**Text Plots**

## Jan Wijffels

**Abstract**

The textplot R package allows one to visualise complex relations in texts. This is done by providing functionalities for displaying text co-occurrence networks, text correlation networks, dependency relationships as well as text clustering. In this vignette, some example visualisations of these are shown.

*Keywords*: Text, network, co-occurrence, correlation, text clustering, dependency parsing, visualisation.

# General

## 1.1. Overview

The package allows you to visualise

* Text frequencies
* Text correlations
* Text cooccurrences
* Text clusters
* Text embeddings
* Dependency parsing results

### *Source code repository*

The source code of the package is on github at <https://github.com/bnosac/textplot>. The R package is distributed under the GPL-2 license.

# Example visualisations

## Dependency Parser

### *Example 1*

This example visualises the result of a text annotation which provides parts of speech tags and dependency relationships.

**library**(textplot) **library**(udpipe) **library**(ggraph) **library**(ggplot2) **library**(igraph)

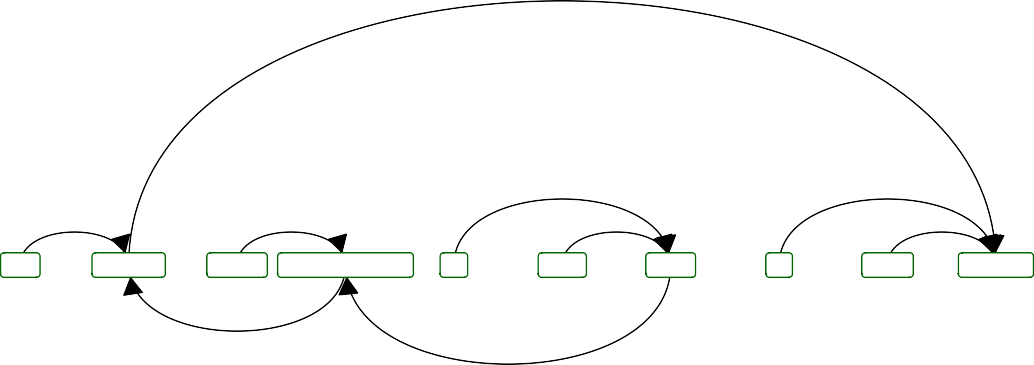
x <- **udpipe**("His speech about marshmallows in New York is utter bullshit", "english")

plt <- **textplot\_dependencyparser**(x, size = 4) plt

#### Dependency Parser

tokenisation, parts of speech tagging & dependency relations

nsubj



nmod:poss

case

case

compound

cop

amod

**His**

PRON

**speech**

NOUN

**about marshmallows**

ADP

nmod

NOUN

**in**

ADP

**New York**

PROPN PROPN

**is**

AUX

**utter**

ADJ

**bullshit**

NOUN

nmod

### *Example 2*

The following visualisation displays the dependency parser results on some larger sentence. Note that this function works only on 1 sentence.

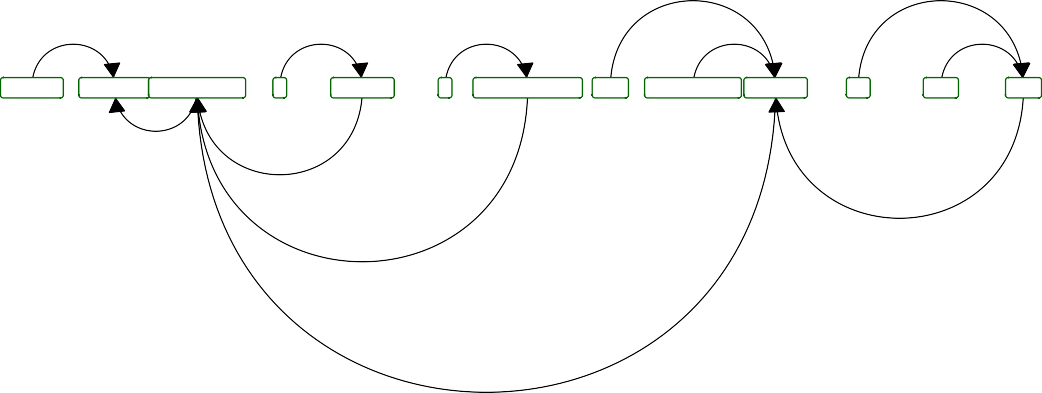
x <- **udpipe**("UDPipe provides tokenization, tagging, lemmatization and dependency parsing of raw text", "english")

plt <- **textplot\_dependencyparser**(x, size = 4) plt

**Dependency Parser**

tokenisation, parts of speech tagging & dependency relations

cc case



nsubj

punct

punct

compound

amod

**UDPipe provides tokenization , tagging**

**, lemmatization and dependency parsing of**

**raw**

**text**

PROPN VERB obj NOUN PUNCT NOUN PUNCT NOUN CCONJ NOUN NOUN ADP ADJ NOUN

conj

nmod

conj

conj

## Biterm Topic Model plots

### *Example 1*

This example shows plotting a biterm topic model which was pretrained and put in the package as an example.

**library**(BTM) **library**(ggplot2) **library**(ggraph) **library**(ggforce) **library**(concaveman) **library**(igraph)

**data**(example\_btm, package = 1textplot1) model <- example\_btm

plt <- **plot**(model, title = "BTM model", top\_n = 5) plt

BTM model

optimization plot

set

distance

base

fuzzy

partitioning

trereegression

recculrasisvesification

algorithm

high

dimensional

local

rule

curve

stochastic

descent

gradient

value

corpora

implementation

introduction

framework

feed

forest

forests

file

netwonrkeural

random

content

data

html

deep

learning

mining text

languagenatural

allocation

lda

learn

machine

lasso fit

latent

topic

variable

selection

string

clustering

path

match

function

provide

linear

model bayesian

additive

use

object

frequency

source

library

package interface

api

word

analysis

document

accelerate

base

descent

gradientalgorithm

stochastic

plt <- **plot**(model, title = "Biterm topic model", subtitle = "Topics 2 to 8", which = 2:8, top\_n = 7)

plt

Biterm topic model

Topics 2 to 8

expression

api

package

library

source

interface

value

curve

coefficient

graph

confidence

interval

use

data

low

introduction

generalize

additivme odel

linear

bayesianregression

cox

tokenization

language

natural

provide

text

mining

framework

random forest

breiman forests

implementation

survival

### *Example 2*

This example shows building a biterm topic model on nouns, adjectives and proper nouns occurring in the neighbourhood of one another and next plotting this model.

E)

**library**(data.table) **library**(udpipe)

*## Annotate text with parts of speech tags*

**data**("brussels\_reviews", package = "udpipe")

anno <- **subset**(brussels\_reviews, language %in% "nl")

anno <- **data.frame**(doc\_id = anno$id, text = anno$feedback, stringsAsFactors = FALS anno <- **udpipe**(anno, "dutch", trace = 10)

*## Get cooccurrences of nouns / adjectives and proper nouns*

biterms <- **as.data.table**(anno)

biterms <- biterms[, **cooccurrence**(x = lemma,

relevant = upos %in% **c**("NOUN", "PROPN", "ADJ"), skipgram = 2),

by = **list**(doc\_id)]

**library**(BTM) **library**(ggplot2) **library**(ggraph) **library**(ggforce) **library**(concaveman) **library**(igraph)

*## Build the BTM model*

**set.seed**(123456)

x <- **subset**(anno, upos %in% **c**("NOUN", "PROPN", "ADJ")) x <- x[, **c**("doc\_id", "lemma")]

model <- **BTM**(x, k = 5, beta = 0.01, iter = 2000, background = TRUE, biterms = biterms, trace = 100)

plt <- **plot**(model) plt

Biterm topic model

rustig

gezellig

buurt

wijk

restauranletuk

cafe

zeker

absoluut

aanrader

groet

echt

ervaring

één

Brussel

prima

erg

gastvrij

aardig

gastvrouw

behulpzaahmeverilendelijk

mooi

groot

kamer

schoon

appartement

ruim

goed

## Biterm relationships

### *Example showing objects of verbs and adjectives modifying nouns*

The below example shows the objects of verbs as well as which adjectives modify nouns. These are displayed as 2 clusters. We start from the annotation of the AirBnB data shown in the previous section [2.2.2](#_bookmark0).

**library**(BTM) **library**(ggplot2) **library**(ggraph) **library**(ggforce) **library**(concaveman) **library**(igraph) **library**(data.table) **library**(udpipe)

x <- **merge**(anno, anno,

by.x = **c**("doc\_id", "paragraph\_id", "sentence\_id", "head\_token\_id"), by.y = **c**("doc\_id", "paragraph\_id", "sentence\_id", "token\_id"),

all.x = TRUE, all.y = FALSE, suffixes = **c**("", "\_parent"), sort = FALSE) x <- **subset**(x, dep\_rel %in% **c**("obj", "amod"))

x$topic <- **factor**(x$dep\_rel) topiclabels <- **levels**(x$topic) x$topic <- **as.integer**(x$topic)

*## Construct biterms/terminology inputs to the plot*

biterms <- **data.frame**(term1 = x$lemma, term2 = x$lemma\_parent,

topic = x$topic, stringsAsFactors = FALSE) terminology <- **document\_term\_frequencies**(x, document = "topic",

term = **c**("lemma", "lemma\_parent")) terminology <- **document\_term\_frequencies\_statistics**(terminology) terminology <- terminology[**order**(terminology$tf\_idf, decreasing = TRUE), ] terminology <- terminology[, **head**(.SD, 50), by = **list**(topic = doc\_id)] terminology <- **data.frame**(topic = terminology$topic,

token = terminology$term,

probability = 1, stringsAsFactors = FALSE) plt <- **textplot\_bitermclusters**(terminology, biterms,

labels = topiclabels,

title = "Objects of verbs and adjectives-nouns", subtitle = "Top 50 by group")

plt

Objects of verbs and adjectives−nouns

Top 50 by group

be

vinden

zich

krijgeunitleggen

zien

verkennenrest

aanbevelen

uitstekend ideaal

zelf

gebruiken weten

raden

aanraden

het

wijk

rustig

echt

groot ligging

vinden

zoeken

markt geweldigomgeving

dit

ook

slaapkamer

mooi

gezellig

dat

leuk

buren

doen

willen

mij

brengen

schoon

ruim

goed

fijn

wachten

waar

matrasheel

vriendelijk douche

alleen

man

gastvrij

heerlijk

ons

net

gaan

helpen

konden

laten

geven

betreffen

auto

me

overhandigen

midden

**amod**

**obj**

## Bar plots

### *Example showing frequency of adjectives*

The plot below shows a simple barplot which works on the output of table.

**library**(udpipe)

**data**("brussels\_reviews\_anno", package = "udpipe")

x <- **subset**(brussels\_reviews\_anno, xpos %in% "JJ") x <- **sort**(**table**(x$lemma))

plt <- **textplot\_bar**(x, top = 20,

panel = "Adjectives", xlab = "Frequency", col.panel = "lightblue", cextext = 0.75, addpct = TRUE, cexpct = 0.5)

plt

goed agreable



Adjectives

195

(2.3%)

239

(2.8%)

152

1(15.80%)

138 (1.8%)

(11.36%8)

130(1.6%)

118 (1.5%)

11(21.4%)

10(18.3%)

1(10.37%)

10(11.3%)

9(18.2%)

9(15.2%)

9(12.1%)

(91.01%)

8(17.1%)

8(16%)

8(15%)

(1%)

173

(2%)

bon mooi bueno leuk erg amable limpio ruim

agradable

prima comodo schoon calme tranquilo heerlijk confortable

propre vriendelijk

100 150 200

Frequency

## Correlation of texts

### *Top correlations above a certain threshold*

Text correlcations are interesting to see, but as there are many, the below function allows one to visualise a subset of these, the ones with the highest correlations above a certain threshold.

**library**(graph) **library**(Rgraphviz) **library**(udpipe)

dtm <- **subset**(anno, upos %in% "ADJ")

dtm <- **document\_term\_frequencies**(dtm, document = "doc\_id", term = "lemma") dtm <- **document\_term\_matrix**(dtm)

dtm <- **dtm\_remove\_lowfreq**(dtm, minfreq = 5)

**textplot\_correlation\_lines**(dtm, top\_n = 25, threshold = 0.01, lwd = 5, label = TRUE)

positief 00..33 nodig

prop

gratis 00..4444parkeren

gelukkig00..2299 extra

heel

druk

00..3355

eigen

blij

00..2288 00..2299

00..3355

handig

00..3333

jammer00..44

00..2277

precies

complee00t..3311

bijzonde00r..2277

hoog

klaar

klein

open

00..3377

00..2277

00..3322persoonlijk

ontzettend

bereikba00a..22r99openbaa00..22r77

park

### *Correlations which are non-zero after fitting a glasso model*

If you have text correlations, you can also fit a glasso model on it. This puts non-relevant correlations to zero, allowing one to plot the correlations in a straightforward way.

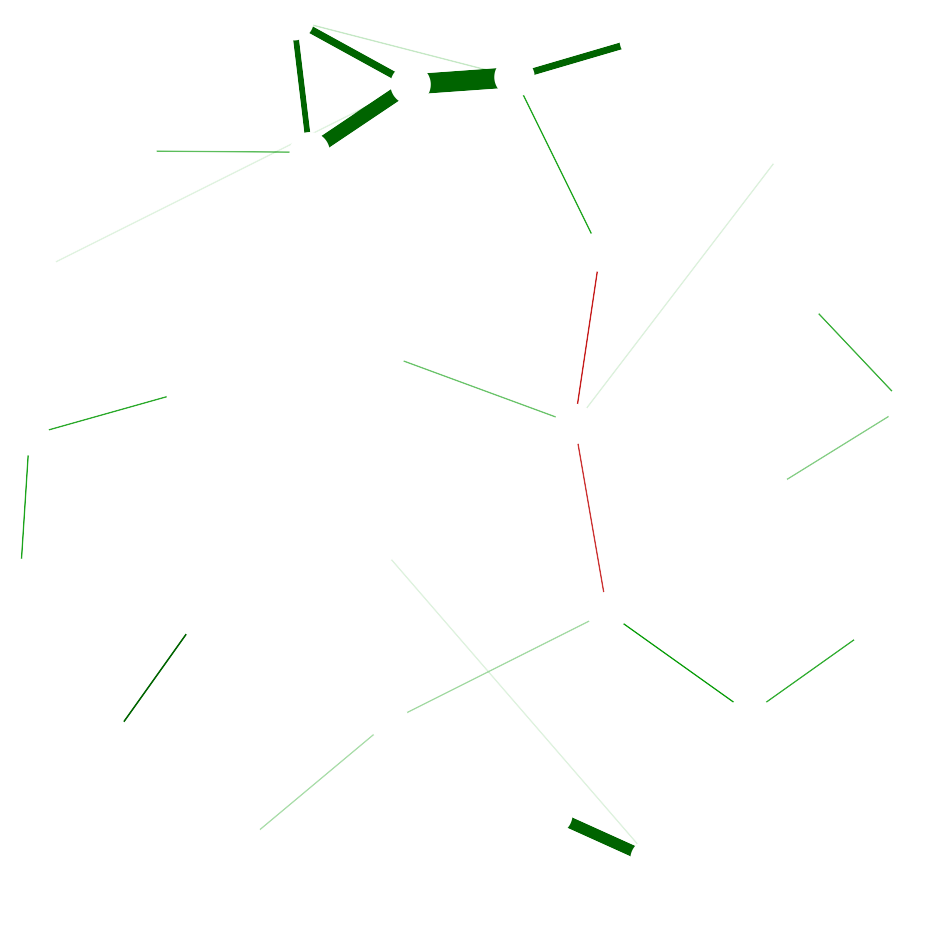
**library**(glasso) **library**(qgraph) **library**(udpipe)

dtm <- **subset**(anno, upos %in% "NOUN")

dtm <- **document\_term\_frequencies**(dtm, document = "doc\_id", term = "token") dtm <- **document\_term\_matrix**(dtm)

dtm <- **dtm\_remove\_lowfreq**(dtm, minfreq = 20) dtm <- **dtm\_remove\_tfidf**(dtm, top = 100) term\_correlations <- **dtm\_cor**(dtm)

**textplot\_correlation\_glasso**(term\_correlations, exclude\_zero = TRUE)



uitzicht

mensen

slaapkambaedr kamer bed

keuken

minuten

ruimte

kamer

gebruik

dagen

tips

super

appartement

vervoer

stad

centrum

tijd

restaurants

huis

aankomst

buurt

straat

contact

ligging

wijk

loopafstand

metro

## Co-occurrence of texts

### *Example showing adjectives occurring in the same document*

The following graph shows how frequently adjectives co-occur across all the documents.

**library**(udpipe) **library**(igraph) **library**(ggraph) **library**(ggplot2)

**data**(brussels\_reviews\_anno, package = 1udpipe1)

x <- **subset**(brussels\_reviews\_anno, xpos %in% "JJ" & language %in% "fr") x <- **cooccurrence**(x, group = "doc\_id", term = "lemma")

plt <- **textplot\_cooccurrence**(x,

title = "Adjective co-occurrences", top\_n = 25)

plt

**Adjective co−occurrences**

lumineux

parfait

sympathique

bel

spacieux propre

agreable

sympa

grand

calme

petit

bon

xcellent

disponible

confortable

chaleureux

### *Example showing objects of verbs / adjectives modifying nouns on our annotated dataset*

The following graph shows a similar visualisation, but instead focussing on the frequency of objects of verbs and adjectives modifying a noun. For this, we start again from the annotation of the AirBnB data shown in the section [2.2.2](#_bookmark0).

**library**(udpipe) **library**(igraph) **library**(ggraph) **library**(ggplot2) **library**(data.table)

biterms <- **merge**(anno, anno,

by.x = **c**("doc\_id", "paragraph\_id", "sentence\_id", "head\_token\_id"), by.y = **c**("doc\_id", "paragraph\_id", "sentence\_id", "token\_id"),

all.x = TRUE, all.y = FALSE, suffixes = **c**("", "\_parent"), sort = FALSE) biterms <- **setDT**(biterms)

biterms <- **subset**(biterms, dep\_rel %in% **c**("obj", "amod"))

biterms <- biterms[, **list**(cooc = .N), by = **list**(term1 = lemma, term2 = lemma\_parent)] plt <- **textplot\_cooccurrence**(biterms,

title = "Objects of verbs + Adjectives-nouns", top\_n = 75,

vertex\_color = "orange", edge\_color = "black", fontface = "bold")

plt

#### Objects of verbs + Adjectives−nouns

**uitstekend**

**voorzigeenmak**

**mogelijk verwaaramniznegtten**

**gastheer kamer**

**beslissing**

**loft**

**laat**

**aanwezig mogenreservering**

**vlot**

**nemen**

**warmevnrsiendelijk**

**mooi feest**

**dit**

**bord**

**verloop**

**één**

**fijne**

**verhuurder er**

**mwuijklticultureel**

**straartustig**

**etage**

**sleutel**

**omgevinsgtil prachtig**

**aanrader zeker**

**Europees**

**minute secondleast**

**konden**

**bezorgen bieden**

**inrichten**

**eventueel**

**eigencafe**

**ruim**

**trein rijden**

**zelf**

**info**

**nodig**

**mogelijkhe ontmoeten**

**andergast**

**super**

**loopafstand**

**appartemefnijtn**

**huren**

**gaan**

**juist**

**Icaro**

**vinden**

**prima**

**Brussel**

**beeld**

**midden**

**dag**

**perfect teruggaan locatie plek goed**

**gezellig**

**centrum**

**gans**

**levendbiguurt leuk**

**goeie**

**aanwijziging**

**verzorgen**

**OV keuken**

**afspreken**

**geven**

**ontbijt**

**ook**

**tijd**

**nachtrust hebben**

**duidelijk instructie**

**lopen alles**

**vers**

**alleen**

**oordop**

**verblijf**

**aangenaam**

**ongeveer douche**

**jevoelen**

## Text embeddings

### *Example showing clustered text embeddings*

The following graph shows the embeddings of the top 7 words emitted by a sample of topics extracted with the Embedding Topic Modelling clustering algorithm ([https://github.com/](https://github.com/bnosac/ETM) [bnosac/ETM](https://github.com/bnosac/ETM)).

The embeddings are mapped onto a 2-dimensional space using UMAP.

**library**(uwot) **set.seed**(1234)

*## Put embeddings in lower-dimensional space (2D)* **data**(example\_embedding, package = "textplot") embed.2d <- **umap**(example\_embedding,

n\_components = 2, metric = "cosine", n\_neighbors = 15, fast\_sgd = TRUE, n\_threads = 2, verbose = FALSE)

embed.2d <- **data.frame**(term = **rownames**(example\_embedding),

x = embed.2d[, 1], y = embed.2d[, 2], stringsAsFactors = FALSE)

**head**(embed.2d, n = 5)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ## |  | term | x | y |
| ## | 1 | tribunal | 2.7050211 | -1.74029937 |
| ## | 2 | noodnummers | 0.6450640 | 1.45599863 |
| ## | 3 | acs | 0.9476272 | -0.48708960 |
| ## | 4 | spi | -3.1926074 | 1.43367848 |
| ## | 5 | alert | -0.4386997 | 0.01686478 |

*## Get a dataset with words assigned to each cluster with a certain probability weight*

**data**(example\_embedding\_clusters, package = "textplot")

terminology <- **merge**(example\_embedding\_clusters, embed.2d, by = "term", sort = FALSE) terminology <- **subset**(terminology, rank <= 7 & cluster %in% **c**(1, 3, 4, 10, 15, 19, 17)) **head**(terminology, n = 10)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ## | term | cluster | rank | weight | x | y |
| ## | 1 zelfstandigen | 1 | 1 | 1.0000000 | -3.0242422 | 1.455762479 |
| ## | 5 opdeling | 1 | 2 | 0.5390060 | 0.2629381 | 2.902682874 |
| ## | 13 werkloosheid | 1 | 3 | 0.4511878 | -2.8441186 | 1.769844148 |
| ## | 16 ocmw | 1 | 4 | 0.3379358 | -2.6204746 | 1.319661769 |
| ## | 19 zelfstandige | 1 | 5 | 0.2172686 | -3.0162311 | 1.620623443 |
| ## | 21 kmo | 1 | 6 | 0.2013531 | -3.0319384 | 0.880055460 |
| ## | 23 overbruggingsrecht | 1 | 7 | 0.1851361 | -2.9195409 | 1.566569063 |
| ## | 54 vzw | 4 | 4 | 0.3867166 | -2.8023301 | 0.039034936 |
| ## | 68 pod | 4 | 3 | 0.4328151 | -1.8947673 | -1.096613792 |
| ## | 211 btw | 4 | 1 | 1.0000000 | -3.4231983 | -0.002880421 |

*## Plot the relevant embeddings*

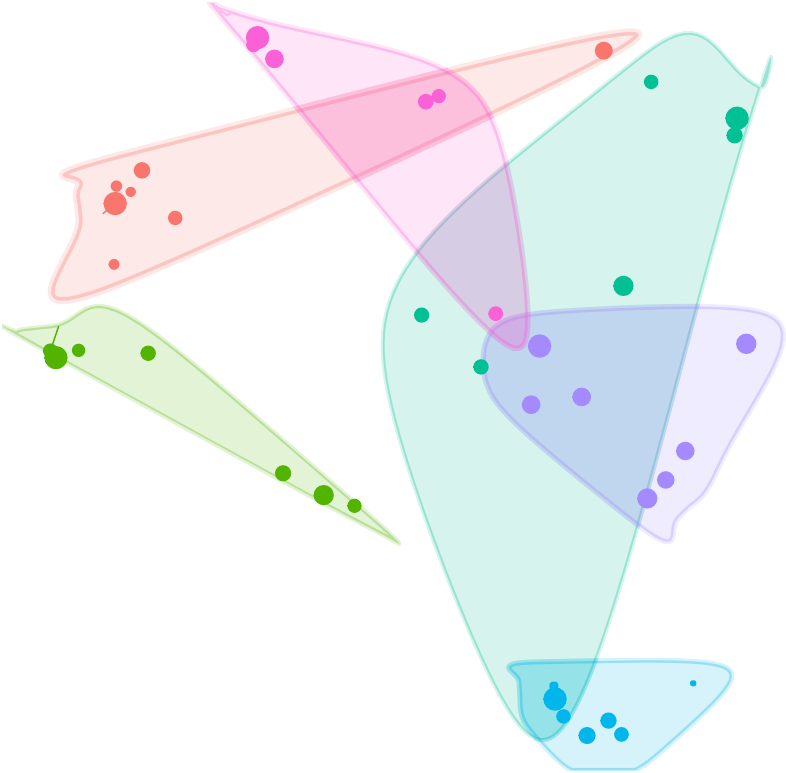
**library**(ggplot2) **library**(ggrepel) **library**(ggalt)

plt <- **textplot\_embedding\_2d**(terminology, encircle = TRUE, points = TRUE,

title = "Embedding Topic Model clusters", subtitle = "embedded in 2D using UMAP")

plt

Embedding Topic Model clusters

embedded in 2D using UMAP

tewerkstelling beperking

chronische

opdeling

handicap

tewerkgesteld

kabinet

politiezone

politiezones

gedetineerden



zelfstandige

werkloosheid

overbruggingsrecht

politieagenten gevangenissen

gevangenis

gevangenen

asielzoekers

zelfstandigen ocmw

kmo

vennootschap

roerende

rechtbanken

ombudsman

app

voertuigen

stoffen

nationaliteit

vluchtelingen

btw

vzw

beelden

persoonsgegevens

pod

kostprijs

kost

websites

geneesmiddelen

testen

mondmaskers

infrabel

nmbs

stations

nmbs

trein

luchthaven

station treinen

## Affiliation:

BNOSAC - Open Analytical Helpers E-mail: [jwijffels@bnosac.be](mailto:jwijffels@bnosac.be) URL: [http://www.bnosac.be](http://www.bnosac.be/)