

# LEXpander: applying colexification networks to automated lexicon expansion

*Anna Di Natale, David Garcia*

*5/2022*

Loading useful packages and libraries

```
library(plyr) #version 1.8.4
library(dplyr) #for data transformations, version 1.0.7

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:plyr':
##
##   arrange, count, desc, failwith, id, mutate, rename, summarise,
##   summarize

## The following objects are masked from 'package:stats':
##
##   filter, lag

## The following objects are masked from 'package:base':
##
##   intersect, setdiff, setequal, union

# library(lsa) ##for the word embedding model, version 0.73.2
# library(text2vec) ##for the GloVe embedding model, version 0.6
library(ggplot2) ##for plots, version 3.3.2
library(reticulate) #for working with Python, version 1.16
library(stringr) ##fro dealing with strings, version 1.3.1
```

Loading useful functions

```
##Set python path, python 3.8 required for running the VADER script
use_python("/usr/bin/python3.8", required = T) #INSERT YOUR PYTHON 3.8 path
source('scripts/expand_wordlist.R') #word lists expansion algorithms
source('scripts/cor_on_texts.R') ##text analysis task
source('scripts/random_wordlist.R') ##computation of the baseline models
source('scripts/count_string.R') ##counts words in a string
source('scripts/compute_correlation.R') ##function for the correlation of text analysis tasks
```

To run the lexicon expansion algorithms on the EVs:

```
method<-'wordnet'
# res<-expand_wordlist_EV(method)
# saveRDS(res,paste0('results/EV_2015en_',method,'.Rda'))
```

Computation of the baseline method. Returns the random word lists. It might take some time

```
# for (i in seq(1,1000)) ##setting the number of repetitions
# {baseline<-random_wordlist('wordnet','EV')}
```

Table 2: Precision, recall and F1 of the expanded lexica on LIWC 2015 English)

```

perc<-"0.3" ##choose the percentage of seed words from LIWC
report_df<-data.frame(stringsAsFactors = F)
for(method in c('freedict','wordnet','empath_new','fasttext','glove')) ##loop on the methods
{
  res<-readRDS(paste0('results/2015en_',method,'.Rda')) ##read the results
  sel1<-res[res$perc==perc,] ##select only the results relative to the chosen percentage of seed words
  sel<-sel1[sel1$length!=0,] ##exclude the word lists with no seed words for said percentage
  baseline<-readRDS(paste0('results/baseline_2015en_',method,'.Rda')) ##read the results of the baseline
  sel_bl<-baseline[baseline$perc==perc,] ##results for the threshold of seed words are selected
  sel_bl<-sel_bl[sel_bl$mean_F1>0,]
  ##computing the means of precision, recall and F1
  bl_prec<-round(mean(sel_bl$mean_prec),digits=2)
  bl_rec<-round(mean(sel_bl$mean_rec),digits=2)
  bl_F1<-round(mean(sel_bl$mean_F1),digits=2)

  df<-data.frame(method=method, perc_seed=perc, mean_prec=round(mean(sel$mean_prec),digits=2), bl_prec=bl_prec,
  report_df<-rbind(report_df,df)
}
report_df$method[report_df$method=='freedict']<-'LEXpander'
report_df$method[report_df$method=='empath_new']<-'Empath 2.0'
report_df$method[report_df$method=='fasttext']<-'FastText'
report_df$method[report_df$method=='glove']<-'GloVe'
report_df$method[report_df$method=='wordnet']<-'WordNet'
print(report_df)

```

##	method	perc_seed	mean_prec	bl_prec	mean_rec	bl_rec	mean_F1	bl_F1
## 1	LEXpander	0.3	0.16	0.01	0.14	0.02	0.13	0.01
## 2	WordNet	0.3	0.10	0.00	0.07	0.00	0.07	0.00
## 3	Empath 2.0	0.3	0.08	0.01	0.22	0.03	0.10	0.01
## 4	FastText	0.3	0.06	0.01	0.29	0.06	0.09	0.02
## 5	GloVe	0.3	0.07	0.01	0.13	0.03	0.08	0.02
##	mean_size							
## 1	614							
## 2	525							
## 3	1293							
## 4	2252							
## 5	773							

Table 1 supplementary materials: length of word lists relative to expansion of a random sample of LIWC 2015 in English.

```

perc<-"0.3" ##percentage of seed words
report_df<-data.frame(stringsAsFactors = F)

for(method in c('freedict','wordnet','empath_new','fasttext','glove')) ##loop on the methods
{
  res<-readRDS(paste0('results/2015en_',method,'.Rda')) ##select the results
  sel<-res[res$perc==perc,] ##select only the results relative to the chosen percentage of seed words
  sel<-sel[sel$length!=0,] ##exclude the word lists with no seed words
  labels<-c('Negemo','Posemo','Anx','Anger','Sad')
  selection<-c(31:35) ##labels relative to emotional word lists
  j<-0
  for (i in selection)
  {
    j<-j+1
  }
}

```

```

    only_one<-sel[sel$cat_id==i,] ##only the category of the loop
    df<-data.frame(method=method, perc_seed=perc,cat=labels[j],mean_length=round(only_one$length),stringsAsFactors=F)
    report_df<-rbind(report_df,df)
  }
  df<-data.frame(method=method, perc_seed=perc,cat='All',mean_length=round(mean(sel$length)),stringsAsFactors=F)
  report_df<-rbind(report_df,df)
}
report_df$method[report_df$method=='freedict']<-'LEXpander'
report_df$method[report_df$method=='empath_new']<-'Empath 2.0'
report_df$method[report_df$method=='fasttext']<-'FastText'
report_df$method[report_df$method=='glove']<-'GloVe'
report_df$method[report_df$method=='wordnet']<-'WordNet'
print(report_df)

```

##	method	perc_seed	cat	mean_length
## 1	LEXpander	0.3	Negemo	1626
## 2	LEXpander	0.3	Posemo	1966
## 3	LEXpander	0.3	Anx	428
## 4	LEXpander	0.3	Anger	656
## 5	LEXpander	0.3	Sad	446
## 6	LEXpander	0.3	All	614
## 7	WordNet	0.3	Negemo	1222
## 8	WordNet	0.3	Posemo	1839
## 9	WordNet	0.3	Anx	331
## 10	WordNet	0.3	Anger	668
## 11	WordNet	0.3	Sad	327
## 12	WordNet	0.3	All	525
## 13	Empath 2.0	0.3	Negemo	3227
## 14	Empath 2.0	0.3	Posemo	4019
## 15	Empath 2.0	0.3	Anx	3170
## 16	Empath 2.0	0.3	Anger	3020
## 17	Empath 2.0	0.3	Sad	2862
## 18	Empath 2.0	0.3	All	1293
## 19	FastText	0.3	Negemo	5916
## 20	FastText	0.3	Posemo	6977
## 21	FastText	0.3	Anx	3681
## 22	FastText	0.3	Anger	4201
## 23	FastText	0.3	Sad	3333
## 24	FastText	0.3	All	2252
## 25	GloVe	0.3	Negemo	1873
## 26	GloVe	0.3	Posemo	1613
## 27	GloVe	0.3	Anx	311
## 28	GloVe	0.3	Anger	516
## 29	GloVe	0.3	Sad	440
## 30	GloVe	0.3	All	773

Figure 3: dependence of F1 on the percentage of seed words (LIWC 2015 English)

```

mean_all<-data.frame(stringsAsFactors = F)
mean_bl<-data.frame(stringsAsFactors = F)
for (method in c('freedict','wordnet','empath_new','fasttext','glove')) ##loop on the methods
{
  res<-readRDS(paste0('results/2015en_',method,'.Rda')) ##load the data
  sel<-res[res$length!=0,] ##only results relative to categories with seed words
  means<-data.frame(mean_F1=apply(res$mean_F1,res$perc,mean), sd_F1=apply(res$sd_F1,res$perc,mean),me

```

```

mean_all<-rbind(mean_all,means)

bl<-readRDS(paste0('results/baseline_2015en_',method,'.Rda')) ##load the results of the baseline me
bl<-bl[bl$mean_F1>0,] ##select only the categories to which we added at least one word
means<-data.frame(mean_F1=apply(bl$mean_F1,bl$perc,mean), sd_F1=apply(bl$sd_F1,bl$perc,mean),metho
mean_bl<-rbind(mean_bl,means)

}

##change names to the methods
mean_all$method[mean_all$method=='empath_new']<-'Empath 2.0'
mean_all$method[mean_all$method=='freedict']<-'LEXpander'
mean_all$method[mean_all$method=='fasttext']<-'FastText'
mean_all$method[mean_all$method=='wordnet']<-'WordNet'
mean_all$method[mean_all$method=='glove']<-'GloVe'

bl<-data.frame(mean_F1=rep(0,9), sd_F1=rep(0,9), minF1=apply(mean_bl$mean_F1,mean_bl$th,min), maxF1=ta

##plot F1 vs percentage
ggplot(data = mean_all, aes(x = th)) +
  geom_line(aes(y=mean_F1,color=method))+
  geom_point(aes(y=mean_F1,shape=method))+
  geom_ribbon(data=bl,aes(ymin=minF1,ymax=maxF1),fill='grey70', alpha = 0.5)+
  labs(x='percentage random seed words',y='mean F1')+
  scale_x_continuous(breaks = c(10,20,30,40,50,60,70,80,90))+
  theme_bw()

```

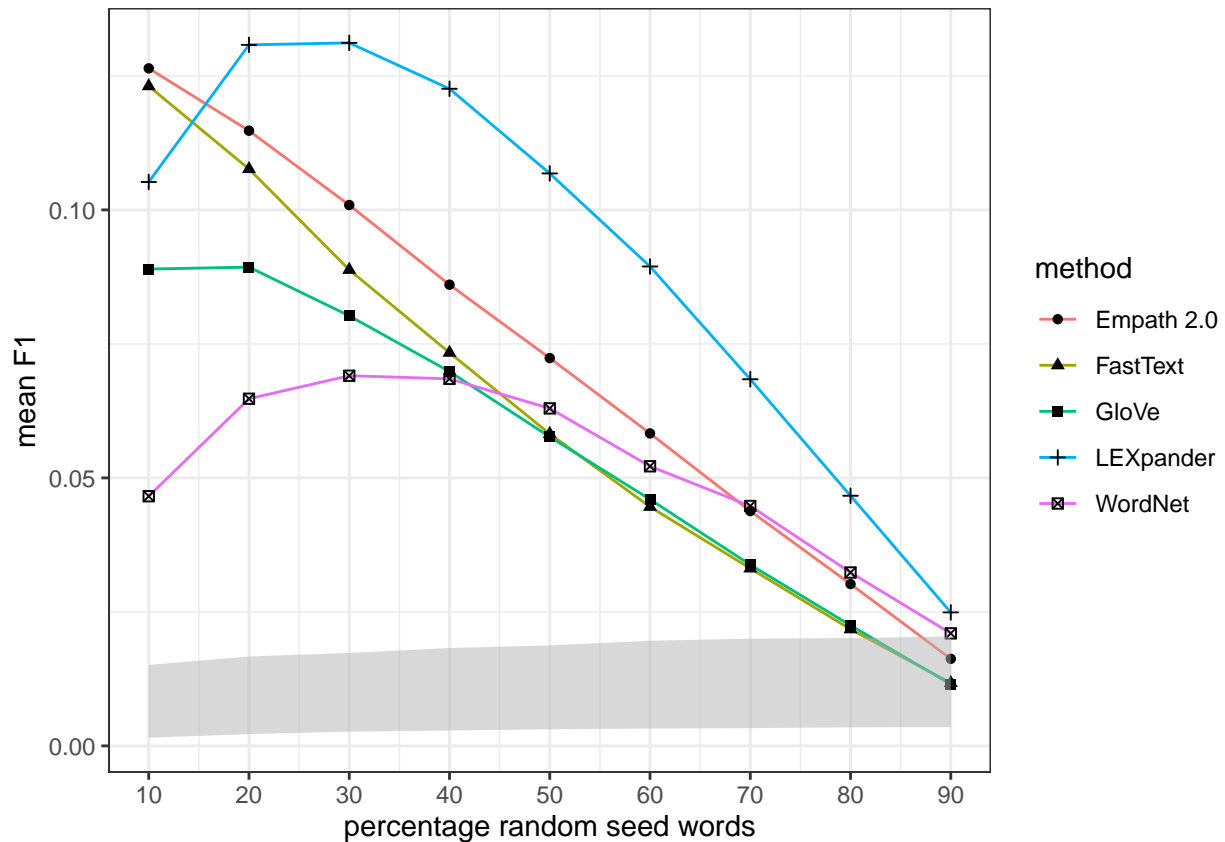


Table 2 supplementary materials: Precision, recall and F1 of the expansion of the EVs

```
report_df<-data.frame(stringsAsFactors = F)

for(method in c('EV','freedict','fasttext','glove','wordnet','empath_new')) ##loop on the methods
{
  if(method=='EV')
  {res<-readRDS('results/2015en_EV.Rda')} ##loading results relative to the original EV dataset
  else
  {res<-readRDS(paste0('results/EV_2015en_',method,'.Rda'))} ##loading the results
  res<-res[res$cat_id %in% seq(1,5),] ##selecting the word lists relative to posemo, negemo, anxifear, s

  if(!method%in%c('EV'))
  {
    baseline<-readRDS(paste0('results/baseline_EV_',method,'.Rda')) ##loading baseline results
    baseline<-baseline[baseline$cat_id%in% seq(1,5),] ##selecting only the emotional categories
    #means of precision, recall and F1
    bl_prec<-round(mean(baseline$prec),digits=2)
    bl_rec<-round(mean(baseline$rec),digits=2)
    bl_F1<-round(mean(baseline$F1),digits=2)
  }
  else ##we don't have a baseline method for the EVs
  {
    bl_prec<-NA
    bl_rec<-NA
    bl_F1<-NA
  }
  df<-data.frame(method=method,selection='EV',mean_prec=round(mean(res$prec),digits=2), bl_prec=bl_prec,
  report_df<-rbind(report_df,df)
}

report_df$method[report_df$method=='freedict']<-'LEXpander'
report_df$method[report_df$method=='empath_new']<-'Empath 2.0'
report_df$method[report_df$method=='fasttext']<-'FastText'
report_df$method[report_df$method=='glove']<-'GloVe'
report_df$method[report_df$method=='wordnet']<-'WordNet'
print(report_df)
```

##	method	selection	mean_prec	bl_prec	mean_rec	bl_rec	mean_F1	bl_F1
## 1	EV	EV	0.86	NA	0.19	NA	0.30	NA
## 2	LEXpander	EV	0.16	0.02	0.10	0.01	0.12	0.02
## 3	FastText	EV	0.06	0.02	0.34	0.10	0.10	0.03
## 4	GloVe	EV	0.07	0.01	0.03	0.01	0.04	0.01
## 5	WordNet	EV	0.11	0.00	0.06	0.00	0.08	0.00
## 6	Empath 2.0	EV	0.07	0.02	0.29	0.07	0.11	0.03
##	mean_size							
## 1	132							
## 2	570							
## 3	3684							
## 4	419							
## 5	492							
## 6	2702							

Table 3: Comparison between the expansion of the EVs and the expansion of random words from LIWC 2015

```

report_df<-data.frame(stringsAsFactors = F) ##results of the expansion of the EVs
report_df_random<-data.frame(stringsAsFactors = F) ##results of the expansion of a random subset of LIW
for(method in c('freedict','wordnet','empath_new','fasttext','glove')) ##loop on the methods
{
  res<-readRDS(paste0('results/EV_2015en_',method,'.Rda')) ##load the results for the expansion of the
  res<-res[res$cat_id %in% seq(1,5),] ##selecting the word lists relative to posemo, negemo, anxifear, s
  df<-data.frame(method=method,selection='EV',mean_prec=round(mean(res$prec),digits = 2),mean_rec=roun
  report_df<-rbind(report_df,df)

  res_random<-readRDS(paste0('results/2015en_comparisonEV_',method,'.Rda')) ##load the results for the
  df<-data.frame(method=method,selection='random',mean_prec=round(mean(res_random$mean_prec),digits = 2),
  report_df_random<-rbind(report_df_random,df)
}
report_df$method[report_df$method=='freedict']<-'LEXpander'
report_df$method[report_df$method=='empath_new']<-'Empath 2.0'
report_df$method[report_df$method=='fasttext']<-'FastText'
report_df$method[report_df$method=='glove']<-'GloVe'
report_df$method[report_df$method=='wordnet']<-'WordNet'

report_df_random$method[report_df_random$method=='freedict']<-'LEXpander'
report_df_random$method[report_df_random$method=='empath_new']<-'Empath 2.0'
report_df_random$method[report_df_random$method=='fasttext']<-'FastText'
report_df_random$method[report_df_random$method=='glove']<-'GloVe'
report_df_random$method[report_df_random$method=='wordnet']<-'WordNet'

print(report_df)

```

```

##      method selection mean_prec mean_rec mean_F1
## 1  LEXpander      EV      0.16      0.10      0.12
## 2   WordNet      EV      0.11      0.06      0.08
## 3 Empath 2.0      EV      0.07      0.29      0.11
## 4   FastText      EV      0.06      0.34      0.10
## 5     GloVe      EV      0.07      0.03      0.04

```

```
print(report_df_random)
```

```

##      method selection mean_prec mean_sdprec mean_rec mean_sdrec mean_F1
## 1  LEXpander  random      0.16          0.02      0.15          0.01      0.15
## 2   WordNet  random      0.12          0.02      0.08          0.01      0.09
## 3 Empath 2.0  random      0.07          0.00      0.34          0.01      0.12
## 4   FastText  random      0.07          0.00      0.40          0.01      0.11
## 5     GloVe  random      0.06          0.01      0.04          0.01      0.04
## mean_sdF1
## 1      0.01
## 2      0.01
## 3      0.00
## 4      0.01
## 5      0.01

```

Table 4: precision study of the expansion of the EVs

```

report_df<-data.frame(stringsAsFactors = F)
table<-data.frame(stringsAsFactors = F)

for(method in c('freedict','wordnet','fasttext','glove','empath_new'))

```

```
{
  res<-readRDS(paste0('results/EV_2015en_',method,'.Rda')) ##loading the results
  res<-res[res$cat_id %in% seq(1,2),] ##selecting only positive and negative categories
  df<-data.frame(method=method,mode='lower_bound',cat='Negative',prec=round(res$prec[res$cat_id==1],dig
  report_df<-rbind(report_df,df)
  df<-data.frame(method=method,mode='lower_bound',cat='Positive',prec=round(res$prec[res$cat_id==2],dig
  report_df<-rbind(report_df,df)
  df<-data.frame(method=method,mode='adjusted',cat='Negative',prec=round(res$prec_adj[res$cat_id==1], d
  report_df<-rbind(report_df,df)
  df<-data.frame(method=method,mode='adjusted',cat='Positive',prec=round(res$prec_adj[res$cat_id==2],di
  report_df<-rbind(report_df,df)
  df1<-data.frame(method=method,cat='Positive',prec=res$prec[res$cat_id==2],adj_prec=res$prec_adj[res$ca
  table<-rbind(table,df1)
  df1<-data.frame(method=method,cat='Negative',prec=res$prec[res$cat_id==1],adj_prec=res$prec_adj[res$ca
  table<-rbind(table,df1)
}
report_df$method[report_df$method=='freedict']<-'LEXpander'
report_df$method[report_df$method=='empath_new']<-'Empath 2.0'
report_df$method[report_df$method=='fasttext']<-'FastText'
report_df$method[report_df$method=='glove']<-'GloVe'
report_df$method[report_df$method=='wordnet']<-'WordNet'
print(report_df)
```

```
##      method      mode      cat prec  ci1  ci2
## 1  LEXpander lower_bound Negative 0.21  NA  NA
## 2  LEXpander lower_bound Positive 0.20  NA  NA
## 3  LEXpander   adjusted Negative 0.64 0.61 0.67
## 4  LEXpander   adjusted Positive 0.43 0.40 0.47
## 5   WordNet lower_bound Negative 0.15  NA  NA
## 6   WordNet lower_bound Positive 0.11  NA  NA
## 7   WordNet   adjusted Negative 0.63 0.60 0.67
## 8   WordNet   adjusted Positive 0.41 0.37 0.45
## 9   FastText lower_bound Negative 0.10  NA  NA
## 10  FastText lower_bound Positive 0.09  NA  NA
## 11  FastText   adjusted Negative 0.41 0.36 0.47
## 12  FastText   adjusted Positive 0.28 0.23 0.33
## 13   GloVe lower_bound Negative 0.11  NA  NA
## 14   GloVe lower_bound Positive 0.10  NA  NA
## 15   GloVe   adjusted Negative 0.25 0.21 0.30
## 16   GloVe   adjusted Positive 0.18 0.15 0.21
## 17 Empath 2.0 lower_bound Negative 0.13  NA  NA
## 18 Empath 2.0 lower_bound Positive 0.10  NA  NA
## 19 Empath 2.0   adjusted Negative 0.47 0.41 0.52
## 20 Empath 2.0   adjusted Positive 0.35 0.30 0.40
```

```
cor.test(table$prec,table$adj_prec) ##computation of the correlation between real and lower bound value
```

```
##
## Pearson's product-moment correlation
##
## data: table$prec and table$adj_prec
## t = 2.8122, df = 8, p-value = 0.02277
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
```



```
## 0.1356620 0.9243477
## sample estimates:
##      cor
## 0.7050643
```

Table 3 supplementary materials: length of the expanded word lists from EVs

```
report_df<-data.frame(stringsAsFactors = F)
for(method in c('EV','freedict','fasttext','wordnet','empath_new','glove')) ##loop on the methods
{
  if(method=='EV')
  {res<-readRDS('results/2015en_EV.Rda')} ##loading the comparison between EV and LIWC 2015
  else
  {res<-readRDS(paste0('results/EV_2015en_',method,'.Rda'))} ##loading the results
  sel<-res[res$length>0,] ##select only the categories to which we added at least one word

  labels<-c('Negemo','Posemo','AnxFear','Anger','Sad')
  selection<-c(1:5) ##labels relative to emotional word lists
  j<-0
  for (i in selection)
  {
    j<-j+1
    only_one<-sel[sel$cat_id==i,] ##selecting only one emotional category
    df<-data.frame(method=method,cat=labels[j],length=only_one$length,stringsAsFactors = F)
    report_df<-rbind(report_df,df)
  }
  df<-data.frame(method=method, cat='All',length=round(mean(sel$length[sel$cat_id%in%selection])),stringsAsFactors = F)
  report_df<-rbind(report_df,df)
}

report_df$method[report_df$method=='freedict']<-'LEXpander'
report_df$method[report_df$method=='empath_new']<-'Empath 2.0'
report_df$method[report_df$method=='fasttext']<-'FastText'
report_df$method[report_df$method=='glove']<-'GloVe'
report_df$method[report_df$method=='wordnet']<-'WordNet'
print(report_df)
```

```
##      method      cat length
## 1      EV  Negemo    276
## 2      EV  Posemo    172
## 3      EV AnxFear     62
## 4      EV   Anger     55
## 5      EV    Sad     95
## 6      EV    All    132
## 7 LEXpander Negemo   1068
## 8 LEXpander Posemo    815
## 9 LEXpander AnxFear    241
## 10 LEXpander  Anger    285
## 11 LEXpander   Sad    443
## 12 LEXpander   All    570
## 13 FastText Negemo   5288
## 14 FastText Posemo   3835
## 15 FastText AnxFear   3312
## 16 FastText  Anger   2960
## 17 FastText   Sad   3023
## 18 FastText   All   3684
```



```
## 19 WordNet Negemo 979
## 20 WordNet Posemo 685
## 21 WordNet AnxFear 194
## 22 WordNet Anger 279
## 23 WordNet Sad 325
## 24 WordNet All 492
## 25 Empath 2.0 Negemo 3325
## 26 Empath 2.0 Posemo 2723
## 27 Empath 2.0 AnxFear 2579
## 28 Empath 2.0 Anger 2417
## 29 Empath 2.0 Sad 2468
## 30 Empath 2.0 All 2702
## 31 GloVe Negemo 672
## 32 GloVe Posemo 878
## 33 GloVe AnxFear 186
## 34 GloVe Anger 128
## 35 GloVe Sad 229
## 36 GloVe All 419
```

Table 5: precision, recall and F1 of the expansion algorithms on the German LIWC

```
perc<-'0.3' ##selecting the percentage of seed words
mean_all<-data.frame(stringsAsFactors = F)
for(method in c('glove_deu', 'empath_new_deu', 'fasttext_deu', 'freedict_deu', 'odenet')) ##loop on the me
{
  res<-readRDS(paste0('results/2007deu_',method,'.Rda')) ##load the results
  res1<-res[res$perc==perc,] ##selecting the results relative to the chosen threshold
  res<-res1[res1$length!=0,] ##selecting the word lists to which we add at least one word
  baseline<-readRDS(paste0('results/baseline_2007deu_',method,'.Rda')) ##loading the baseline results
  sel_bl<-baseline[baseline$perc==perc,] ##results for the threshold of seed words are selected
  sel_bl<-sel_bl[sel_bl$mean_F1>0,] ##select only the categories to which we added at least one word

  df<-data.frame(method=method,mean_prec=round(mean(res$mean_prec),digits=2),bl_prec=round(mean(sel_bl$me
  mean_all<-rbind(mean_all,df)
}
mean_all$method[mean_all$method=='empath_new_deu']<-'Empath 2.0'
mean_all$method[mean_all$method=='freedict_deu']<-'LEXpander'
mean_all$method[mean_all$method=='fasttext_deu']<-'FastText'
mean_all$method[mean_all$method=='odenet']<-'OdeNet'
mean_all$method[mean_all$method=='glove_deu']<-'GloVe'
print(mean_all)
```

##	method	mean_prec	bl_prec	mean_rec	bl_rec	mean_F1	bl_F1	mean_size
## 1	GloVe	0.05	0.01	0.13	0.02	0.05	0.01	722
## 2	Empath 2.0	0.03	0.01	0.14	0.02	0.04	0.01	1905
## 3	FastText	0.03	0.01	0.16	0.03	0.04	0.01	2350
## 4	LEXpander	0.07	0.02	0.20	0.05	0.09	0.02	1714
## 5	OdeNet	0.03	0.00	0.00	0.00	0.00	0.00	170

Table 4 supplementary materials: length of word lists expanded from the LIWC 2007 German

```
perc<-"0.3" ##percentage of seed words
report_df<-data.frame(stringsAsFactors = F)
for(method in c('freedict_deu', 'odenet', 'fasttext_deu', 'glove_deu', 'empath_new_deu')) ##loop on the me
{
  res<-readRDS(paste0('results/2007deu_',method,'.Rda')) ##select the results
```

```

sel<-res[res$perc==perc,] ##results for a threshold of 30% seed words are selected
sel<-sel[sel$length!=0,] ##select only the categories to which we added at least one word
labels<-c('Negemo','Posemo','Anx','Anger','Sad')
selection<-c(16,13,17:19) #labels relative to emotional word lists
j<-0
for (i in selection)
{
  j<-j+1
  only_one<-sel[sel$cat_id==i,] ##select the results relative to only one emotional category
  if(nrow(only_one)>0)
  {df<-data.frame(method=method, perc_seed=perc,cat=labels[j],mean_length=round(only_one$length),stringsAsFactors=FALSE)
  report_df<-rbind(report_df,df)}
}
df<-data.frame(method=method, perc_seed=perc,cat='All',mean_length=round(mean(sel$length)),stringsAsFactors=FALSE)
report_df<-rbind(report_df,df)
}
report_df$method[report_df$method=='freedict_deu']<-'LEXpander'
report_df$method[report_df$method=='empath_new_deu']<-'Empath 2.0'
report_df$method[report_df$method=='fasttext_deu']<-'FastText'
report_df$method[report_df$method=='glove_deu']<-'GloVe'
report_df$method[report_df$method=='odenet']<-'OdeNet'
print(report_df)

```

##	method	perc_seed	cat	mean_length
## 1	LEXpander	0.3	Negemo	4501
## 2	LEXpander	0.3	Posemo	5385
## 3	LEXpander	0.3	Anx	1075
## 4	LEXpander	0.3	Anger	1559
## 5	LEXpander	0.3	Sad	1404
## 6	LEXpander	0.3	All	1714
## 7	OdeNet	0.3	Negemo	676
## 8	OdeNet	0.3	Posemo	503
## 9	OdeNet	0.3	Anx	94
## 10	OdeNet	0.3	Anger	169
## 11	OdeNet	0.3	Sad	147
## 12	OdeNet	0.3	All	170
## 13	FastText	0.3	Negemo	7726
## 14	FastText	0.3	Posemo	6793
## 15	FastText	0.3	Anx	2945
## 16	FastText	0.3	Anger	3572
## 17	FastText	0.3	Sad	3502
## 18	FastText	0.3	All	2350
## 19	GloVe	0.3	Negemo	1475
## 20	GloVe	0.3	Posemo	1791
## 21	GloVe	0.3	Anx	204
## 22	GloVe	0.3	Anger	361
## 23	GloVe	0.3	Sad	464
## 24	GloVe	0.3	All	722
## 25	Empath 2.0	0.3	Negemo	4900
## 26	Empath 2.0	0.3	Posemo	4656
## 27	Empath 2.0	0.3	Anx	3138
## 28	Empath 2.0	0.3	Anger	3123
## 29	Empath 2.0	0.3	Sad	3728
## 30	Empath 2.0	0.3	All	1905

Figure 4: dependence on the percentage of seed words (LIWC 2007 German)

```

mean_all<-data.frame(stringsAsFactors = F) #stores the results of the methods
mean_bl<-data.frame(stringsAsFactors = F) ##stores the baseline results
for (method in c('freedict_deu','glove_deu','empath_new_deu','fasttext_deu','odenet')) ##loop on the m
{
  res<-readRDS(paste0('results/2007deu_',method,'.Rda'))
  res<-res[res$length!=0,] ##select only the categories to which we added at least one word
  means<-data.frame(mean_F1=tapply(res$mean_F1,res$perc,mean), sd_F1=tapply(res$sd_F1,res$perc,mean), m
  mean_all<-rbind(mean_all,means)

  baseline<-readRDS(paste0('results/baseline_2007deu_',method,'.Rda')) ##load the results of the baseli
  bl<-baseline[baseline$mean_F1>0,] ##select only the categories to which we added at least one word
  means<-data.frame(mean_F1=tapply(bl$mean_F1,bl$perc,mean), sd_F1=tapply(bl$sd_F1,bl$perc,mean),meth
  mean_bl<-rbind(mean_bl,means)
}

bl<-data.frame(mean_F1=rep(0,9), sd_F1=rep(0,9), minF1=tapply(mean_bl$mean_F1,mean_bl$th,min), maxF1=ta

mean_all$method[mean_all$method=='empath_new_deu']<-'Empath 2.0'
mean_all$method[mean_all$method=='freedict_deu']<-'LEXpander'
mean_all$method[mean_all$method=='fasttext_deu']<-'FastText'
mean_all$method[mean_all$method=='odenet']<-'OdeNet'
mean_all$method[mean_all$method=='glove_deu']<-'GloVe'
##plot F1
ggplot(data = mean_all, aes(x = th)) +
  geom_line(aes(y=mean_F1,color=method))+
  geom_point(aes(y=mean_F1,shape=method))+
  geom_ribbon(data=bl,aes(ymin=minF1,ymax=maxF1),fill='grey70', alpha = 0.5)+
  labs(x='percentage random seed words',y='mean F1')+
  scale_x_continuous(breaks = c(10,20,30,40,50,60,70,80,90))+
  theme_bw()

```

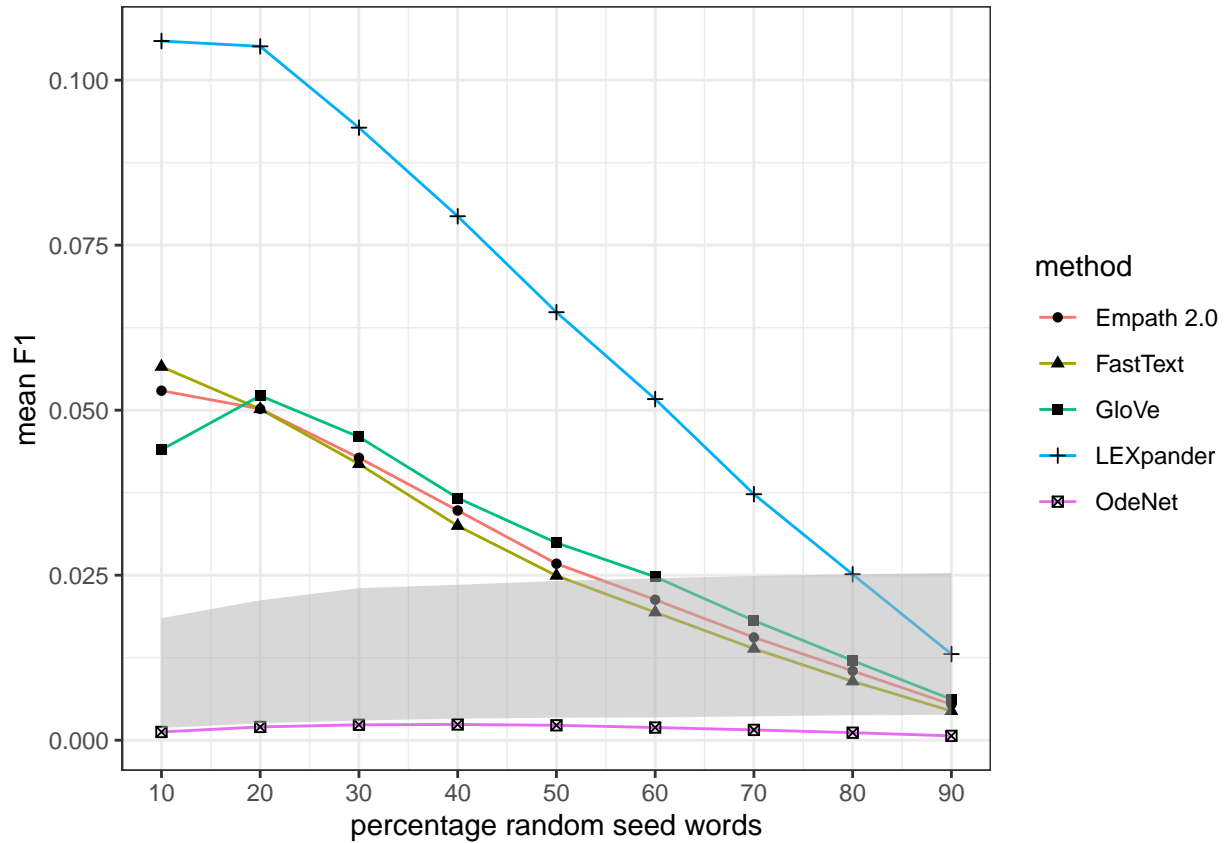


Table 5 supplementary materials: percentage of word lists computed

```
perc<-"0.3" ##percentage of seed words
report_df<-data.frame(stringsAsFactors = F)
for (method in c('freedict','glove','empath_new','fasttext','wordnet','odenet')) ##loop on the methods
{
  if(!method%in%c('wordnet'))
  {
    if (method=='odenet')
    {
      res<-readRDS(paste0('results/2007deu_',method,'.Rda')) ##loading the results with the German lexi
      res<-res[res$length!=0,]##select only the categories for which we had at least one seed word
      res<-res[res$mean_prec>0|res$mean_rec>0|res$mean_F1>0,] ##select the word lists to which we added
      res<-res[res$perc==perc,] ##select only the results relative to the chosen percentage of seed wor
      coverage_de<-round((length(unique(res$cat_id))*100)/68) ##percentage of word lists computed
      coverage_en<-NA
      coverage_ev<-NA
    }
    else
    {
      res<-readRDS(paste0('results/2007deu_',method,'_deu.Rda')) ##loading the results with the German
      res<-res[res$length!=0,]##select only the categories for which we had at least one seed word
      res<-res[res$mean_prec>0|res$mean_rec>0|res$mean_F1>0,] ##select the word lists to which we added
      res<-res[res$perc==perc,] ##select only the results relative to the chosen percentage of seed wor
      coverage_de<-round((length(unique(res$cat_id))*100)/68) ##percentage of word lists computed

      res<-readRDS(paste0('results/2015en_',method,'.Rda')) #####loading the results with the English le
```

```

sel<-res[res$perc==perc,] ##select only the results relative to the chosen percentage of seed wor
sel<-sel[sel$length!=0,] ##select only the categories for which we had at least one seed word
res<-res[res$mean_prec>0|res$mean_rec>0|res$mean_F1>0,] ##select the word lists to which we added
coverage_en<-round((length(unique(sel$cat_id))*100)/72) ##percentage of word lists computed

res<-readRDS(paste0('results/EV_2015en_',method,'.Rda')) ####loading the results with the EVs
res<-res[res$length!=0,] ##select only the categories for which we had at least one seed word
res<-res[res$prec>0|res$rec>0|res$F1>0,] ##select the word lists to which we added at least one w
coverage_ev<-round((length(unique(res$cat_id))*100)/6) ##percentage of word lists computed
}
}
else if (method=='wordnet')
{
coverage_de<-NA
res<-readRDS(paste0('results/2015en_',method,'.Rda')) ##read the results
sel<-res[res$perc==perc,] ##select only the results relative to the chosen percentage of seed words
sel<-sel[sel$length!=0,] ##select only the categories for which we had at least one seed word
sel<-sel[sel$mean_prec>0|sel$mean_rec>0|sel$mean_F1>0,] ##select the word lists to which we added a
coverage_en<-round((length(unique(sel$cat_id))*100)/72) ##percentage of word lists computed

res<-readRDS(paste0('results/EV_2015en_',method,'.Rda')) ####loading the results with the EVs
sel<-res[res$length!=0,] ##select only the categories for which we had at least one seed word
sel<-sel[sel$prec>0|sel$rec>0|sel$F1>0,] ##select the word lists to which we added at least one wor
coverage_ev<-round((length(unique(sel$cat_id))*100)/6) ##percentage of word lists computed
}
report_df<-rbind(report_df,data.frame(method=method,coverage_ev=coverage_ev,coverage_en=coverage_en,cov
}
report_df$method[report_df$method=='freedict']<-'LEXpander'
report_df$method[report_df$method=='empath_new']<-'Empath 2.0'
report_df$method[report_df$method=='fasttext']<-'FastText'
report_df$method[report_df$method=='glove']<-'GloVe'
report_df$method[report_df$method=='odenet']<-'OdeNet'
report_df$method[report_df$method=='wordnet']<-'WordNet'
print(report_df)

```

```

##      method coverage_ev coverage_en coverage_de
## 1  LEXpander         100         100          97
## 2    GloVe          100         100          88
## 3 Empath 2.0         100         100          94
## 4   FastText         100         100          94
## 5   WordNet          100          92          NA
## 6    OdeNet          NA          NA          40

```

Text analysis: counts of word occurrences in texts

```

# tab<-cor_on_texts('EV','wordnet')
# saveRDS(tab,paste0('results/all_counts_EV_', 'wordnet', '.Rda'))

```

Figure 5: text analysis with annotated word lists

```

corr_table<-readRDS('results/corr_table.Rda')
corr_alltogether_pos<-data.frame(stringsAsFactors = F)
corr_alltogether_neg<-data.frame(stringsAsFactors = F)
for(method in unique(corr_table$method.y))
{

```

```

for (dataset in unique(corr_table$dataset))
{
  sel<-corr_table[which((corr_table$method.y==method)&(corr_table$dataset==dataset)&(corr_table$cat==method.y))
  # ncat<-length(unique(sel$cat))
  corr_alltogether_pos<-rbind(corr_alltogether_pos,data.frame(method=method,dataset=dataset,corr=sel$corr))

  sel<-corr_table[which((corr_table$method.y==method)&(corr_table$dataset==dataset)&(corr_table$cat==method.y))
  # ncat<-length(unique(sel$cat))
  corr_alltogether_neg<-rbind(corr_alltogether_neg,data.frame(method=method,dataset=dataset,corr=sel$corr))
}
}

# corr_table1<-mean_corr_alltogether[!mean_corr_alltogether$dataset%in%c('reddit_home','reddit_family',
corr_table1<-corr_alltogether_neg
corr_table1<-corr_table1[!corr_table1$dataset %in% c('hourly_tweets_random1000','hourly_tweets'),]
corr_table1<-corr_table1[!corr_table1$dataset %in% c('coha_selected','reddit_home','reddit_family','reddit_tweets'),]
corr_table1$order<-NA
corr_table1$order[corr_table1$method=='EV']<-1
corr_table1$order[corr_table1$method=='empath_new']<-4
corr_table1$order[corr_table1$method=='freedict']<-2
corr_table1$order[corr_table1$method=='fasttext']<-5
corr_table1$order[corr_table1$method=='wordnet']<-3
corr_table1$order[corr_table1$method=='glove']<-6

corr_table1$method[corr_table1$method=='empath_new']<-'Empath 2.0'
corr_table1$method[corr_table1$method=='freedict']<-'LEXpander'
corr_table1$method[corr_table1$method=='fasttext']<-'FastText'
corr_table1$method[corr_table1$method=='wordnet']<-'WordNet'
corr_table1$method[corr_table1$method=='glove']<-'GloVe'

ggplot(corr_table1, aes(x=dataset, y=corr, fill=reorder(method,order))) +
  geom_errorbar(aes(ymin=ci1, ymax=ci2),
    size=.3, # Thinner lines
    width=.2,
    position=position_dodge(.9)) +
  geom_point(aes(colour=reorder(method,order),
    shape=reorder(method,order)),
    size=2,
    position=position_dodge(.9))+
  xlab("Dataset") +
  ylab("Mean correlation") +
  # labs(fill='method')+
  scale_x_discrete(labels=c('Brown corpus','COHA','Daily tweets','Reddit'))+
  theme(axis.text.x = element_text(angle = 45,hjust=1))+
  theme_bw()

```

