UNIVERSITÄT LEIPZIG

European Research Council

Connectivity semantic maps and graphs

Natalia Levshina ©2017

Summer School of Linguistics

Litomyšl, August 2017

Outline

1. Introduction to semantic maps

2. Case study: colexification patterns of causative constructions

3. Appendix: how to create graphs in (1) with R

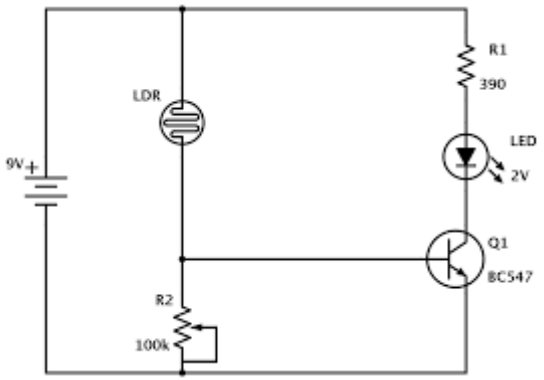
Semantic maps

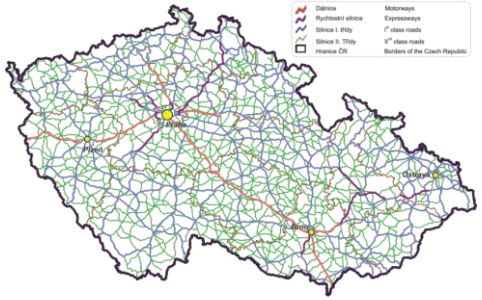
- Represent a convenient tool for comparison of semantic and pragmatic functions across languages
- Based on different kinds of data: grammars and typological databases, parallel corpora and experimental data

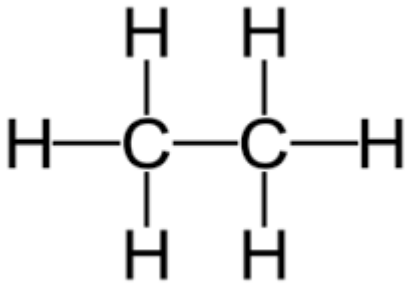
Main types of semantic maps

- Connectivity maps, with meanings as nodes (**vertices**) and links (**edges**) between them. Such maps are called **graphs**.
  - Undirected vs. directed (e.g. diachrony)
  - Weighted vs. unweighted
- Probabilistic maps, with distances between objects (e.g. examples from a parallel corpus or stimuli in an experiment)

Graphs are everywhere!

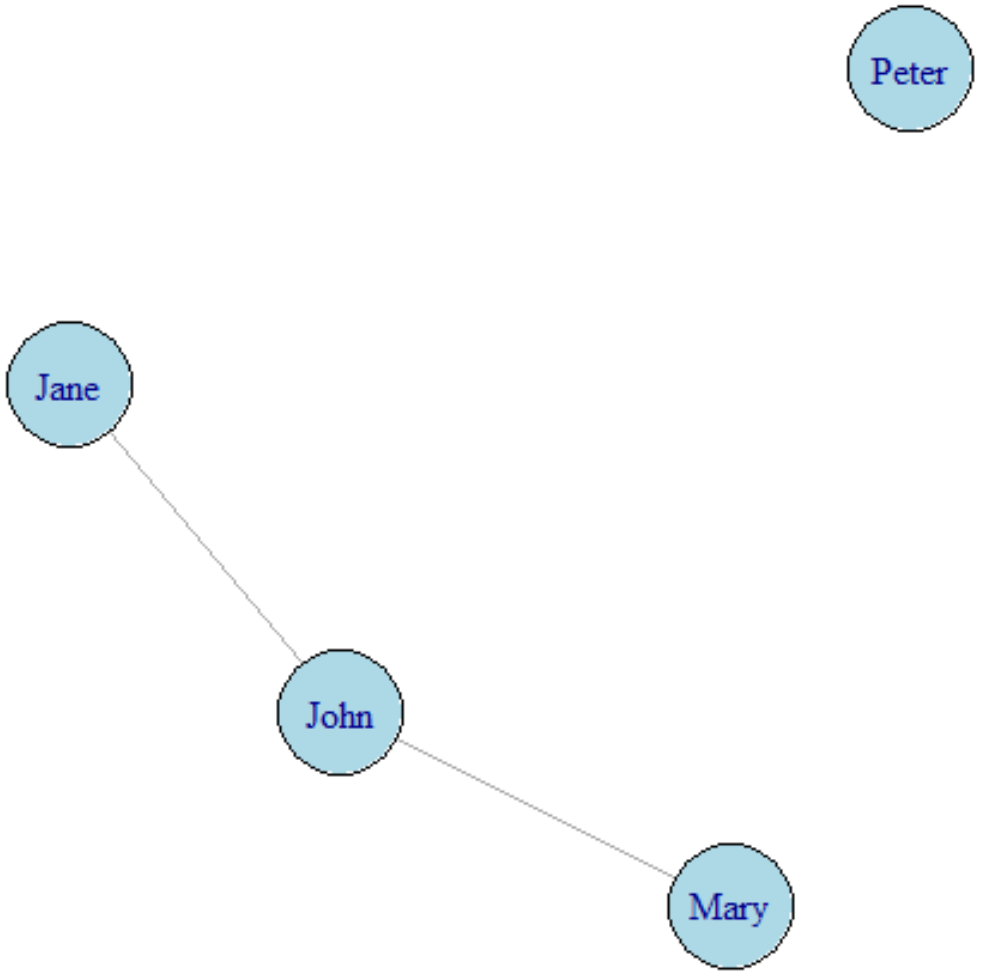




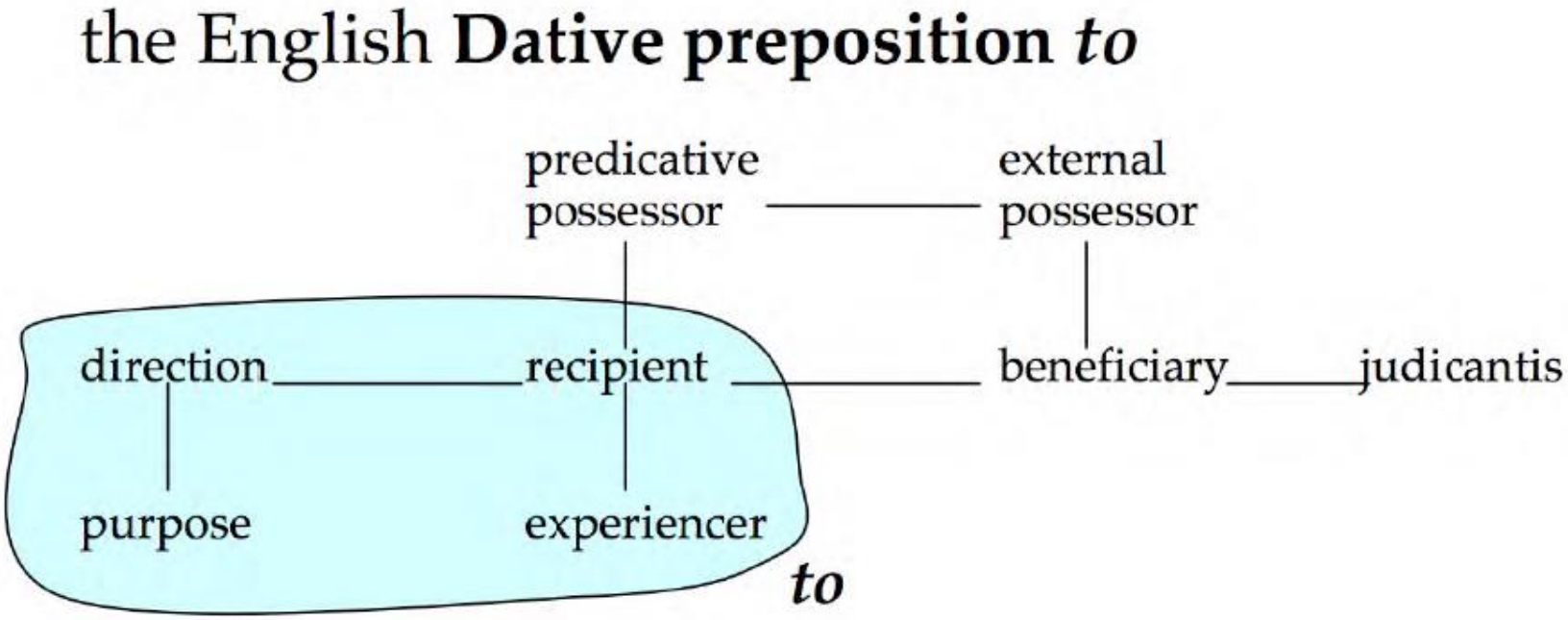


Undirected graphs: marriage

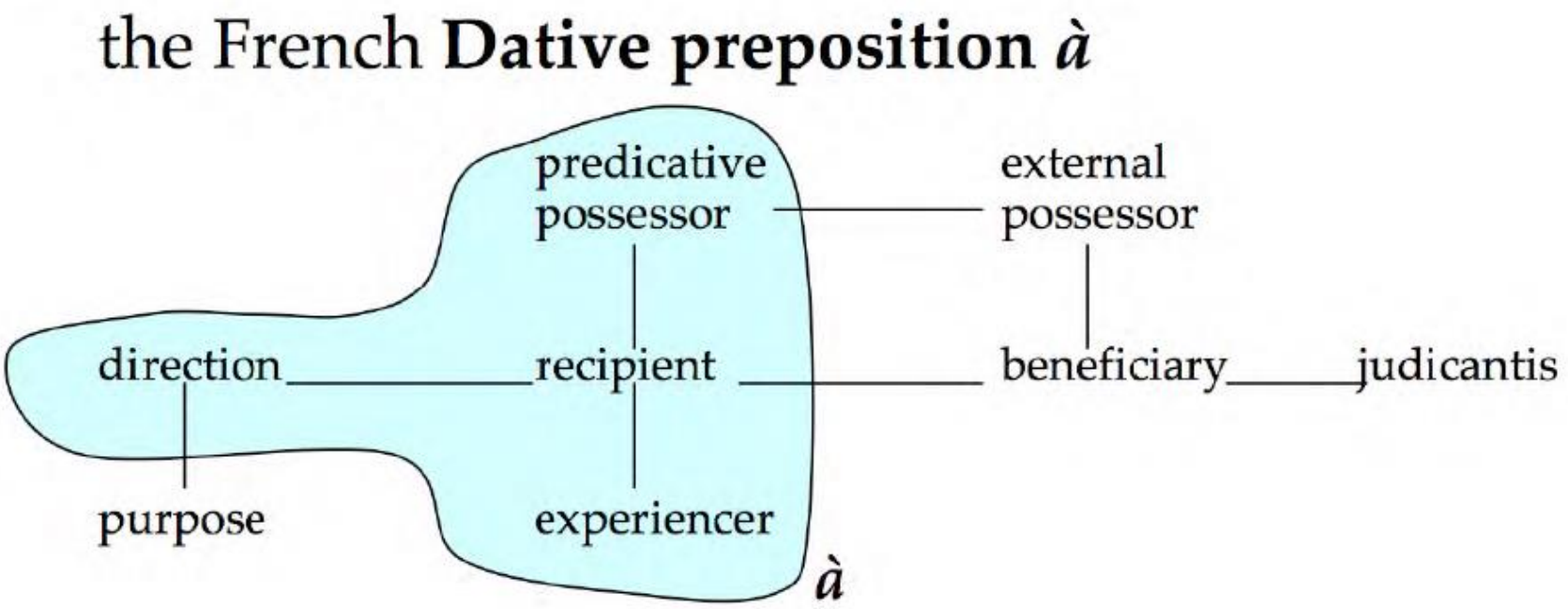
- Mary is John's ex-wife.
- John and Jane are married.
- Peter is single.



Haspelmath 2003: Datives



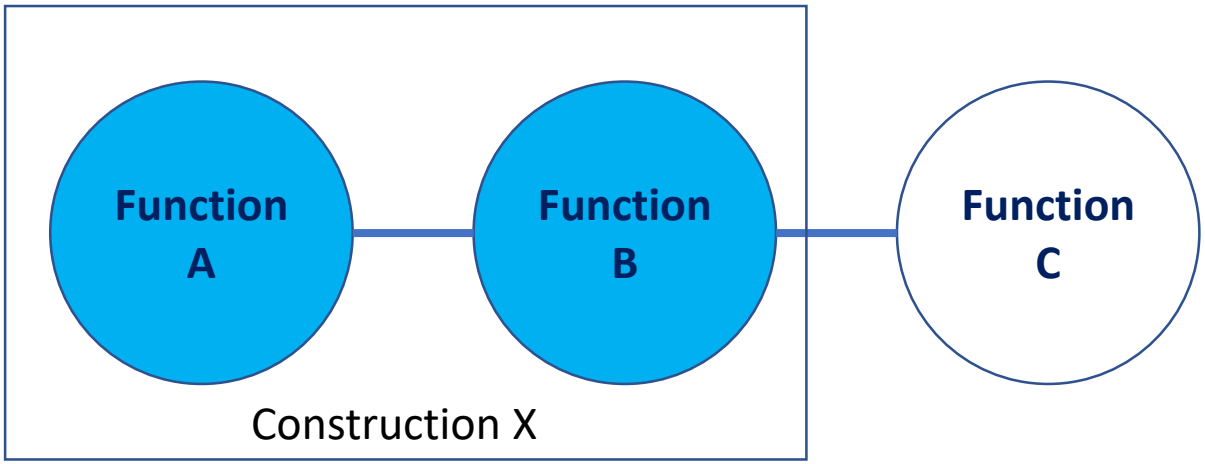
Haspelmath 2003: Datives



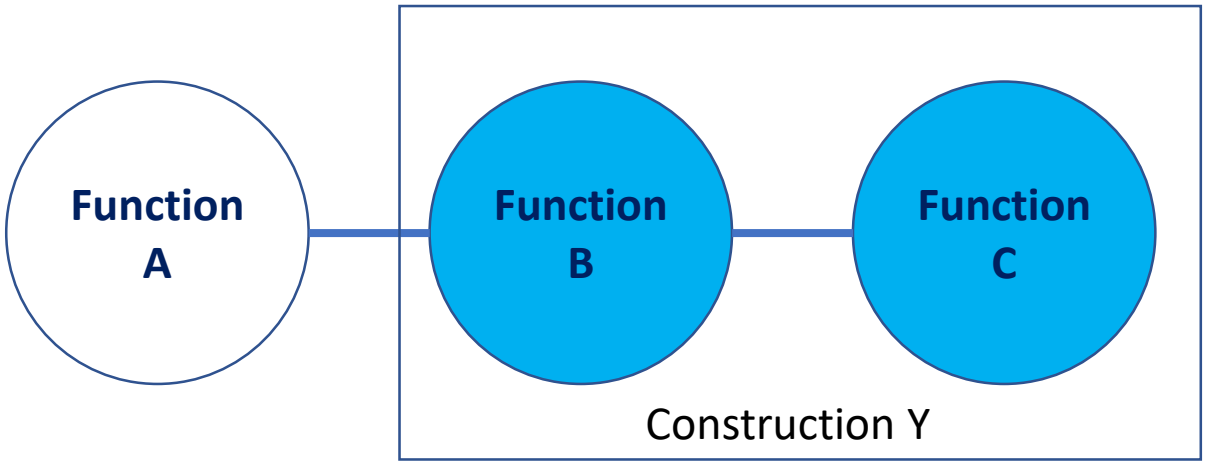
Main principles of semantic maps

- Nodes: A function is put on a map when there's at least one pair of languages which differ wrt. this function (Haspelmath 2003)
- Links: the principle of connectivity (adjacency/contiguity):
  - if a construction has more than one function, they should be connected (see van der Auwera 2013)

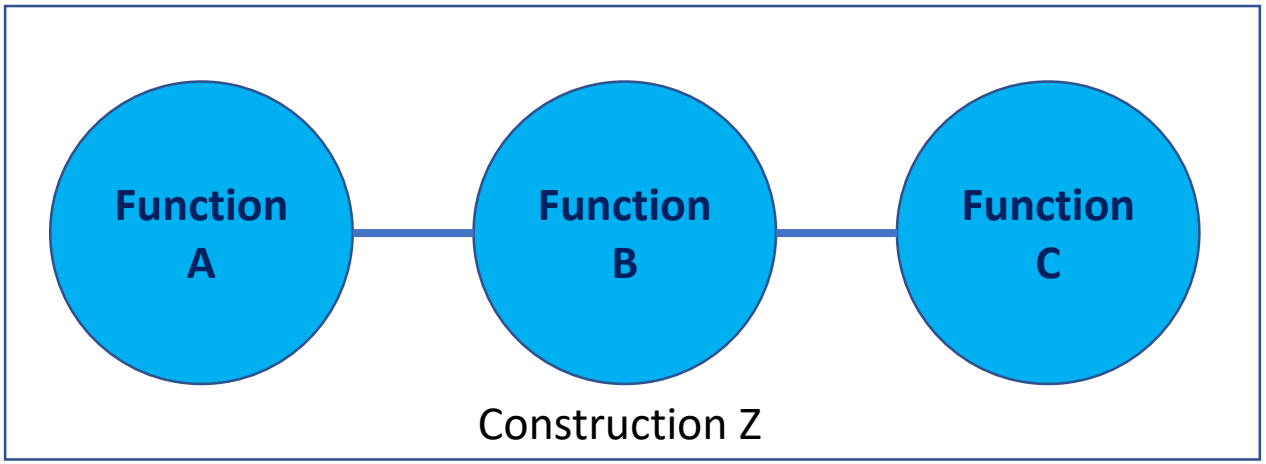
Example 1



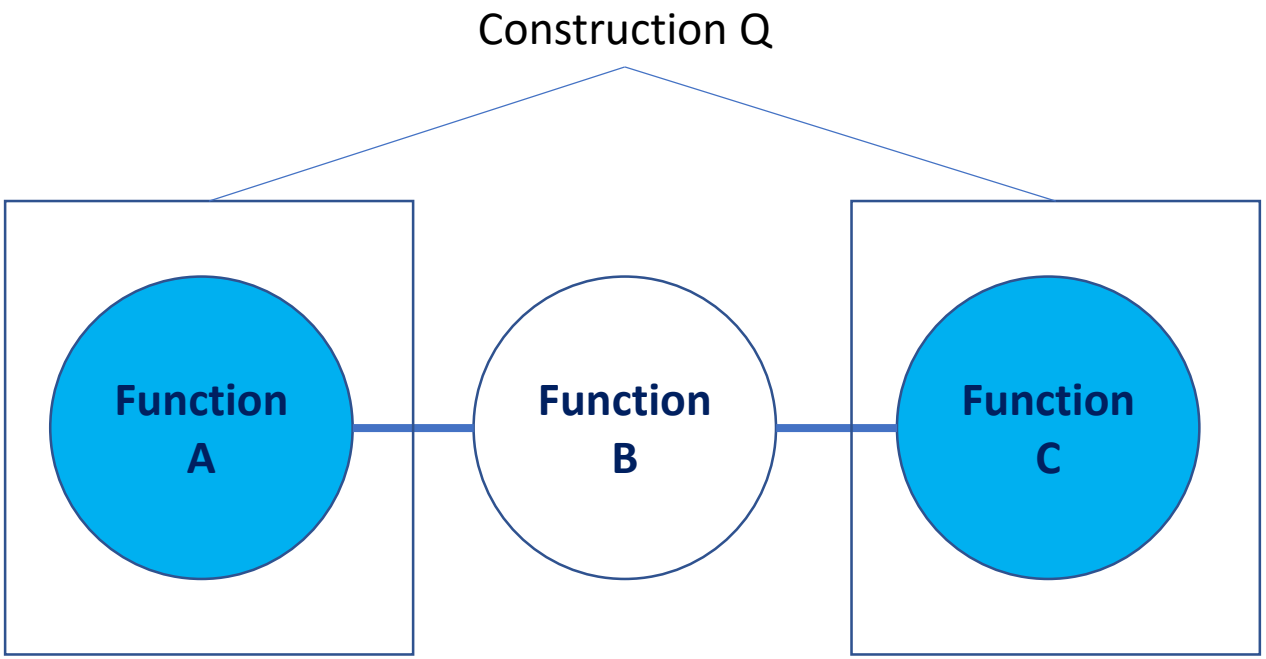
Example 2



Example 3

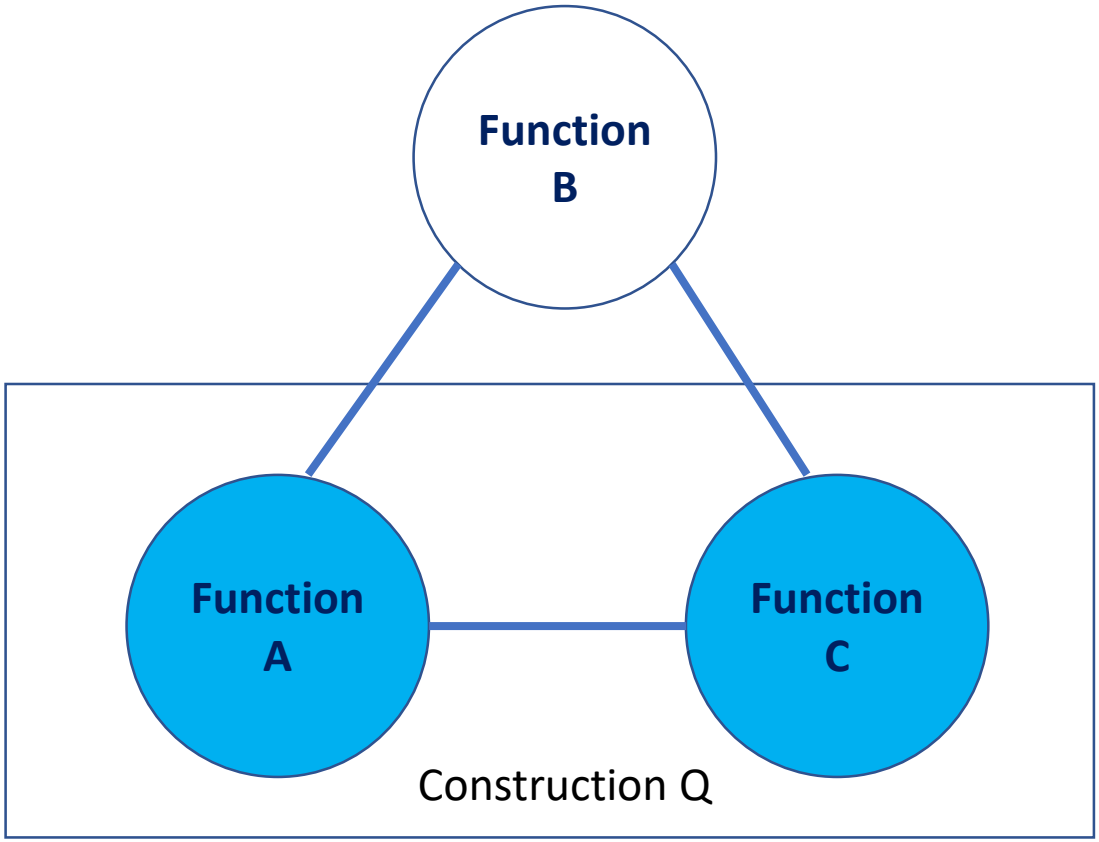


Example 4



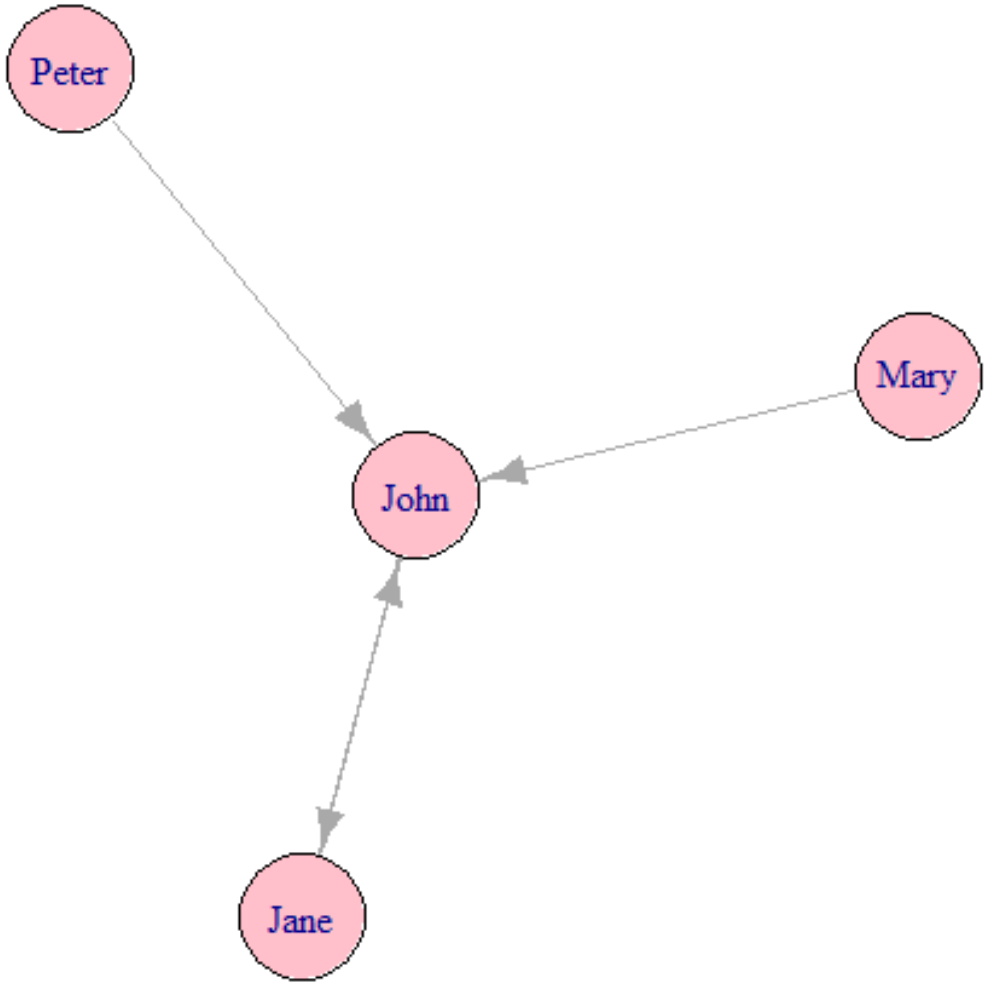
Wrong:
the connectivity principle is not observed!

A fix

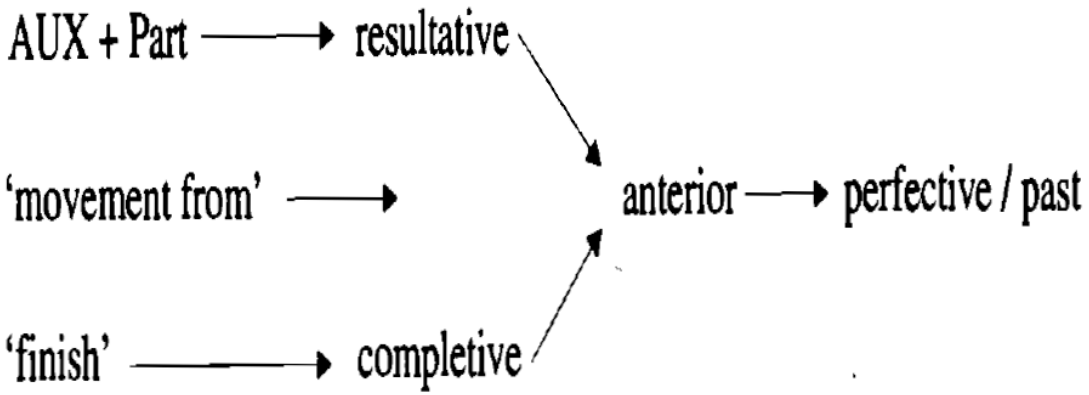


Directed graphs: feelings

- Mary loves John.
- John loves Jane.
- Jane loves John.
- And Peter fancies John, too.

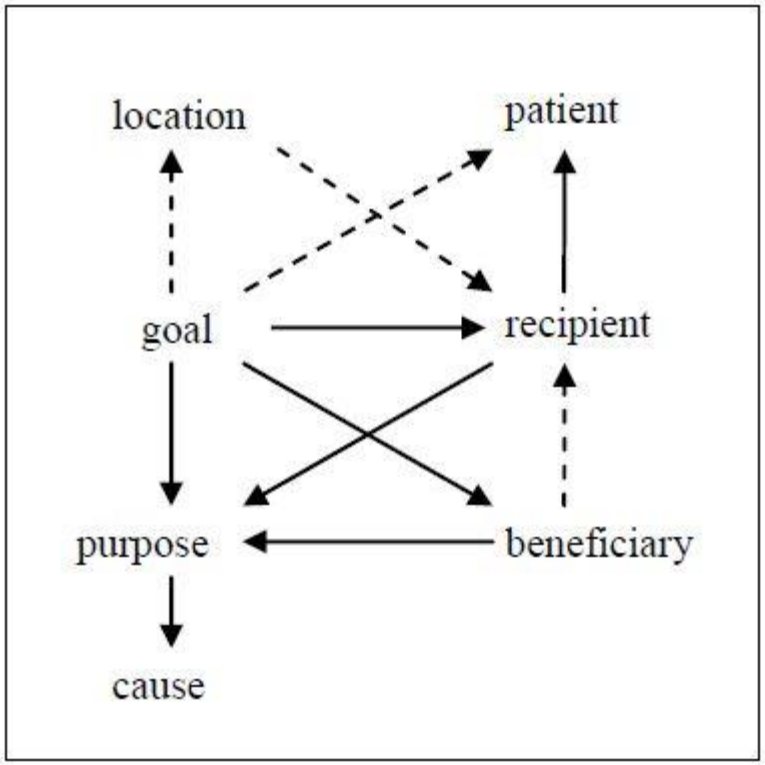


Tense and aspect grams



Bybee et al. (1994)

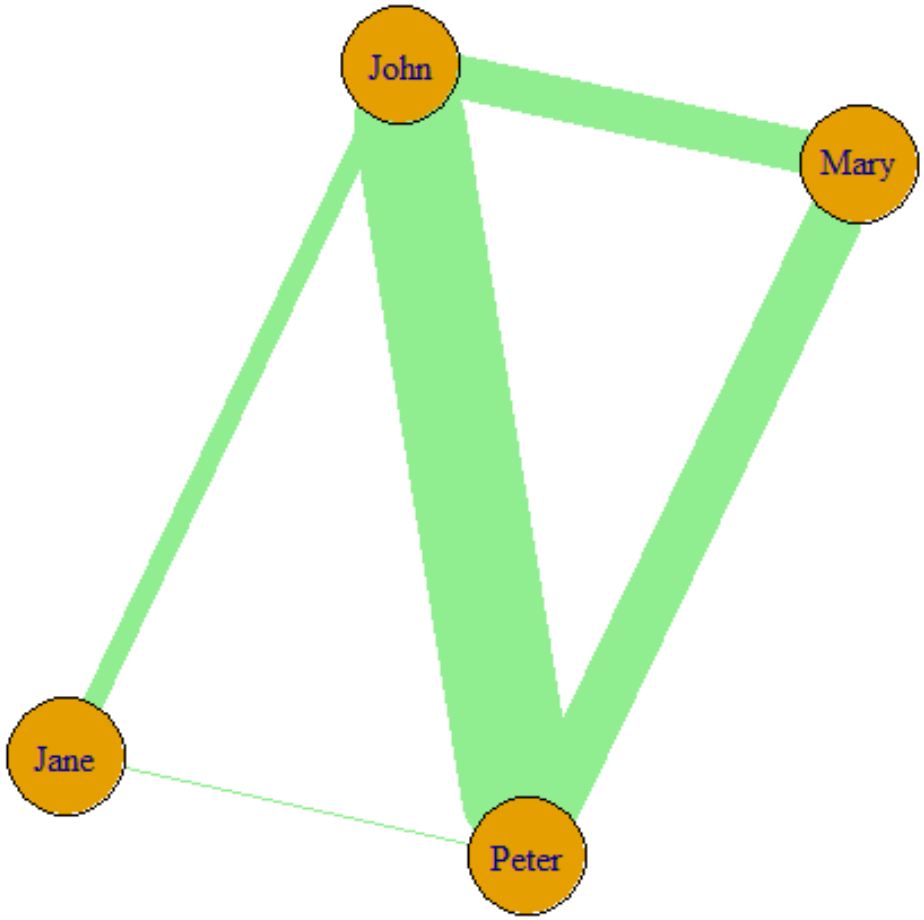
Goal-Recipient Domain



Narrog (2010)

Weighted graphs: number of pizzas eaten together

- Mary and John have eaten 20 pizzas.
- John and Jane have eaten 10 pizzas.
- Peter and John have eaten 50 pizzas.
- Mary and Peter have eaten 25 pizzas.
- Jane and Mary have eaten 0 pizzas.
- Jane and Peter have eaten 1 pizza.



CLICS: colexification patterns of HAND

Line opacity: Line weights: Coloring: Family

85 links for "arm" and "hand":

Language	Family	Form
1. Gawwada	Afro-Asiatic	harʤo
2. Hausa	Afro-Asiatic	hannu
3. Iraqw	Afro-Asiatic	dawa
4. Polci	Afro-Asiatic	aam
5. Tarifit	Afro-Asiatic	fus
6. Azerbaijani, South	Altaic	əл
7. Kumyk	Altaic	къол
8. Oroqen	Altaic	ŋa:la
9. Gurinji	Australian	wartarn
10. Vietnamese	Austro-Asiatic	tay

List et al. (2014) http://clics.lingpy.org/

Outline

1. Introduction to semantic maps

2. Case study: colexification patterns of causative markers

3. Appendix: how to create graphs in (1) with R

Typological data

- Which lexical verbs develop into causative markers?
- Data from 28 typologically diverse languages
- Examples:
  - MAKE: I **make** coffee. -> It **makes** me laugh.
  - DO: Dutch Ik **doe** wat ik wil. “I do what I want” -> Je kapsel **doet** me denken aan een vogelnest. “Your hairstyle makes me think of a bird’s nest”
  - PUT: Gumuz (gmz), Africa

    b-a-**t’oo**-gá ára ká-m-faat-ára

    AFF-3SG.TR-**put**-NFUT 1SG DAT-NMLZ-fall-1SG.INTR

    ‘She made me fall (e.g. by leaving water on the floor).’

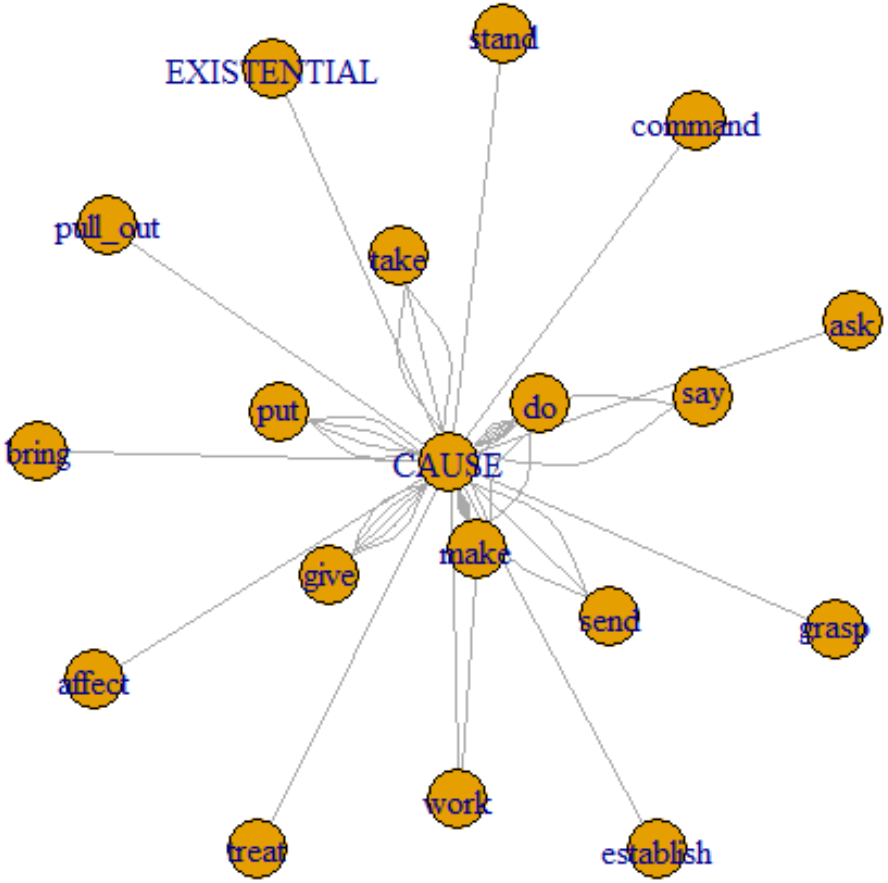
What about your language?

Data frame *colex*

```
> head(colex)
  ISO Colex1 Colex2
1 jup   make  CAUSE
2 jup   work  CAUSE
3 jup   make   work
4 jup   take  CAUSE
5 jup   send  CAUSE
6 jup  stand  CAUSE
```

Make a graph from a matrix with edges

```
> library(igraph)
> colex_graph <-
graph_from_edgelist(as.matrix(colex[, -1]),
directed = FALSE)
> plot(colex_graph)
```


stand
EXISTENTIAL
command
pull_out
take
ask
say
do
put
bring
CAUSE
make
give
send
grasp
affect
work
treat
establish

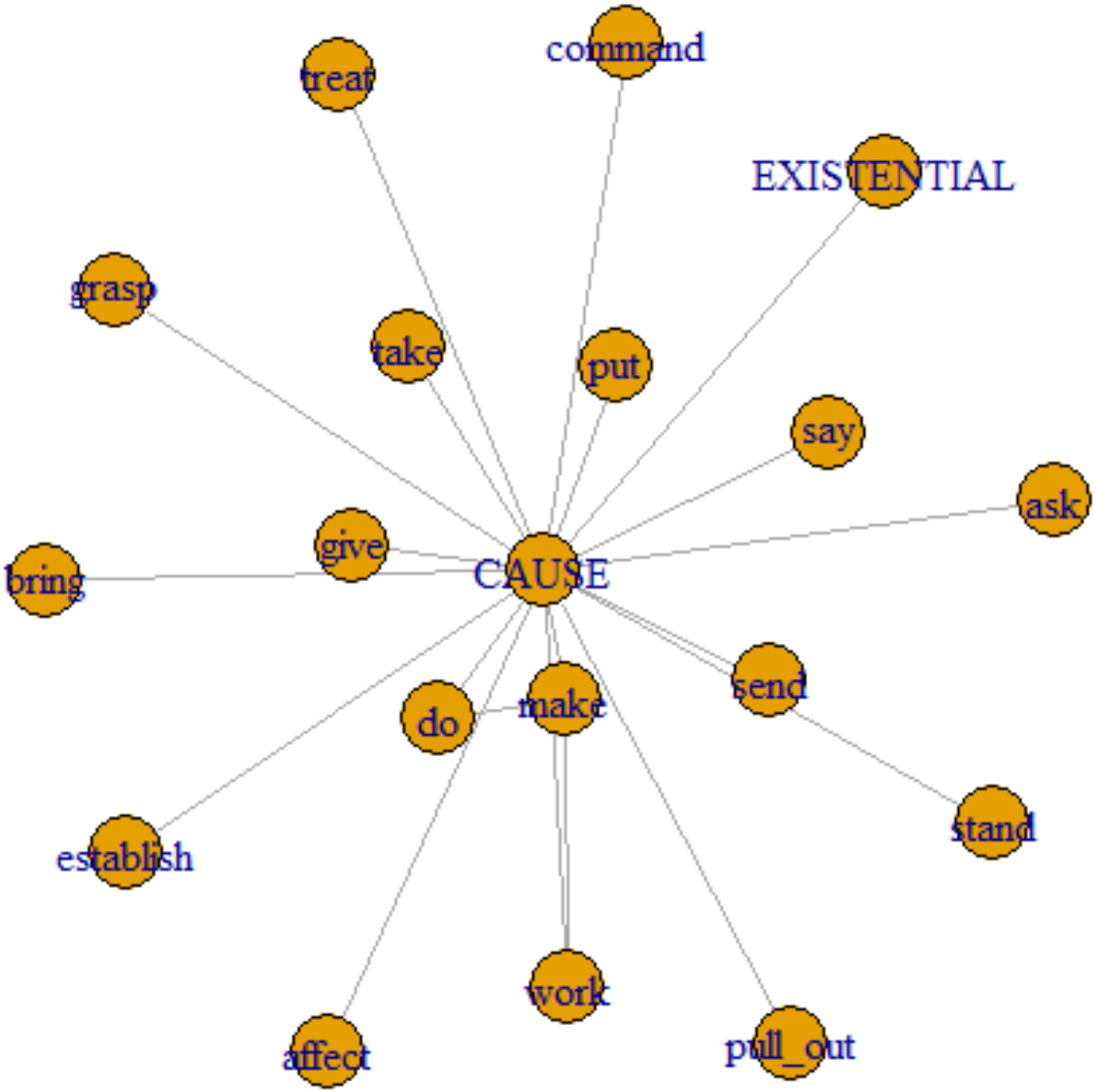
Turn the graph into an adjacency matrix

```
> colex_adj <- as_adj(colex_graph)
> colex_adj
19 x 19 sparse Matrix of class "dgCMatrix"
   [[ suppressing 19 column names ‘make’, ‘CAUSE’, ‘work’ ... ]]

make          . 15 1 . . . . . . . . . . . 2 . . .
CAUSE        15  . 1 3 3 1 1 4 5 1 1 1 1 6 1 2 1
work          1  1 . . . . . . . . . . . . . . .
…
```

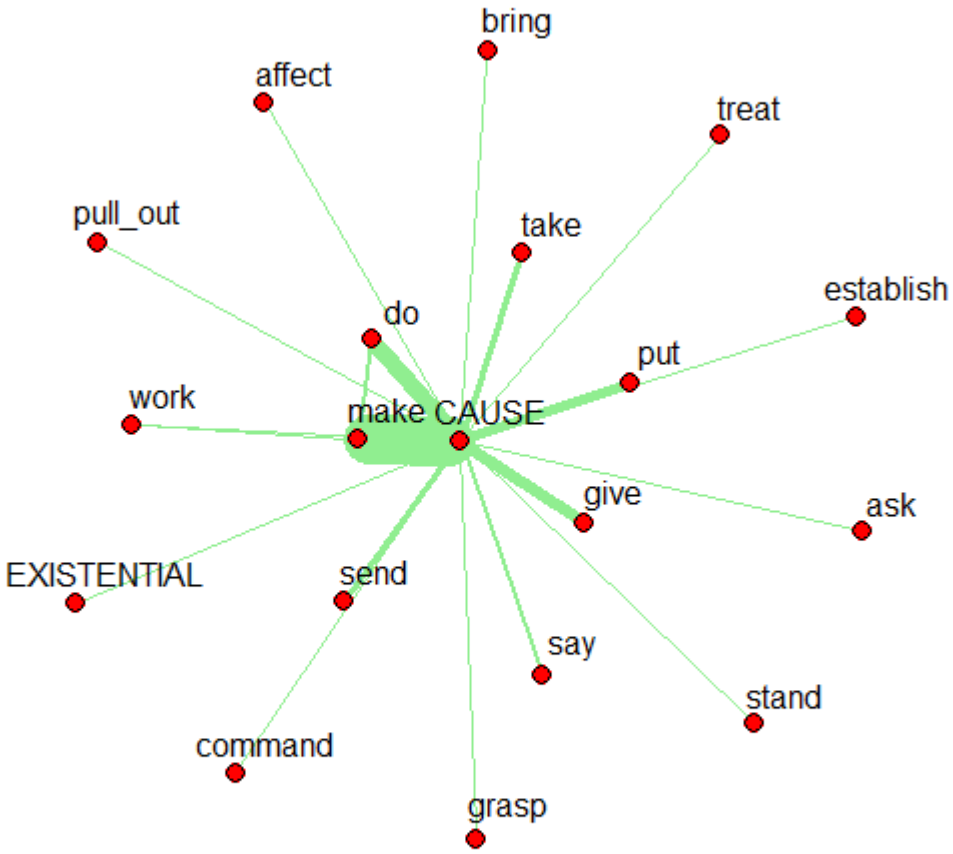
Make a graph again

```
> colex_graph1 <-
graph_from_adjacency_matrix(colex_adj, mode =
"undirected", weighted = TRUE)
> E(colex_graph1)
+ 20/20 edges from 8cd2b48 (vertex names):
 [1] make --CAUSE       make --work
 [3] make --do          CAUSE--work
 [5] CAUSE--take        CAUSE--send
 [7] CAUSE--stand       CAUSE--treat
> E(colex_graph1)$weight
 [1] 15  1  2  1  3  3  1  1  4  5  1  1  1  1
[15]  6  1  2  1  1  1
> plot(colex_graph1)
```


command
treat
EXISTENTIAL
grasp
take
put
say
ask
give
bring
CAUSE
send
do
make
stand
establish
work
affect
pull_out

Make your graph pretty

```
> plot(colex_graph1, edge.width =
E(colex_graph1)$weight^1.2, edge.color =
"lightgreen", vertex.label.cex = 1, vertex.size =
5, vertex.label.color = "black",
vertex.label.family = "sans", vertex.color =
"red", vertex.label.dist = 1.5)
```


bring
affect
treat
pull_out
take
establish
do
put
work
make CAUSE
give
ask
EXISTENTIAL
send
say
stand
command
grasp

Outline

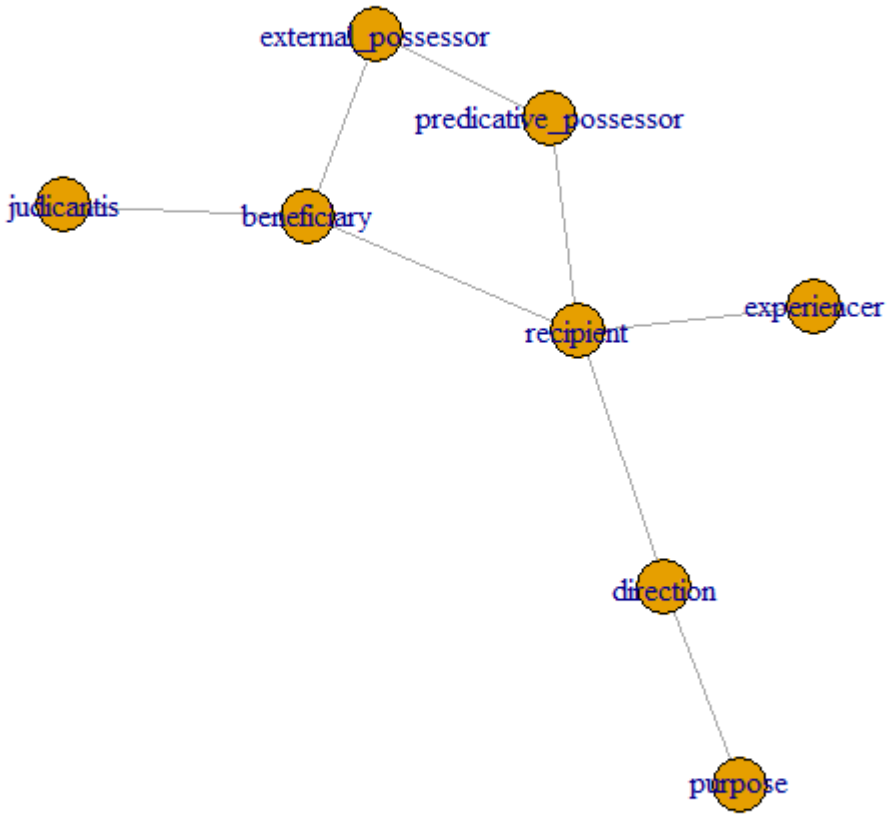
1. Introduction to semantic maps

2. Case study: colexification patterns of causative constructions

3. Appendix: how to create graphs in (1) with R

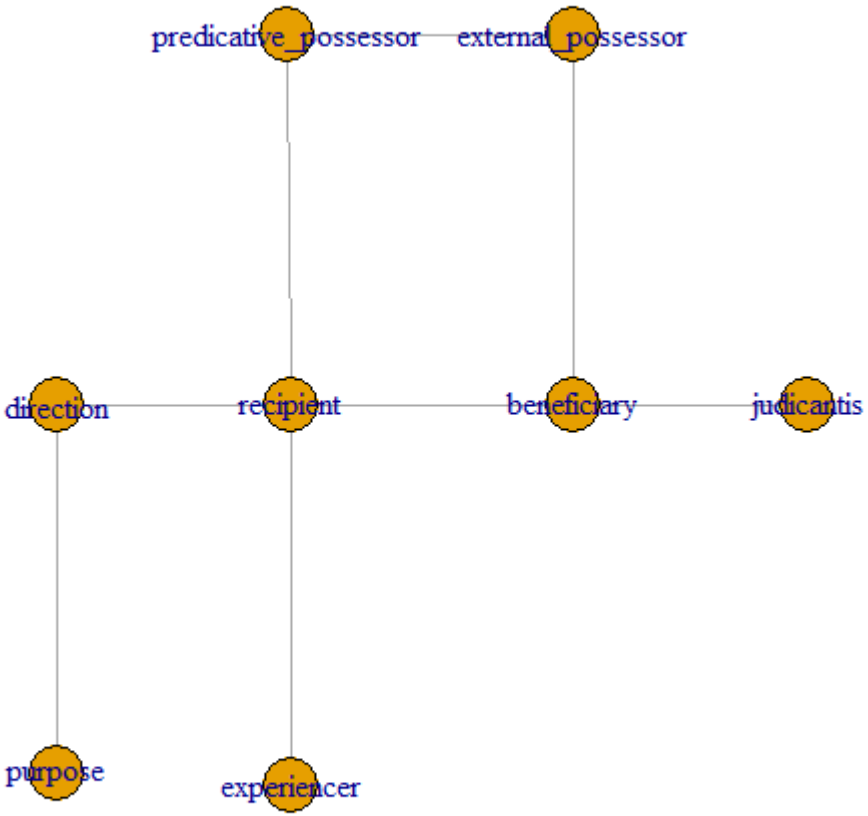
Making undirected graphs with R (datives)

```
> dat_graph <- make_graph(~ purpose - direction - recipient - beneficiary -  judicantis, experiencer - recipient - predicative_possessor - external_possessor - beneficiary) #one way of providing the edges
> plot(dat_graph)
```


external_possessor
predicative_possessor
judicantis
beneficiary
recipient
experiencer
direction
purpose

Edit your graph interactively

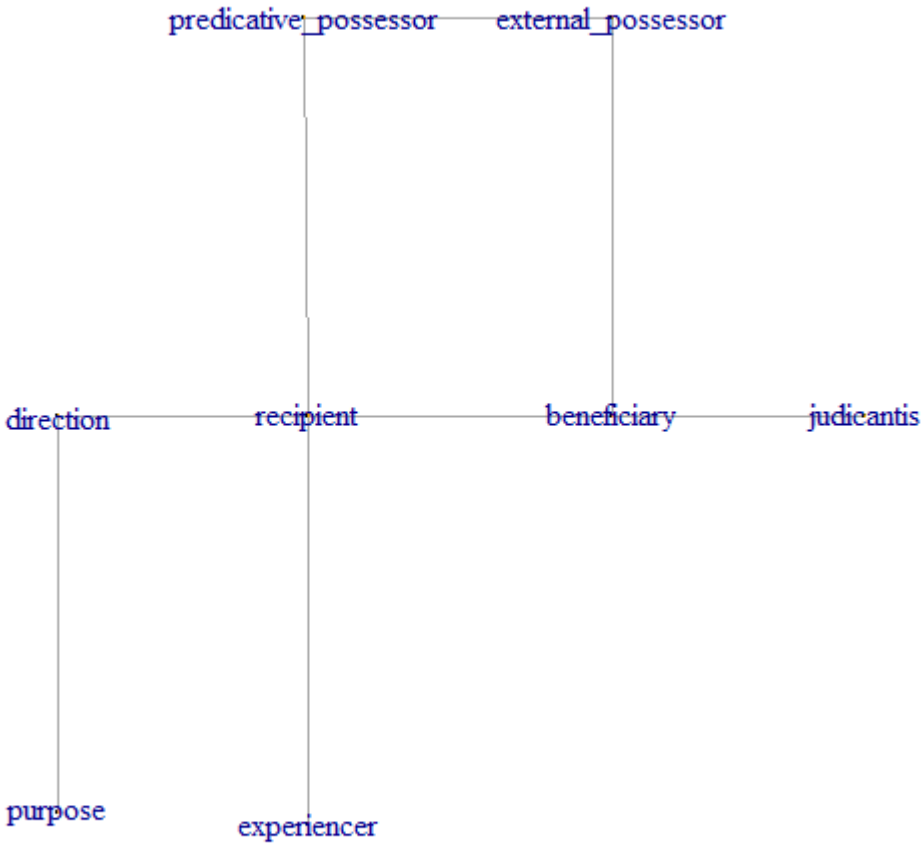
```
> tkplot(dat_graph) #call the interactive plot
[1] 1 #ID of the plotting device
> xy <- tk_coords(1) #save coordinates; change the ID, if necessary
```


predicative_possessor
external_possessor
direction
recipient
beneficiary
judicantis
purpose
experiencer

Plot your graph with new coordinates

```
> plot(dat_graph, layout = xy) #plot with the new coordinates
```

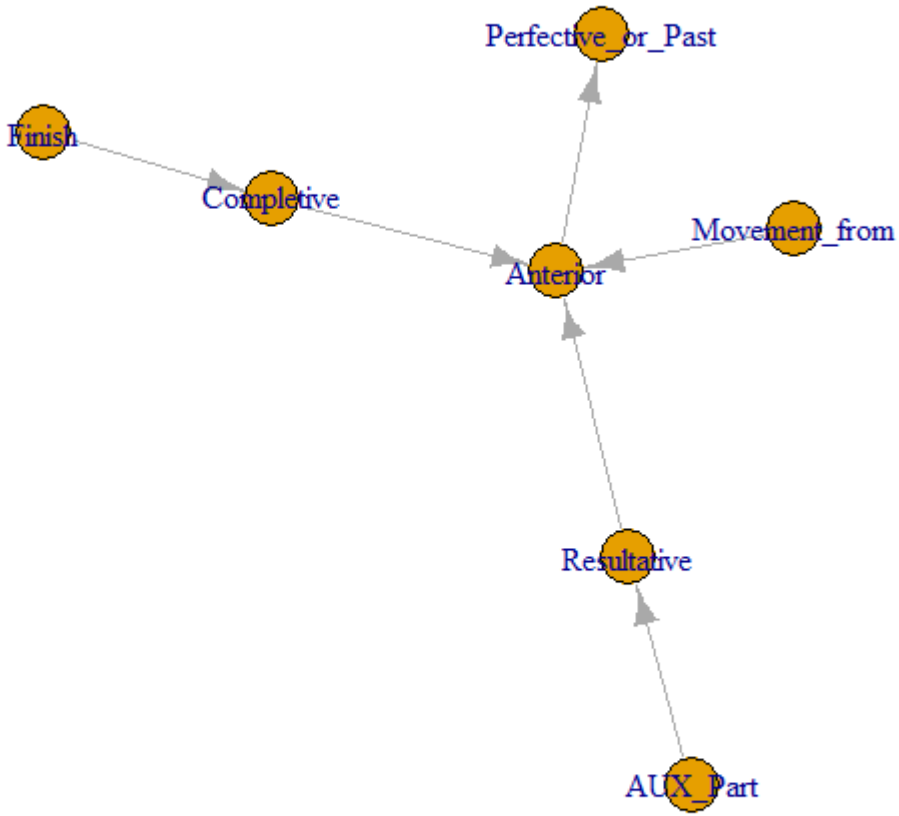
```
> plot(dat_graph, layout = xy, vertex.size = 0) #no vertex symbols
```


predicative_possessor
external_possessor
direction
recipient
beneficiary
judicantis
purpose
experiencer

Directed graphs with R (tense and aspect grams)

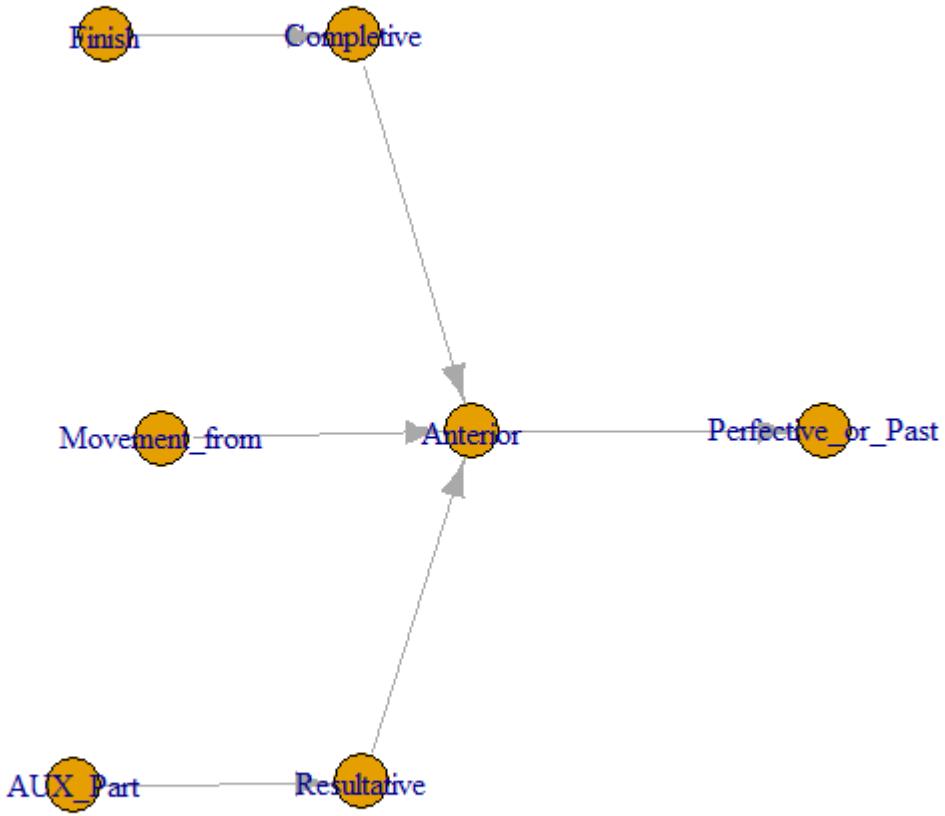
```
> ta_graph <- make_directed_graph(c("AUX_Part", "Resultative", "Resultative", "Anterior", "Anterior",  "Perfective_or_Past", "Finish", "Completive", "Completive",  "Anterior", "Movement_from", "Anterior")) #another way of entering the data
```

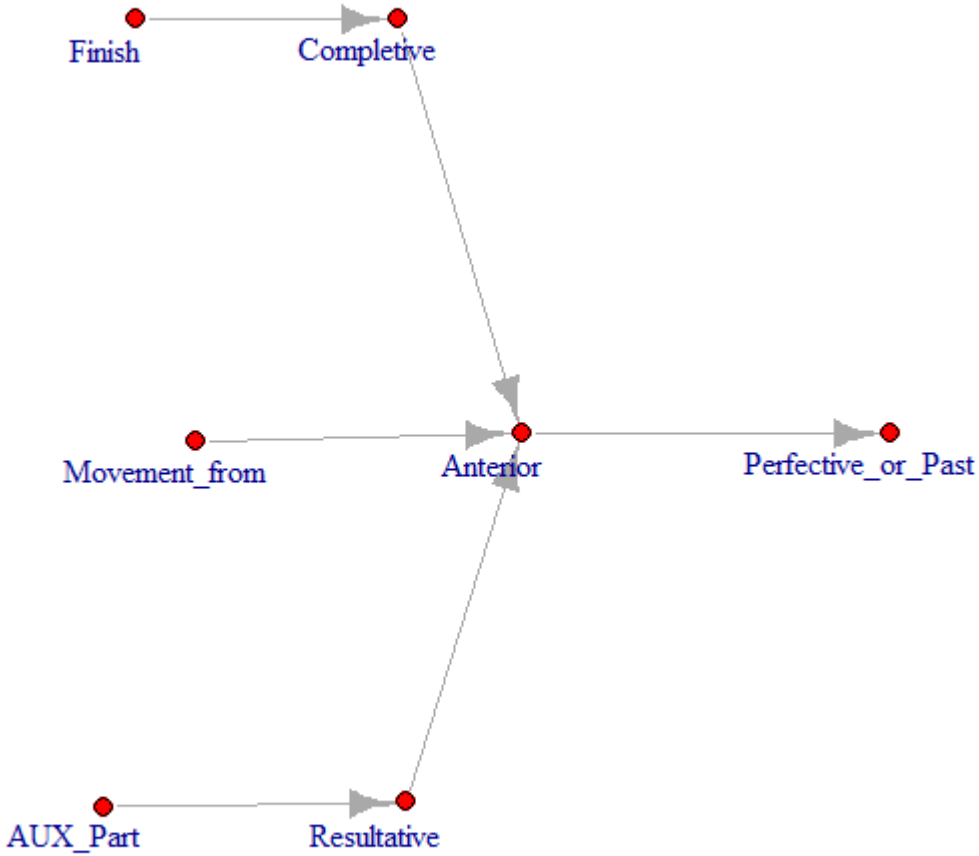
```
> plot(ta_graph)
```


Perfective_or_Past
Finish
Completive
Movement_from
Anterior
Resultative
AUX_Part

Edit the graph interactively and plot it with new coordinates

```
> tkplot(ta_graph)
> tk_coords(2) #or another id of the device
> xy <- tk_coords(2)
> plot(ta_graph, layout = xy)
> plot(ta_graph, layout = xy, vertex.size = 5,
vertex.label.dist = -1.5, edge.arrow.size= 1.2,
vertex.color = "red") #a prettier version
```


Finish
Completive
Movement_from
Anterior
Perfective_or_Past
AUX_Part
Resultative


Finish
Completive
Movement_from
Anterior
Perfective_or_Past
AUX_Part
Resultative

References

- Haspelmath, M. (2003) The geometry of grammatical meaning: Semantic maps and cross-linguistic comparison. In Tomasello, Michael (ed.), *The new psychology of language*, vol. 2. Mahwah, NJ: Lawrence Erlbaum, 211-242.
- Bybee, J. L., R. Perkins & W. Pagliuca. 1994. *The Evolution of Grammar: Tense, Aspect, and Modality in the Languages of the World*. Chicago/London: The University of Chicago Press.
- van der Auwera, J. (2013) Semantic maps, for synchronic and diachronic typology. In A. Giacalone Ramat, C. Mauri & P. Molinelli (eds.), *Synchrony and diachrony: a dynamic interface*. Amsterdam: Benjamins, 153-176.