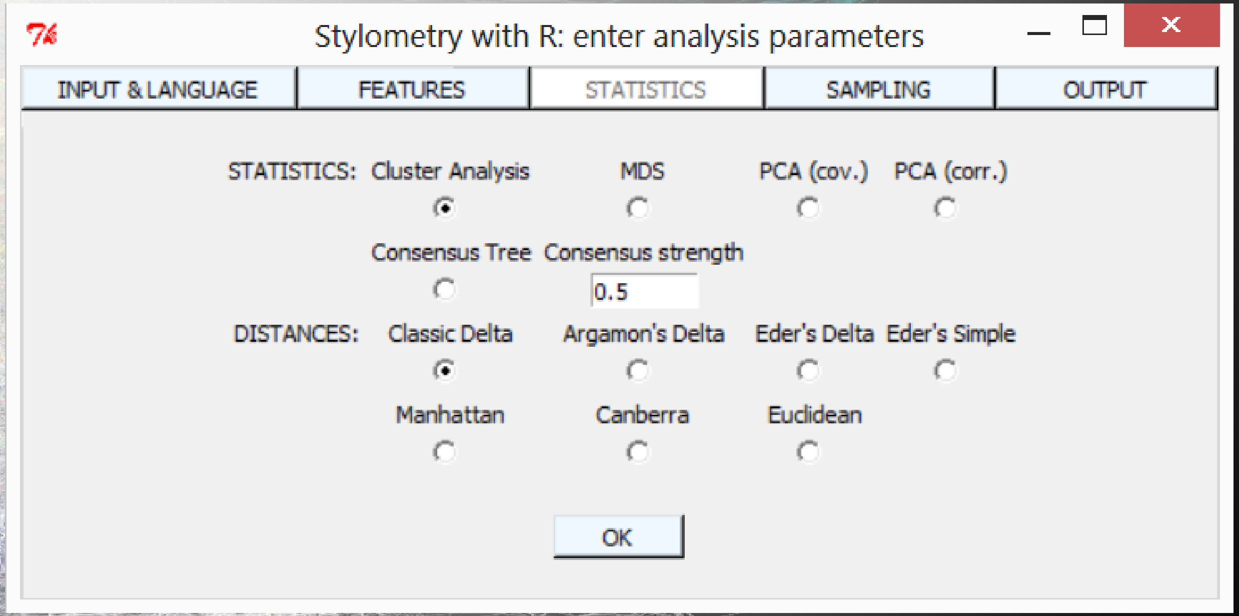
INPUT: ● plain text ○ xml ○ xml (plays) ○ xml (no titles) ○ html

LANGUAGE: ○ English ○ English (contr.) ● English (ALL) ○ Latin ○ Latin (u/v > u) ○ Polish ○ Hungarian ○ French ○ Italian ○ Spanish ○ Dutch ○ German ○ CJK ○ Other

UTF-8 □

STYLO() PARAMETERS

- STATISTICS: PICK STATISTICS METHOD (Cluster Analysis)
- DISTANCES: TYPE OF DISTANCE MEASURE (Classic Delta)
- DON'T PRESS „OK" YET!!!



Stylometry with R: enter analysis parameters

STATISTICS: Cluster Analysis ● MDS ○ PCA (cov.) ○ PCA (corr.) ○

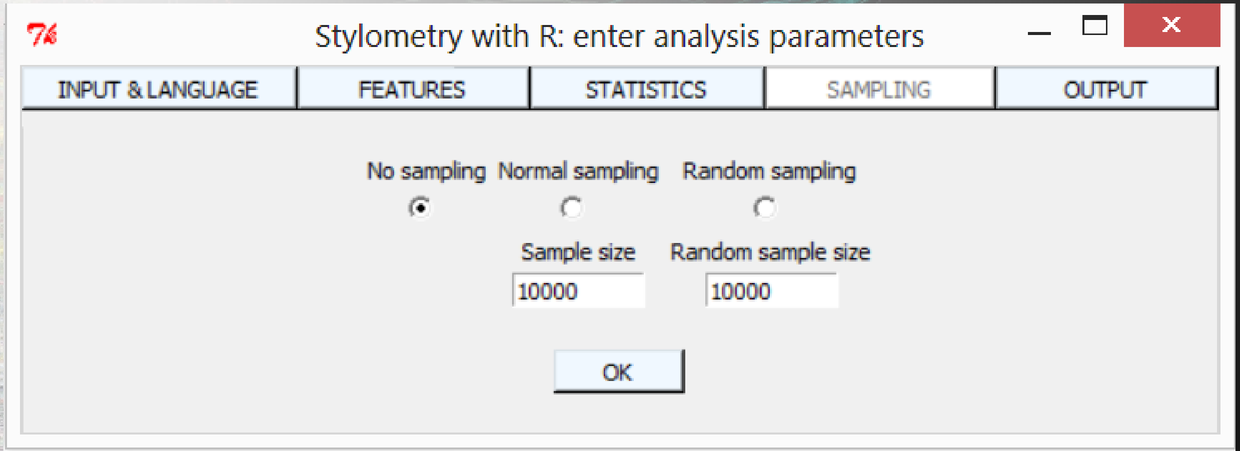
Consensus Tree ○ Consensus strength 0.5

DISTANCES: Classic Delta ● Argamon's Delta ○ Eder's Delta ○ Eder's Simple ○

Manhattan ○ Canberra ○ Euclidean ○

STYLO() PARAMETERS

- SAMPLING: (No sampling)
  - DO I WANT TO SAMPLE THE TEXTS
  - AND HOW
- DON'T PRESS „OK" YET!!!



Stylometry with R: enter analysis parameters

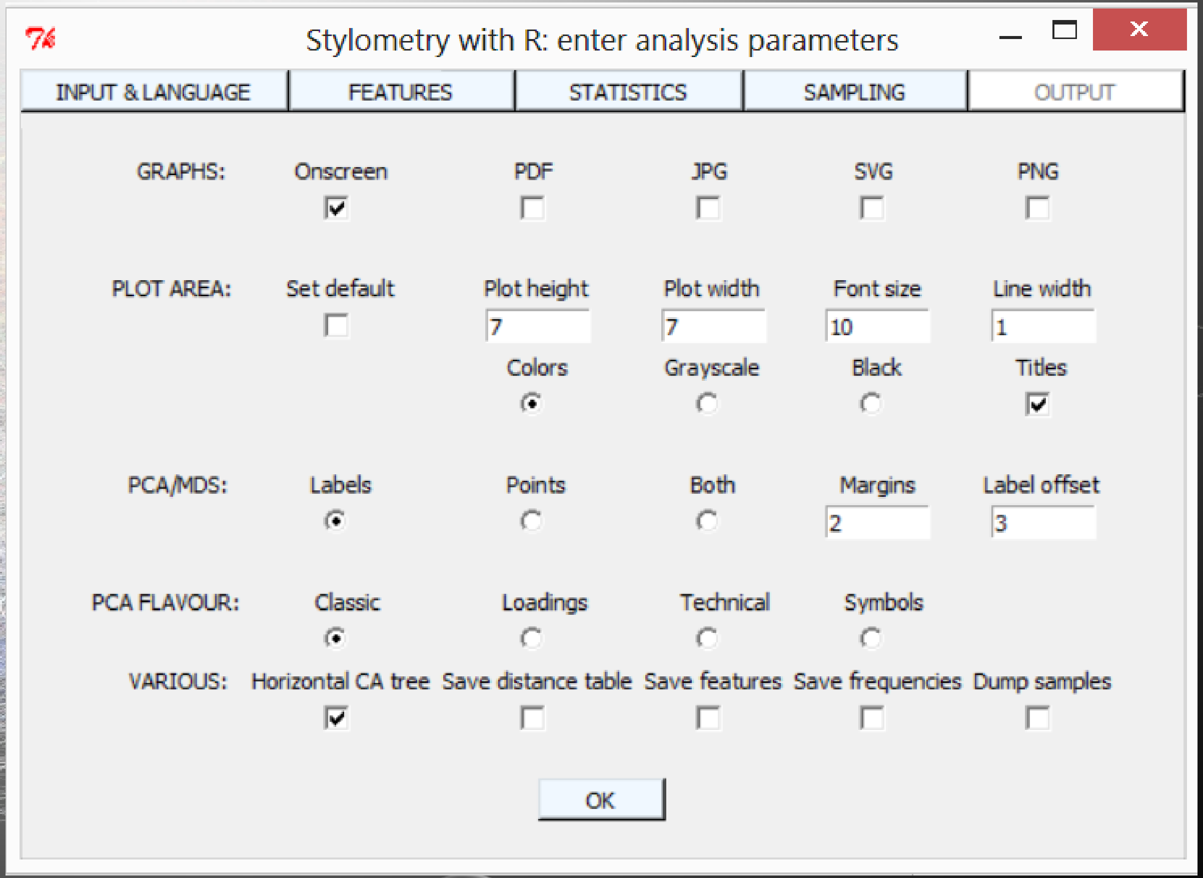
No sampling ● Normal sampling ○ Random sampling ○

Sample size: 10000

Random sample size: 10000

STYLO() PARAMETERS

- OUTPUT: (Onscreen)
  - GRAPH FORMAT ETC.
- PRESS „OK”!!!
- …WAIT FOR IT…

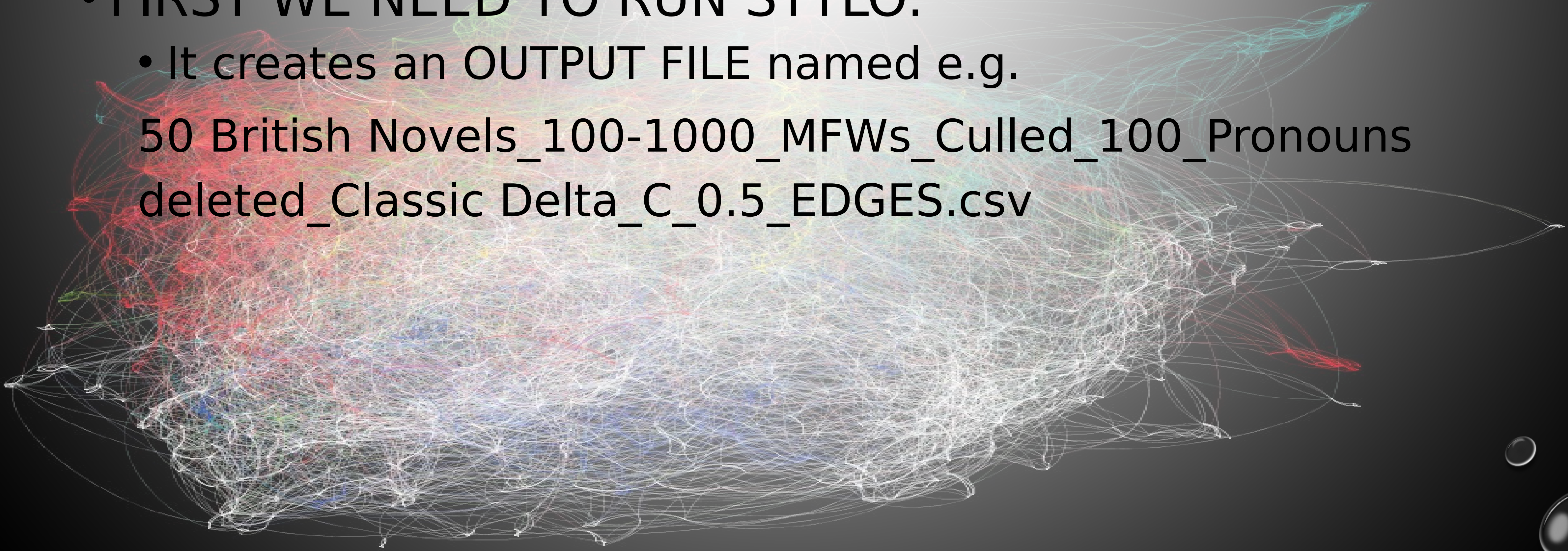


GRAPHS:	Onscreen ☑	PDF □	JPG □	SVG □	PNG □
PLOT AREA:	Set default □	Plot height 7	Plot width 7	Font size 10	Line width 1
		Colors ●	Grayscale ○	Black ○	Titles ☑
PCA/MDS:	Labels ●	Points ○	Both ○	Margins 2	Label offset 3
PCA FLAVOUR:	Classic ●	Loadings ○	Technical ○	Symbols ○	
VARIOUS:	Horizontal CA tree ☑	Save distance table □	Save features □	Save frequencies □	Dump samples □

RUNNING GEPHI

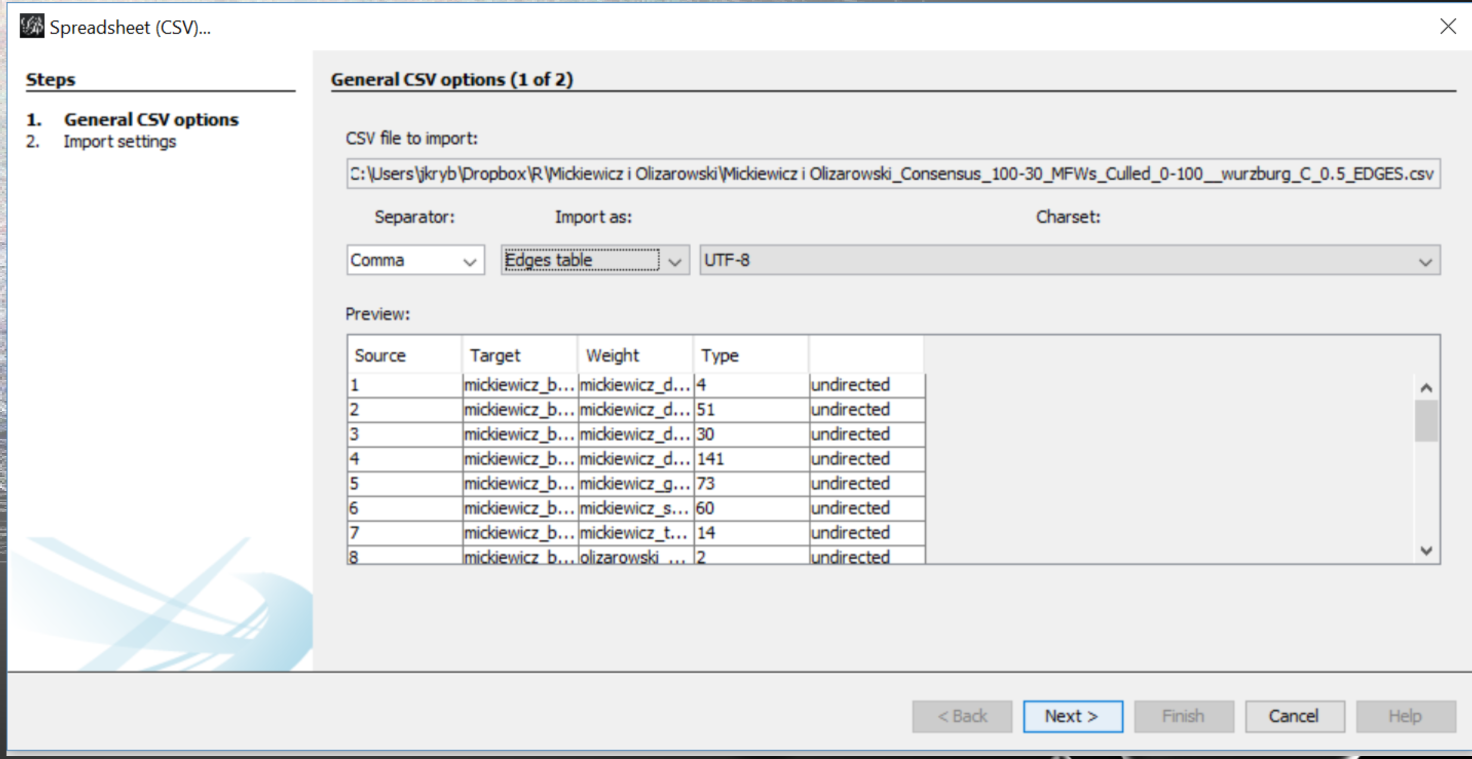
- FIRST WE NEED TO RUN STYLO:
  - It creates an OUTPUT FILE named e.g.

  50 British Novels_100-1000_MFWs_Culled_100_Pronouns deleted_Classic Delta_C_0.5_EDGES.csv



RUNNING GEPHI

- SELECT GEPHI>NEW PROJECT
- Data laboratory>Import Spreadsheet
- Import settings:
  - Separation: Comma
  - As table: Edges table
  - Charset: UTF-8? Windows-1252…
- Don’t worry about this being somewhat illogical…
- Next



Spreadsheet (CSV)...

Steps

1. **General CSV options**
2. Import settings

General CSV options (1 of 2)

CSV file to import:

C:\Users\jkryb\Dropbox\R\Mickiewicz i Olizarowski\Mickiewicz i Olizarowski_Consensus_100-30_MFWs_Culled_0-100__wurzburg_C_0.5_EDGES.csv

Separator: Comma

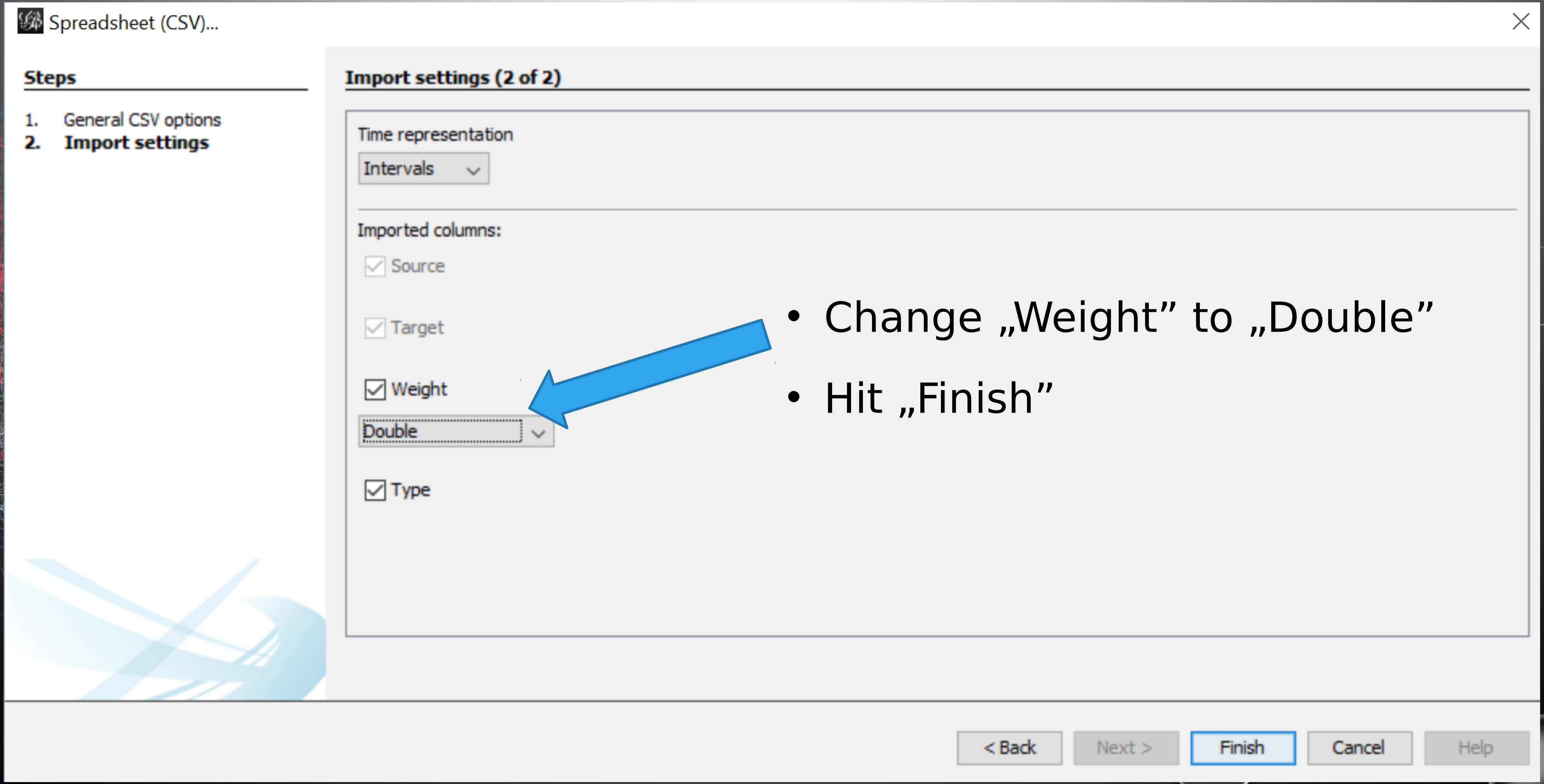
Import as: Edges table

Charset: UTF-8

Preview:

Source	Target	Weight	Type	
1	mickiewicz_b...	mickiewicz_d...	4	undirected
2	mickiewicz_b...	mickiewicz_d...	51	undirected
3	mickiewicz_b...	mickiewicz_d...	30	undirected
4	mickiewicz_b...	mickiewicz_d...	141	undirected
5	mickiewicz_b...	mickiewicz_g...	73	undirected
6	mickiewicz_b...	mickiewicz_s...	60	undirected
7	mickiewicz_b...	mickiewicz_t...	14	undirected
8	mickiewicz_b...	olizarowski_...	2	undirected

RUNNING GEPHI



Spreadsheet (CSV)...

Steps

1. General CSV options
2. **Import settings**

Import settings (2 of 2)

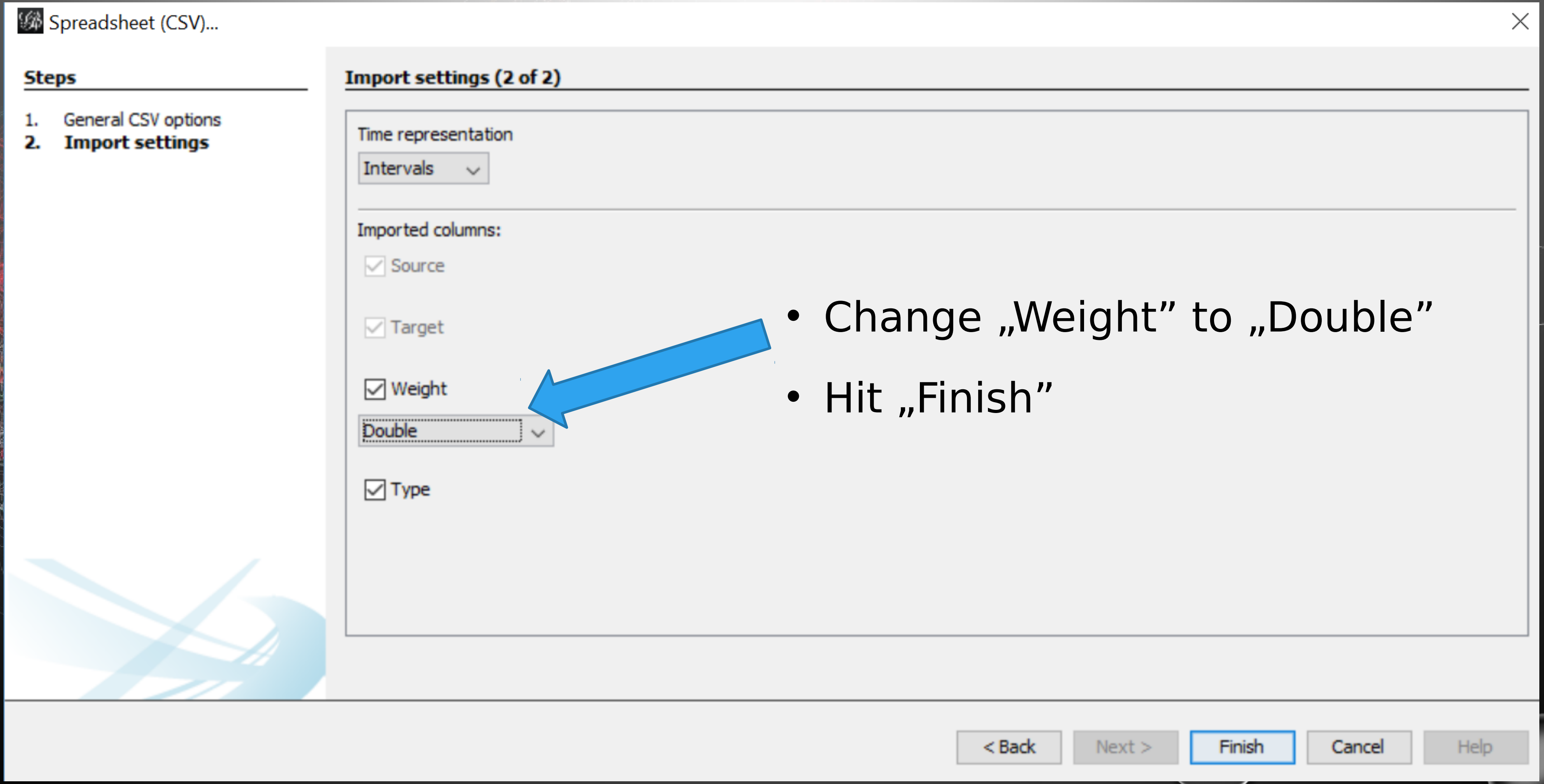
Time representation: Intervals

Imported columns:

- ☑ Source
- ☑ Target
- ☑ Weight: Double
- ☑ Type

- Change „Weight” to „Double”
- Hit „Finish”

RUNNING GEPHI



Spreadsheet (CSV)...

Steps

1. General CSV options
2. **Import settings**

Import settings (2 of 2)

Time representation: Intervals

Imported columns:

☑ Source

☑ Target

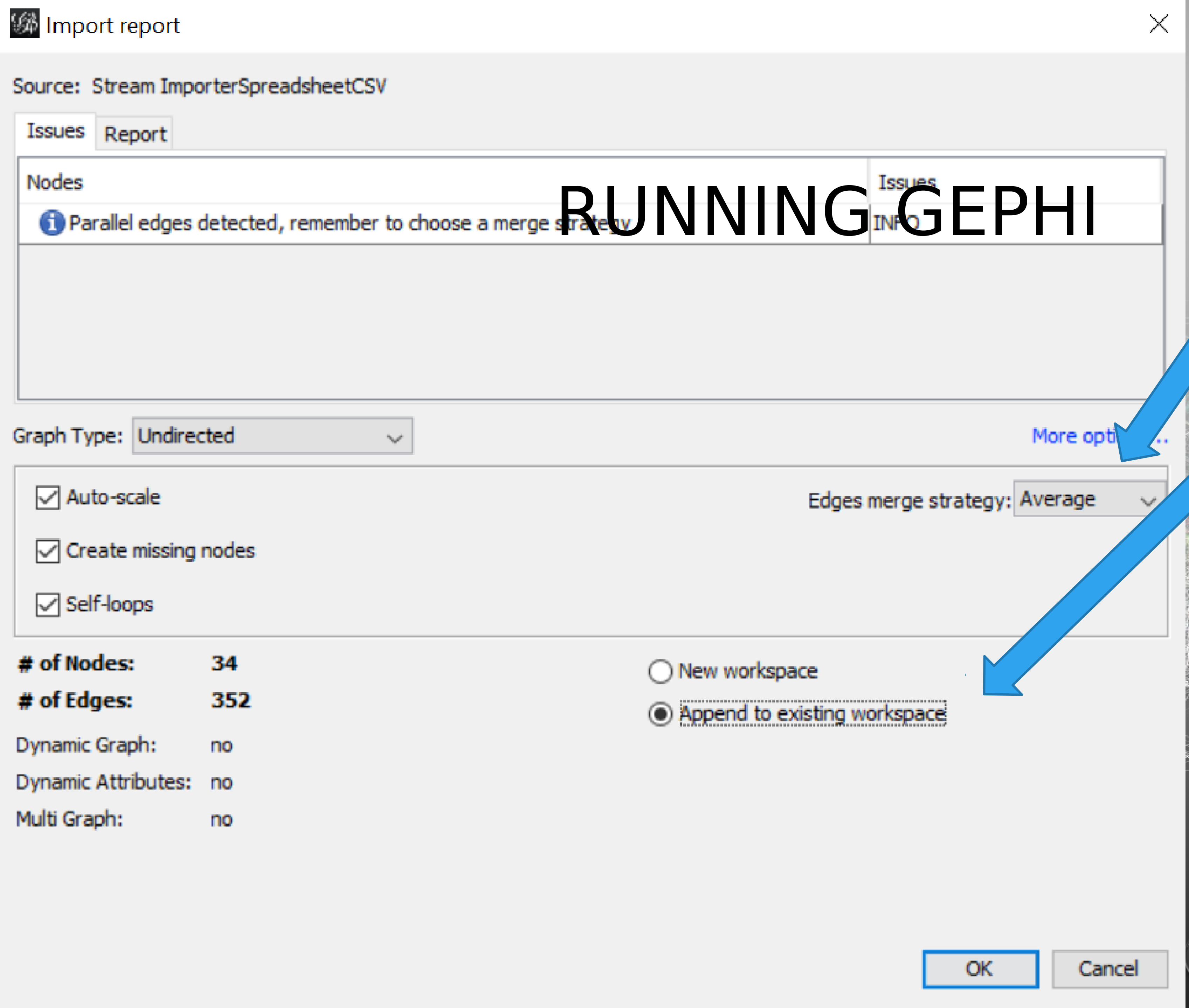
☑ Weight

Double

☑ Type

- Change „Weight” to „Double”
- Hit „Finish”

RUNNING GEPHI



Import report

Source: Stream ImporterSpreadsheetCSV

Issues | Report

Nodes	Issues
Parallel edges detected, remember to choose a merge strategy	INFO

Graph Type: Undirected

More options...

☑ Auto-scale

☑ Create missing nodes

☑ Self-loops

Edges merge strategy: Average

# of Nodes: **34**

# of Edges: **352**

Dynamic Graph: no

Dynamic Attributes: no

Multi Graph: no

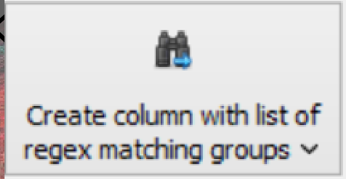
○ New workspace

● Append to existing workspace

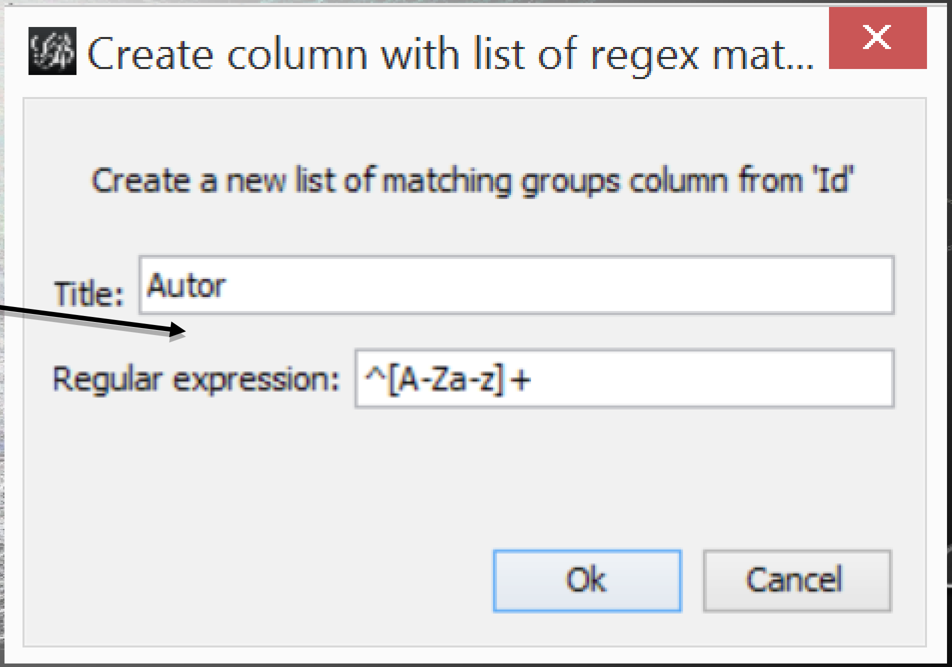
- Change „Edges merge strategy to „Average”
- Set „Append to existin workspace”
- Hit „OK”

RUNNING GEPHI

- We need to get authors' names…
- Click



- And set:
- OK!
- Copy ID column to LABEL

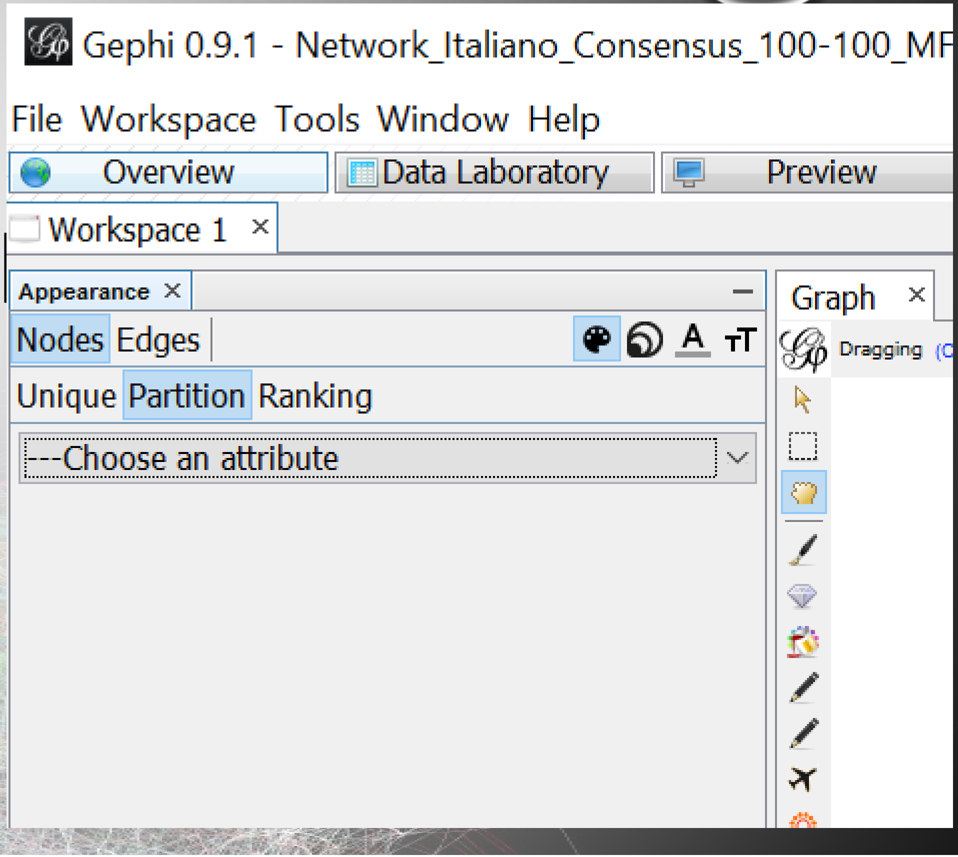

Create column with list of regex mat...

Create a new list of matching groups column from 'Id'

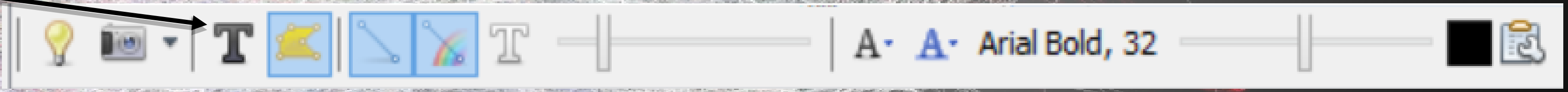
Title: Autor

Regular expression: ^[A-Za-z]+

RUNNING GE

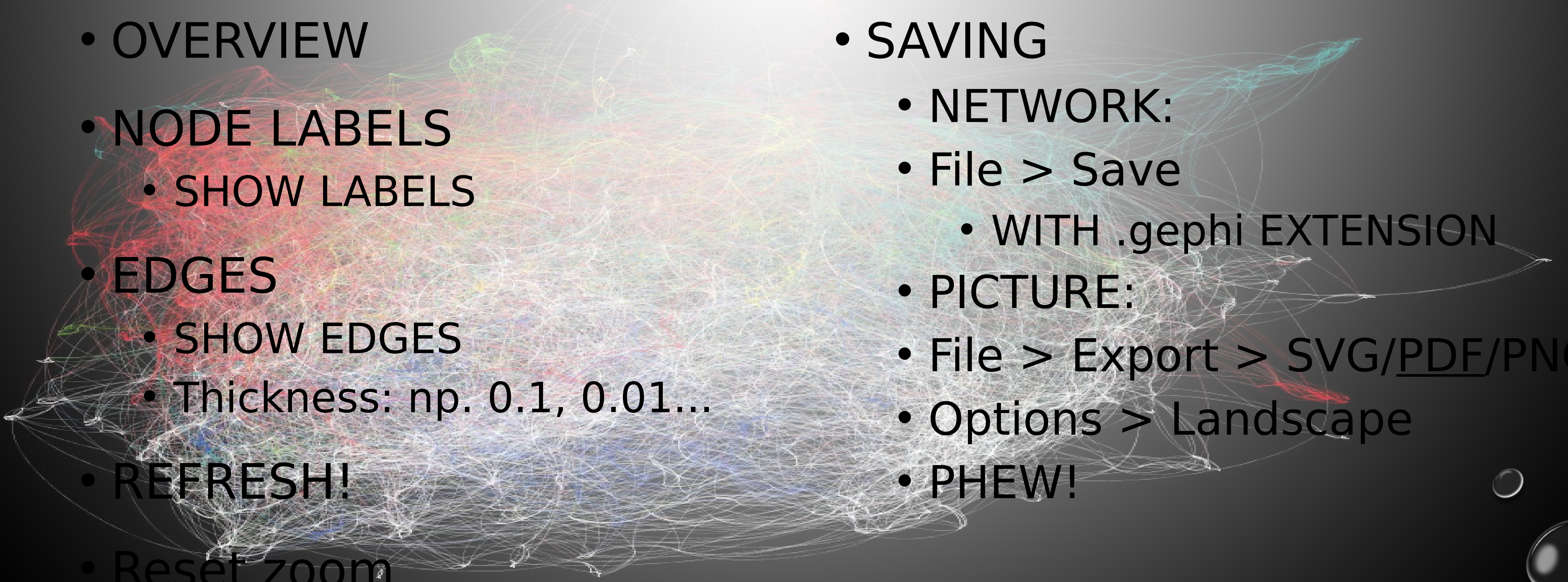


- PREVIEW
- PARTITION
  - CLICK
  - SELECT: e.g. Author
  - Apply
- Show labels



- LAYOUT
  - ForceAtlas 2
    - Dissuade Hubs
    - Prevent Overlap
    - Edge Weight Influence 0.5
    - Scaling: 500
    - RUN!
- LAYOUT (CONT)
  - Expansion
    - RUN!

RUNNING GEPHI

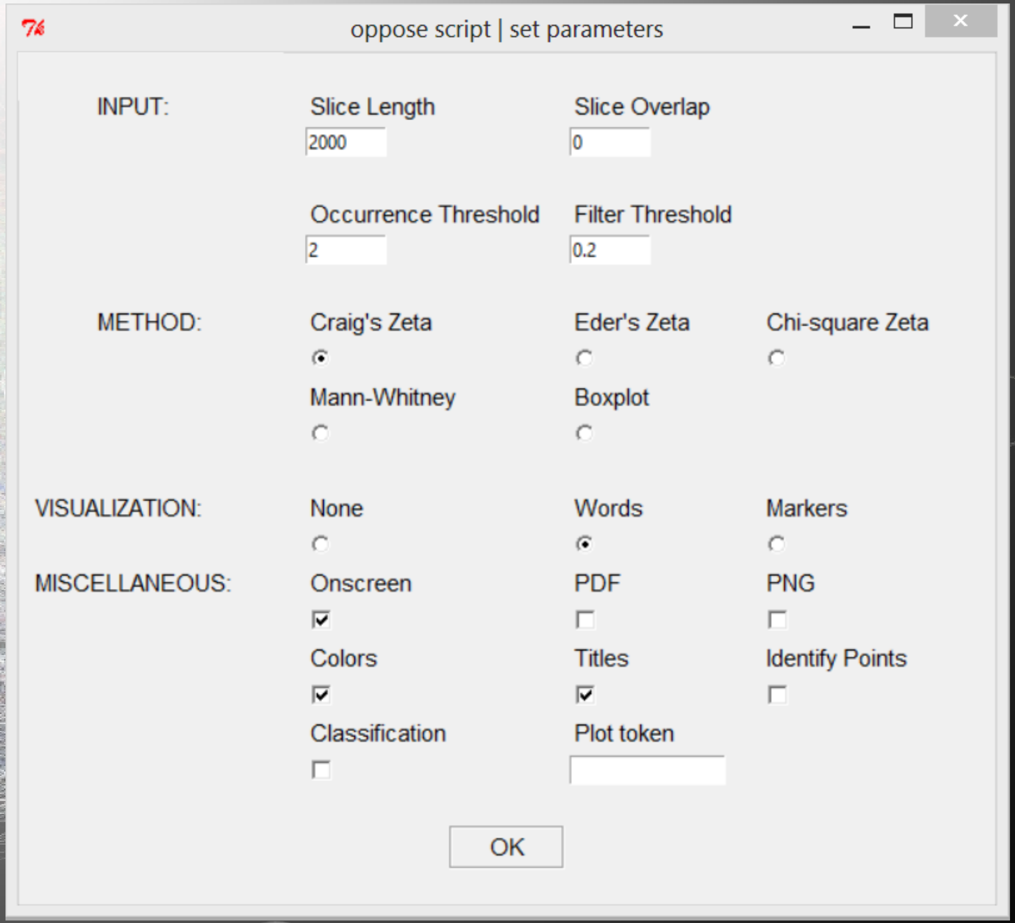


- OVERVIEW
- NODE LABELS
  - SHOW LABELS
- EDGES
  - SHOW EDGES
  - Thickness: np. 0.1, 0.01…
- REFRESH!
- Reset zoom

- SAVING
  - NETWORK:
  - File > Save
    - WITH .gephi EXTENSION
  - PICTURE:
  - File > Export > SVG/PDF/PNG
  - Options > Landscape
  - PHEW!

RUNNING OPPOSE

- DIFFERENT SUBFOLDER STRUCTURE:
  - primary_set
  - secondary_set
  - test_set (OPTIONAL)
- library(stylo) <ENTER>
- oppose() <ENTER>
- What we get:
  - WORDS_PREFERRED characteristic for the primary_set texts
  - WORDS_AVOIDED characteristic for the secondary_set texts
  - word frequency graph



oppose script | set parameters

INPUT:
Slice Length: 2000
Slice Overlap: 0
Occurrence Threshold: 2
Filter Threshold: 0.2

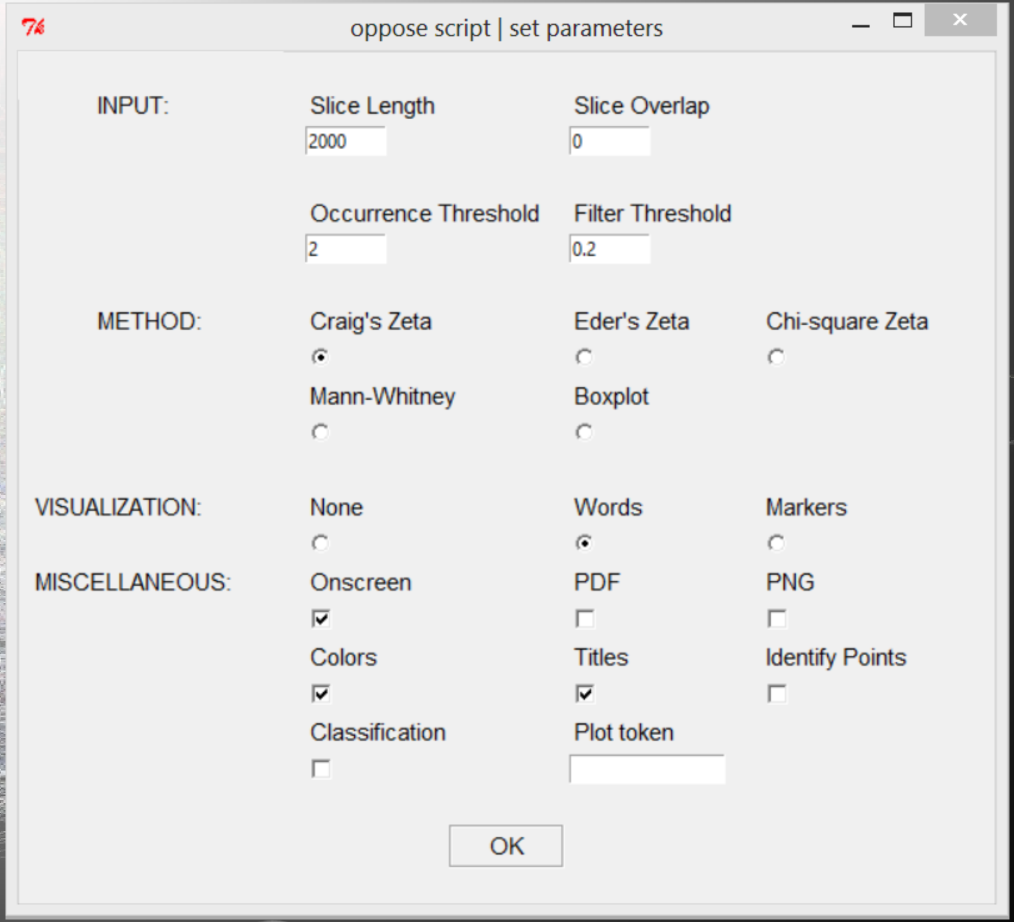
METHOD:
Craig's Zeta ●
Eder's Zeta ○
Chi-square Zeta ○
Mann-Whitney ○
Boxplot ○

VISUALIZATION:
None ○
Words ●
Markers ○

MISCELLANEOUS:
Onscreen ☑
PDF □
PNG □
Colors ☑
Titles ☑
Identify Points □
Classification □
Plot token:

OPPOSE() PARAMETERS

- Slice length: size (in words) of the samples (5000)
- Slice overlap: (0)
- Method: (Craig’s Zeta)
- Visualization: type of graph (Markers)



oppose script | set parameters

INPUT:
Slice Length: 2000
Slice Overlap: 0
Occurrence Threshold: 2
Filter Threshold: 0.2

METHOD:
● Craig's Zeta
○ Eder's Zeta
○ Chi-square Zeta
○ Mann-Whitney
○ Boxplot

VISUALIZATION:
○ None
● Words
○ Markers

MISCELLANEOUS:
☑ Onscreen
□ PDF
□ PNG
☑ Colors
☑ Titles
□ Identify Points
□ Classification
Plot token:

RUNNING ROLLING.CLASSIFY

- DIFFERENT SUBFOLDER STRUCTURE (AGAIN):
  - reference_set (individual writings)
  - test_set (collaborative text)
- library(stylo) <ENTER>
- rolling.classify(write.png.file = TRUE, classification.method = "delta",mfw=100, training.set.sampling = "normal.sampling", slice.size = 5000,slice.overlap = 4500)
- What we get:
  - Similarity graph