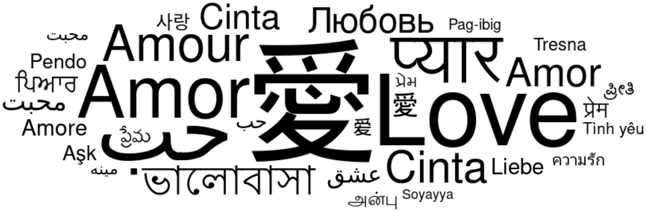
ggwordcloud

a word cloud geometry for ggplot2

E. Le Pennec

useR! - Toulouse - 11/07/2019

Word Clouds?



Text visualization tool

- **Words scaled** according to the a quantity (often a count or a proportion)
- Nice (**meaningless**) word **position**/order.

- For documents, vocabulary often pruned (stop words/rarely used words)

Word Clouds?

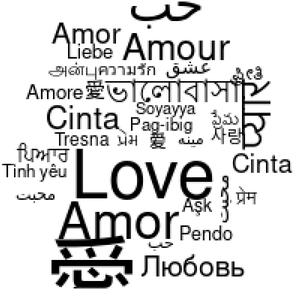
محبت
사랑
Cinta
Любовь
Pag-ibig
Amour
Pendo
ਪਿਆਰ
Amor
爱
प्यार
Tresna
Amor
પ્રેમ
愛
Love
प्रेम
Tình yêu
محبت
Amore
ప్రేమ
حب
爱
Aşk
حب
مينه
ভালোবাসা
عشق
Cinta
Liebe
ความรัก
அன்பு
Soyayya

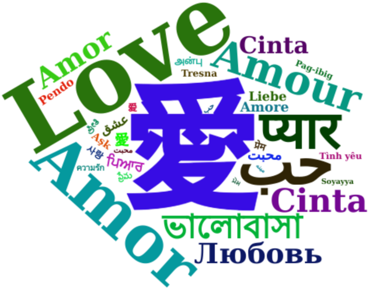
Text visualization tool

- **Words scaled** according to the a quantity (often a count or a proportion)
- Nice (**meaningless**) word **position**/order.

- For documents, vocabulary often pruned (stop words/rarely used words)

Word Clouds in R





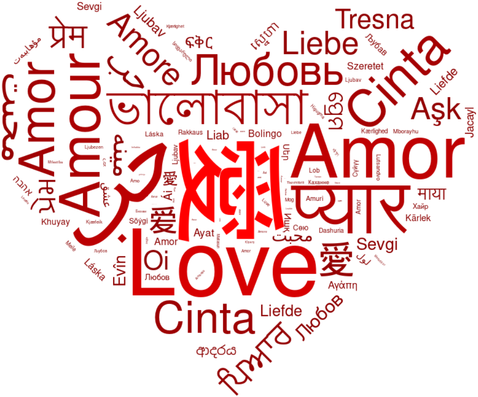
wordcloud

- **base graphics**
- fast but crude placement algorithm in R

wordcloud2

- **`html` graphics**
- placement algorithm in js
- lots of bell and whistles (arbitrary rotation, mask...)

ggwordcloud

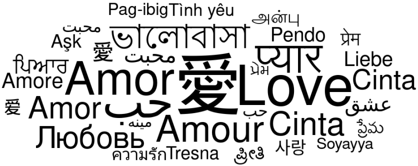


A new ggplot2 geometry

- **ggplot2 ecosystem**
- a **new geometry** similar to `geom_text` and `geom_text_repel`
- fast placement algorithm in `C`
- more functionalities than `wordcloud` and `wordcloud2`...

geom_text_wordcloud

```
library(ggwordcloud)
data("love_words_small")
set.seed(42)
ggplot(love_words_small,
       aes(label = word,
           size = speakers)) +
  geom_text_wordcloud() +
  scale_size_area(max_size = 24)
  theme_minimal()
```

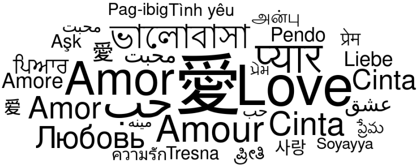


A dedicated text geometry

- geometry with the **geom_text syntax**:
  - label for the word
  - size for the count
- automatic placement without overlapping around a default $(0,0)$ position

geom_text_wordcloud

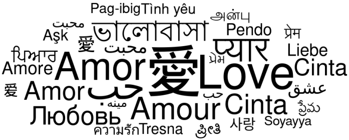
```
library(ggwordcloud)
data("love_words_small")
set.seed(42)
ggplot(love_words_small,
       aes(label = word,
           size = speakers)) +
  geom_text_wordcloud() +
  scale_size_area(max_size = 24)
  theme_minimal()
```

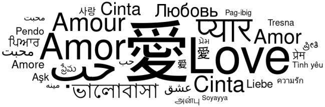


A dedicated text geometry

- geometry with the **geom_text syntax**:
  - label for the word
  - size for the count
- automatic placement without overlapping around a default $(0, 0)$ position

Text Scaling





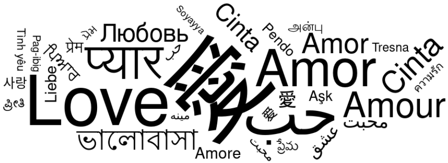
geom_text_wordcloud

- **Font size proportional to** (the square root of) the count/proportion.
- Ink area depends on the number of letters...

geom_text_wordcloud_area

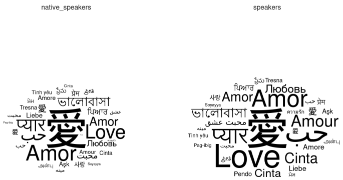
- **Ink area proportional to** the count/proportion.
- Perception not biased by number of letters.

Rotation and Facet



Rotation

- **Arbitrary rotation** with `angle` aesthetic.



Facet

- Compatible with `ggplot2` **faceting system**.

Shapes and Mask

square

Amor
Liebe
Cinta
ভালোবাসা
Pendo
Amor
爱
प्यार
Cinta
Aşk
Amore
Love
Amour
Любовь
Tresna
Soyayya
Tinh yêu
사랑

triangle-forward

Tinh yêu
عشق
Pendo
사랑
Amor
Liebe
Soyayya
ভালোবাসা
Amour
Tresna
Amore
Aşk
爱
Love
Cinta
Pag-ibig
प्यार
Amor
Cinta
Любовь

star

Любовь
Pag-ibig
Amour
Soyayya
Pendo
Amore
Liebe
Cinta
Love
爱
Amor
Amor
Tresna
प्यार
Aşk
عشق
Cinta
ভালোবাসা
Tinh yêu

Shapes

- Cloud may have **different base shapes**.

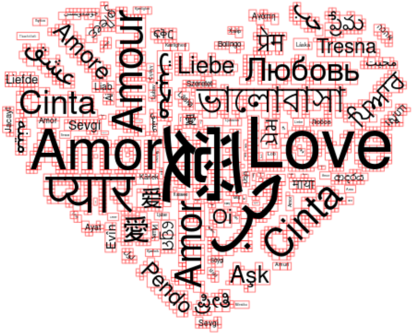
Pag-ibig
Tinh yêu
사랑
Pendo
Cinta
Love
عشق
Cinta
Amor
प्यार
爱
حب
Aşk
Tresna
Liebe
Amour
Amore
Soyayya

Mask

- Words stay within a **prescribed mask**.

- Other functionalities: color, position, fonts...

Algorithm

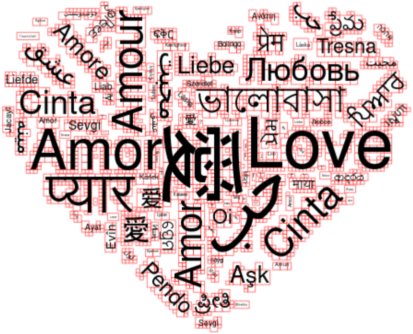


Algorithm

- Compute text area using `textGrob` and deduce font size
- Draw text using again `textGrob` and compute a mask made of small bounding boxes
- Use a fast spiraling placement algorithm (in `C++`) to place the words without any boxes overlap

- **Bottleneck:** text size computation!

Algorithm

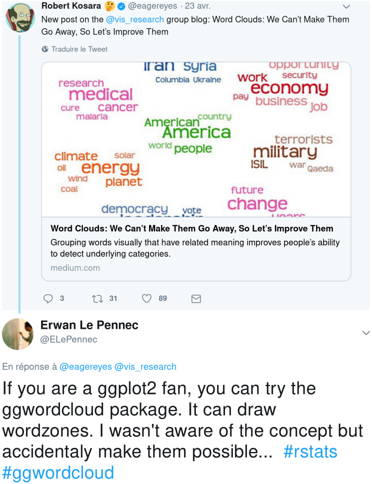


Algorithm

- Compute text area using `textGrob` and deduce font size
- Draw text using again `textGrob` and compute a mask made of small bounding boxes
- Use a fast spiraling placement algorithm (in `C++`) to place the words without any boxes overlap

- **Bottleneck:** text size computation!

Word Zones



Robert Kosara 🤔 @eagereyes · 23 avr.

New post on the @vis_research group blog: Word Clouds: We Can't Make Them Go Away, So Let's Improve Them

Traduire le Tweet

Word Clouds: We Can't Make Them Go Away, So Let's Improve Them

Grouping words visually that have related meaning improves people's ability to detect underlying categories.

medium.com

3 31 89

Erwan Le Pennec
@ELePennec

En réponse à @eagereyes @vis_research

If you are a ggplot2 fan, you can try the ggwordcloud package. It can draw wordzones. I wasn't aware of the concept but accidentaly make them possible... #rstats #ggwordcloud

Interest of a Grammar

- Can produce graphs which where not planned!

Thank you to the R ecosystem

Code and algorithm

- ggplot2: environment
- ggrepel: basis of the code
- wordcloud/wordcloud2: source of inspiration

Environment

- rstudio...
- usethis: package skeleton
- testthat: unit testing
- devtools/rhub: package dev/testing before CRAN
- pkgdown: documentation

ggwordcloud

Sevgi
प्रेम
Ljubav
Amore
ፍቅር
Liebe
Tresna
Любовь
Szeretet
Ljubav
Cinta
Liefde
Amour
Amour
ভালোবাসা
Aşk
Jacaył
Láska
Rakkaus
Liab
Bolingo
Kærlighed
Mborayhu
Amor
Ljubezen
Lob
Kaxahne
Amuri
माया
Хайр
Kärlek
Dashuria
Ayat
Söygi
Khuyay
Kjærleik
Ai 愛
愛
Amor
Oi
Любов
Evin
Love
Sevgi
愛
Αγάπη
Láska
Cinta
Liefde
Любов
ପ୍ରେମ
ආදරය
Сою
Mog
प्यार

A word cloud geometry for ggplot2

- A lovely, easy to use and full of functionalities package.
- Available on CRAN.
- Source and bug tracker at `https://github.com/lepennec/ggwordcloud`
- **Website:** `https://lepennec.github.io/ggwordcloud/`