Supplementary materials to: Describing vocalizations in young children: A big data approach through citizen science annotations

Contents

History:

- 2020-11-03 first version
- 2021-03-08 increased reproducibility

Correspondence between lab & zooniverse annotation at the level of segments

Here we look at to what extent zooniverse and lab annotations match at the level of individual segments. Each data point is one segment (using LENA segmentation). Unlike in the main paper, here we will show results before applying the ordered rules that give prevalence to canonical, non-canonical, laughing, crying (in that order).

```
## PHASE 3 -- Generate views on the data: majority judgment on the segment

#dictionary that relates chunk to segment created by Chiara
dict <- fromJSON(file="dict_4.json")
dict.simple=unlist(dict)
length(dict.simple)
```

```
## [1] 33728
```

```
dict=data.frame(cbind(dict.simple,gsub(".$","",as.character(names(dict.simple)))))
colnames(dict)<-c("chunk","segmentId_DB")
length(levels(factor(dict$chunk)))
```

```
## [1] 33728
```

```
#33728 chunks
length(levels(factor(dict$segmentId_DB)))
```

```
## [1] 12170
```

```
#12170 segments
#NOTE!!! THERE ARE TWO CHUNKS MISSING FROM THIS DICTIONARY!!!
```

```
#laboratory annotations
read.csv("lab_annotations.csv")->lab_jud
dim(lab_jud)
```

```
## [1] 11982     8
```

```
#11982 segments

sum(lab_jud$segmentId_DB %in% levels(factor(dict$segmentId_DB))) #11980 in common
```

```
## [1] 11980
```

```
sum(!(lab_jud$segmentId_DB %in% levels(factor(dict$segmentId_DB)))) #2 fund in lab but not dict
```

```
## [1] 2
```

```
sum(!(levels(factor(dict$segmentId_DB)) %in% lab_jud$segmentId_DB )) #190 found in dict but not in lab
```

```
## [1] 190
```

```
#assuming that we should follow lab data, I'll kick out any segment not in there

#chunk info
read.csv("chunks_maj_judgments.csv",colClasses="factor")->maj_jud
maj_jud$AudioData=gsub(".mp3","",maj_jud$filename)


# next I need to match up the segments with the chunks
sum(dict.simple %in% maj_jud$AudioData) # 33727 found
```

```
## [1] 33727
```

```
sum(!(dict.simple %in% maj_jud$AudioData)) # 1 not found
```

```
## [1] 1
```

```
rownames(dict)<-dict$chunk
maj_jud$segmentId_DB <- dict[maj_jud$AudioData,"segmentId_DB"]
length(levels(factor(maj_jud$segmentId_DB))) #12170 segments
```

```
## [1] 12170
```

```
#in postprocess, the following code is executed, but not here, because we want to keep info about chunk
  # #generate majority at the segment level using our rule:
  # # canonical > non-canonical > crying > laughing > junk
  # table(maj_jud$segmentId_DB,maj_jud$Answer)->mytab
  # mytype<-ifelse(mytab[,"Canonical"]>0,"Canonical",
  #                ifelse(mytab[,"Non-Canonical"]>0,"Non-Canonical",
  #                       ifelse(mytab[,"Crying"]>0,"Crying",
  #                              ifelse(mytab[,"Laughing"]>0,"Laughing",
  #                                     ifelse(mytab[,"Junk"]>0,"Junk",""
  #                                     )))))

table(maj_jud$segmentId_DB,maj_jud$Answer)->mytab #we still create the table with the number of judgmen

# #but the mytype vector will include n's of each type
# mytype <- rep("",n=length(levels(factor(maj_jud$segmentId_DB))))
# mytype<-ifelse(mytab[,"Canonical"]>0,paste(mytype,"Ca",mytab[,"Canonical"],sep=""),mytype)
# mytype<-ifelse(mytab[,"Non-Canonical"]>0,paste(mytype,"N",mytab[,"Non-Canonical"],sep=""),mytype)
# mytype<-ifelse(mytab[,"Crying"]>0,paste(mytype,"Cr",mytab[,"Crying"],sep=""),mytype)
# mytype<-ifelse(mytab[,"Laughing"]>0,paste(mytype,"L",mytab[,"Laughing"],sep=""),mytype)
# mytype<-ifelse(mytab[,"Junk"]>0,paste(mytype,"J",mytab[,"Junk"],sep=""),mytype)
#
# levels(factor(mytype))
# #there are over 350 combinations!! So we won't do that

mytype <- rep("",n=length(levels(factor(maj_jud$segmentId_DB))))
mytype<-ifelse(mytab[,"Canonical"]>0,paste(mytype,"Ca",sep=""),mytype)
mytype<-ifelse(mytab[,"Non-Canonical"]>0,paste(mytype,"N",sep=""),mytype)
mytype<-ifelse(mytab[,"Crying"]>0,paste(mytype,"Cr",sep=""),mytype)
mytype<-ifelse(mytab[,"Laughing"]>0,paste(mytype,"L",sep=""),mytype)
mytype<-ifelse(mytab[,"Junk"]>0,paste(mytype,"J",sep=""),mytype)
levels(factor(mytype))
```

```
##  [1] ""       "Ca"     "CaCr"   "CaCrJ"  "CaCrL"  "CaCrLJ" "CaJ"    "CaL"
##  [9] "CaLJ"   "CaN"    "CaNCr"  "CaNCrJ" "CaNCrL" "CaNJ"   "CaNL"   "CaNLJ"
## [17] "Cr"     "CrJ"    "CrL"    "CrLJ"   "J"      "L"      "LJ"     "N"
## [25] "NCr"    "NCrJ"   "NCrL"   "NCrLJ"  "NJ"     "NL"     "NLJ"
```

```
# about 30 unique combinations, let's go for it

zoo_jud=cbind(row.names(mytab),mytype)
colnames(zoo_jud)<-c("segmentId_DB","Answer")

merge(zoo_jud,lab_jud,by="segmentId_DB")->all_jud
dim(all_jud)
```

```
## [1] 11980     9
```

```
#11980 segments
levels(factor(all_jud$ChildID))
```

```
##  [1] "1111_1" "1151_1" "1801_1" "2881_1" "2931_1" "2991_1" "3021_1" "3041_1"
##  [9] "3131_1" "3201_1" "3211_1" "3211_2" "3291_1" "3401_1" "3451_1" "3491_7"
## [17] "3681_1" "3741_1" "3831_1" "5031_1"
```

```
length(levels(factor(all_jud$ChildID)))#all 20 kids here
```

```
## [1] 20
```

```
# create columns with names that match the following chunks
all_jud$Zoon_classif = all_jud$Answer
all_jud$lab = all_jud$Major_Choice

# for the lab case, simplify
all_jud$lab[all_jud$lab=="Canonical syllables"]<-"Ca"
all_jud$lab[all_jud$lab=="Words"]<-"Ca"
all_jud$lab[all_jud$lab=="Crying"]<-"Cr"
all_jud$lab[all_jud$lab=="Laughing"]<-"L"
all_jud$lab[all_jud$lab=="Don't mark"]<-"J"
all_jud$lab[all_jud$lab=="None"]<-""
all_jud$lab[all_jud$lab=="Non-canonical syllables"]<-"N"

table(all_jud$lab)
```

```
##
```

```
##        Ca   Cr    J    L    N
##  333 1860  598 2361  188 6640
```

```
table(all_jud$Zoon_classif)
```

```
## 
##            Ca   CaCr  CaCrJ  CaCrL CaCrLJ    CaJ    CaL   CaLJ    CaN  CaNCr 
##    393    465     30      3      1      1    266     13      2    662     54 
## CaNCrJ CaNCrL   CaNJ   CaNL  CaNLJ     Cr    CrJ    CrL   CrLJ      J      L 
##     10      3    167     23      6    732    194    115      3   1464    275 
##     LJ      N    NCr   NCrJ   NCrL  NCrLJ     NJ     NL    NLJ 
##    167   3899    863    129     71      7   1631    270     61
```

```
#remove classes with fewer than 10 instances
table(all_jud$Zoon_classif)[table(all_jud$Zoon_classif)<10]
```

```
## 
##  CaCrJ  CaCrL CaCrLJ   CaLJ CaNCrL  CaNLJ   CrLJ  NCrLJ 
##      3      1      1      2      3      6      3      7
```

```
all_jud=all_jud[!(all_jud$Zoon_classif %in% names(table(all_jud$Zoon_classif)[table(all_jud$Zoon_classi

#remove classes with no majority judgment
all_jud=all_jud[all_jud$Zoon_classif !="",]
all_jud=all_jud[all_jud$lab !="",]
dim(all_jud)
```

```
## [1] 11243    11
```

```
all_jud$Zoon_classif=factor(all_jud$Zoon_classif)
all_jud$lab=factor(all_jud$lab, levels=levels(all_jud$Zoon_classif))
```

```
mycf=confusionMatrix(all_jud$lab, all_jud$Zoon_classif, dnn = c("Lab","Zooniverse"))
conf_tab=mycf$table
# this package uses sensitivity & specificity
#Sensitivity=recall
#Specificity=precision
mycf
```

```
## Confusion Matrix and Statistics
## 
##          Zooniverse
## Lab        Ca CaCr  CaJ  CaL  CaN CaNCr CaNCrJ CaNJ CaNL   Cr  CrJ  CrL    J
##   Ca      336   21  106    2  447    26      9   89   12   26    5    3   79
##   CaCr      0    0    0    0    0     0      0    0    0    0    0    0    0
##   CaJ       0    0    0    0    0     0      0    0    0    0    0    0    0
##   CaL       0    0    0    0    0     0      0    0    0    0    0    0    0
##   CaN       0    0    0    0    0     0      0    0    0    0    0    0    0
##   CaNCr     0    0    0    0    0     0      0    0    0    0    0    0    0
##   CaNCrJ    0    0    0    0    0     0      0    0    0    0    0    0    0
##   CaNJ      0    0    0    0    0     0      0    0    0    0    0    0    0
##   CaNL      0    0    0    0    0     0      0    0    0    0    0    0    0
##   Cr        0    1    0    0    0     3      0    0    0  215   33   50    6
##   CrJ       0    0    0    0    0     0      0    0    0    0    0    0    0
##   CrL       0    0    0    0    0     0      0    0    0    0    0    0    0
##   J        81    1  108    4   72     7      0   47    4   33   36    6  989
##   L         1    0    0    0    0     0      0    0    2    5    3   10    6
##   LJ        0    0    0    0    0     0      0    0    0    0    0    0    0
##   N        38    7   42    5  126    13      1   26    5  427  109   35  361
##   NCr       0    0    0    0    0     0      0    0    0    0    0    0    0
##   NCrJ      0    0    0    0    0     0      0    0    0    0    0    0    0
##   NCrL      0    0    0    0    0     0      0    0    0    0    0    0    0
##   NJ        0    0    0    0    0     0      0    0    0    0    0    0    0
##   NL        0    0    0    0    0     0      0    0    0    0    0    0    0
##   NLJ       0    0    0    0    0     0      0    0    0    0    0    0    0
##         Zooniverse
## Lab         L   LJ    N  NCr NCrJ NCrL   NJ   NL  NLJ
##   Ca       22    7  396   38    3    3  136   26    3
##   CaCr      0    0    0    0    0    0    0    0    0
##   CaJ       0    0    0    0    0    0    0    0    0
##   CaL       0    0    0    0    0    0    0    0    0
##   CaN       0    0    0    0    0    0    0    0    0
##   CaNCr     0    0    0    0    0    0    0    0    0
##   CaNCrJ    0    0    0    0    0    0    0    0    0
##   CaNJ      0    0    0    0    0    0    0    0    0
##   CaNL      0    0    0    0    0    0    0    0    0
##   Cr        7    1   27  163   24   40    6    7    1
##   CrJ       0    0    0    0    0    0    0    0    0
##   CrL       0    0    0    0    0    0    0    0    0
##   J        46   54  287   27   11    1  394   34   27
##   L        81   36    7    1    0    2    4   25    3
##   LJ        0    0    0    0    0    0    0    0    0
##   N        97   58 3127  605   85   19 1049  149   25
##   NCr       0    0    0    0    0    0    0    0    0
##   NCrJ      0    0    0    0    0    0    0    0    0
##   NCrL      0    0    0    0    0    0    0    0    0
##   NJ        0    0    0    0    0    0    0    0    0
##   NL        0    0    0    0    0    0    0    0    0
##   NLJ       0    0    0    0    0    0    0    0    0
##
## Overall Statistics
##
##                Accuracy : 0.4223
##                  95% CI : (0.4132, 0.4315)
##     No Information Rate : 0.3419
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.2489
##
##  Mcnemar's Test P-Value : NA
##
## Statistics by Class:
##
##                      Class: Ca Class: CaCr Class: CaJ Class: CaL Class: CaN
## Sensitivity            0.73684    0.000000    0.00000  0.0000000    0.00000
## Specificity            0.86474    1.000000    1.00000  1.0000000    1.00000
## Pos Pred Value         0.18719         NaN        NaN        NaN        NaN
## Neg Pred Value         0.98730    0.997332    0.97723  0.9990216    0.94263
## Prevalence             0.04056    0.002668    0.02277  0.0009784    0.05737
## Detection Rate         0.02989    0.000000    0.00000  0.0000000    0.00000
## Detection Prevalence   0.15965    0.000000    0.00000  0.0000000    0.00000
## Balanced Accuracy      0.80079    0.500000    0.50000  0.5000000    0.50000
##                        Class: CaNCr Class: CaNCrJ Class: CaNJ Class: CaNL
## Sensitivity                0.000000     0.0000000     0.00000    0.000000
## Specificity                1.000000     1.0000000     1.00000    1.000000
## Pos Pred Value                  NaN           NaN         NaN         NaN
## Neg Pred Value             0.995642     0.9991106     0.98559    0.997954
## Prevalence                 0.004358     0.0008894     0.01441    0.002046
## Detection Rate             0.000000     0.0000000     0.00000    0.000000
## Detection Prevalence       0.000000     0.0000000     0.00000    0.000000
## Balanced Accuracy          0.500000     0.5000000     0.50000    0.500000
##                        Class: Cr Class: CrJ Class: CrL Class: J Class: L
## Sensitivity              0.30453    0.00000    0.00000  0.68633 0.320158
## Specificity              0.96498    1.00000    1.00000  0.86941 0.990446
## Pos Pred Value           0.36815        NaN        NaN  0.43587 0.435484
## Neg Pred Value           0.95394    0.98346    0.99075  0.94963 0.984444
## Prevalence               0.06279    0.01654    0.00925  0.12817 0.022503
## Detection Rate           0.01912    0.00000    0.00000  0.08797 0.007204
## Detection Prevalence     0.05194    0.00000    0.00000  0.20181 0.016544
## Balanced Accuracy        0.63476    0.50000    0.50000  0.77787 0.655302
##                        Class: LJ Class: N Class: NCr Class: NCrJ Class: NCrL
## Sensitivity              0.00000   0.8135    0.00000     0.00000    0.000000
## Specificity              1.00000   0.5564    1.00000     1.00000    1.000000
## Pos Pred Value               NaN   0.4879        NaN         NaN         NaN
## Neg Pred Value           0.98612   0.8517    0.92582     0.98906    0.994219
## Prevalence               0.01388   0.3419    0.07418     0.01094    0.005781
## Detection Rate           0.00000   0.2781    0.00000     0.00000    0.000000
## Detection Prevalence     0.00000   0.5700    0.00000     0.00000    0.000000
## Balanced Accuracy        0.50000   0.6850    0.50000     0.50000    0.500000
##                        Class: NJ Class: NL Class: NLJ
## Sensitivity               0.0000   0.00000   0.000000
## Specificity               1.0000   1.00000   1.000000
## Pos Pred Value               NaN       NaN        NaN
## Neg Pred Value            0.8587   0.97856   0.994752
## Prevalence                0.1413   0.02144   0.005248
## Detection Rate            0.0000   0.00000   0.000000
## Detection Prevalence      0.0000   0.00000   0.000000
## Balanced Accuracy         0.5000   0.50000   0.500000
```

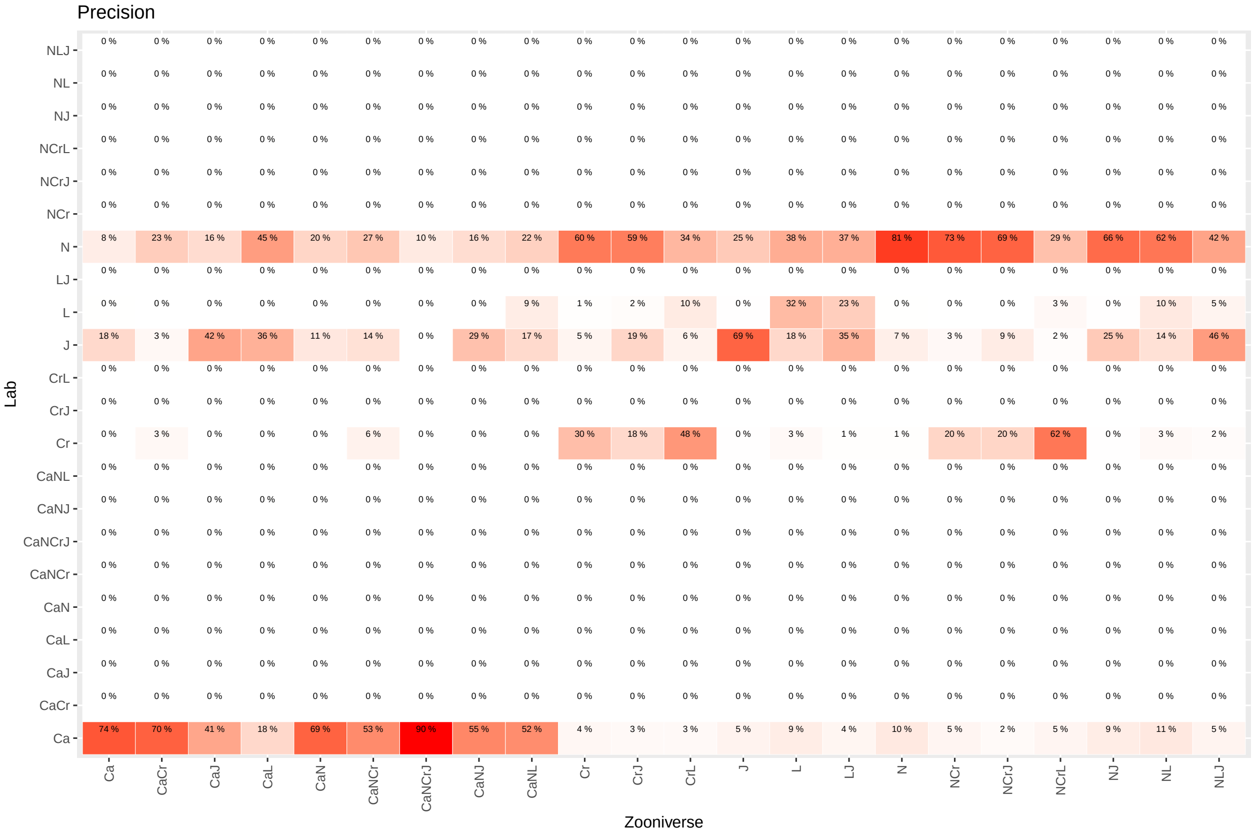
Precision

Precision means: If a segment was called X by zooniverse coders, what proportion of the time was it called X by lab coders?

```
colsums=colSums(conf_tab)
my_conf_tab=conf_tab
for(i in 1:dim(my_conf_tab)[2]) my_conf_tab[,i]=my_conf_tab[,i]/colsums[i]
colSums(my_conf_tab)
```

```
##     Ca   CaCr    CaJ    CaL    CaN  CaNCr CaNCrJ   CaNJ   CaNL     Cr    CrJ
##      1      1      1      1      1      1      1      1      1      1      1
##    CrL      J      L     LJ      N    NCr   NCrJ   NCrL     NJ     NL    NLJ
##      1      1      1      1      1      1      1      1      1      1      1
```

```
prop_cat=data.frame(my_conf_tab*100) #generates precision because columns
prop_cat$id=paste(prop_cat$Lab,prop_cat$Zooniverse)
colnames(prop_cat)[3]<-"pr"
data.frame(conf_tab)->stall
stall$id=paste(stall$Lab,stall$Zooniverse)
stall=merge(stall,prop_cat[c("id","pr")])
ggplot(data = stall, mapping = aes(y = Lab, x=Zooniverse)) +
 geom_tile(aes(fill= rescale(pr)), colour = "white") +
  geom_text(aes(label = paste(round(pr),"%")), vjust = -1,size=2) +
#  geom_text(aes(label = Freq), vjust = 1,size=1) +
  scale_fill_gradient(low = "white", high = "red", name = "Percentage") +
     theme(legend.position = "none") +
  xlab("Zooniverse") + ylab("Lab") +
  ggtitle("Precision")+ theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))
```



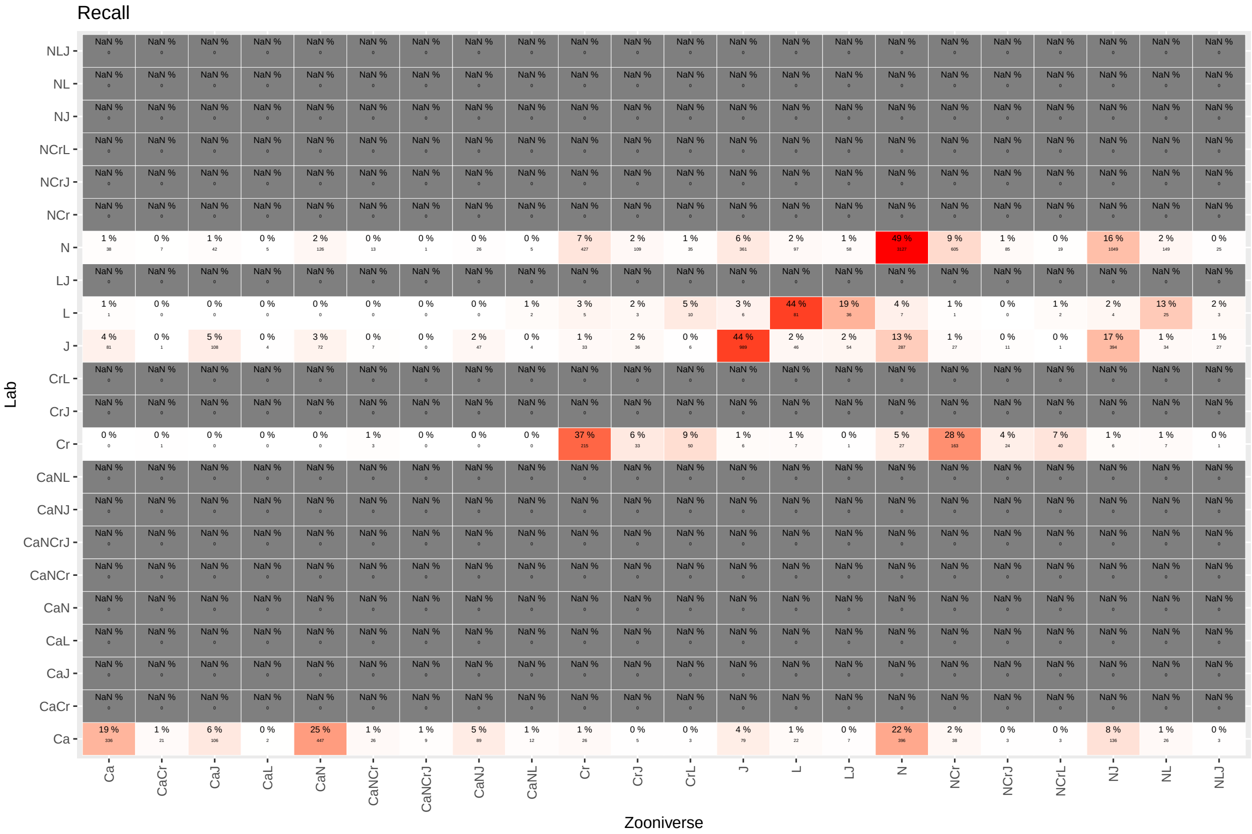
Recall

Recall means: If a segment was called X by lab coders, what proportion of the time was it called X by zooniverse coders?

```
rowsums=rowSums(conf_tab)
my_conf_tab=conf_tab
for(i in 1:dim(my_conf_tab)[1]) my_conf_tab[,i]=my_conf_tab[,i]/rowsums[i]
rowSums(my_conf_tab)
```

```
##      Ca   CaCr    CaJ    CaL    CaN  CaNCr CaNCrJ   CaNJ   CaNL     Cr    CrJ
##     Inf    NaN    NaN    NaN    NaN    NaN    NaN    NaN    NaN    NaN    NaN
##     CrL      J      L     LJ      N    NCr   NCrJ   NCrL     NJ     NL    NLJ
##     NaN    NaN    NaN    NaN    Inf    NaN    NaN    NaN    NaN    NaN    NaN
```

```
prop_cat=data.frame(conf_tab/rowSums(conf_tab)*100)  #generates recall because rows
prop_cat$id=paste(prop_cat$Lab,prop_cat$Zooniverse)
colnames(prop_cat)[3]<-"rec"
data.frame(conf_tab)->stall
stall$id=paste(stall$Lab,stall$Zooniverse)
stall=merge(stall,prop_cat[c("id","rec")])
ggplot(data = stall, mapping = aes(y = Lab, x=Zooniverse)) +
 geom_tile(aes(fill= rescale(rec)), colour = "white") +
  geom_text(aes(label = paste(round(rec),"%")), vjust = -1,size=2) +
  geom_text(aes(label = Freq), vjust = 1,size=1) +
  scale_fill_gradient(low = "white", high = "red", name = "Percentage") +
     theme(legend.position = "none") +
  xlab("Zooniverse") + ylab("Lab") +
  ggtitle("Recall")+ theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))
```



Read in & clean up final data

```
read.csv("key_info.csv")->x
rownames(x)<-x$X


read.csv("chunks_maj_judgments.csv")->chunks
read.csv("zoo_lab_maj_judgments.csv")->data_all

label_options=c("Canonical" , "Non-Canonical"  ,     "Crying"   ,    "Laughing",        "Junk" )

#use better names
data_all$Zoon_classif=data_all$Answer

# create lab column with easier to read correspondance
data_all$lab<-as.character(data_all$Major_Choice)
data_all$lab[data_all$lab=="Non-canonical syllables"]<-"Non-Canonical"
data_all$lab[data_all$lab=="Canonical syllables"]<-"Canonical"
data_all$lab[data_all$lab %in% c("Don't mark","None")]<-"Junk"
data_all$lab=factor(data_all$lab,levels=label_options)
#apply same factor levels as zooniverse so that we can do symmetrical confusion matrices

#add binomials for Linguistic Proportion
data_all$lab_ling=ifelse(data_all$lab %in% c("Canonical","Non-Canonical"),1,0)
data_all$zoo_ling=ifelse(data_all$Zoon_classif %in% c("Canonical","Non-Canonical"),1,0)
data_all$lab_ling[data_all$lab=="Junk"]<-NA
data_all$zoo_ling[data_all$lab=="Junk"]<-NA

#add binomials for Canonical Proportion
data_all$lab_can=data_all$zoo_can=NA
data_all$lab_can[data_all$lab=="Canonical"]<-1
data_all$lab_can[data_all$lab=="Non-Canonical"]<-0
data_all$zoo_can[data_all$Zoon_classif=="Canonical"]<-1
data_all$zoo_can[data_all$Zoon_classif=="Non-Canonical"]<-0


demo_data=read.csv("demo-data.tsv",sep ="\t")
#add filenames to demo data, to be used later
demo_data_fn <- demo_data %>%
     left_join(select(data_all, filename, ChildID), by = c("ChildID"))
demo_data_fn<-unique(demo_data_fn)
```

Separate confusion matrices for Angelman syndrome children

```
# CM with just AS kids
data_AS<-subset(data_all, Diagnosis=="AngelmanSyndrome")
mycf=confusionMatrix(data_AS$lab, data_AS$Zoon_classif, dnn = c("Lab","Zooniverse"))
conf_tab=mycf$table
mycf
```

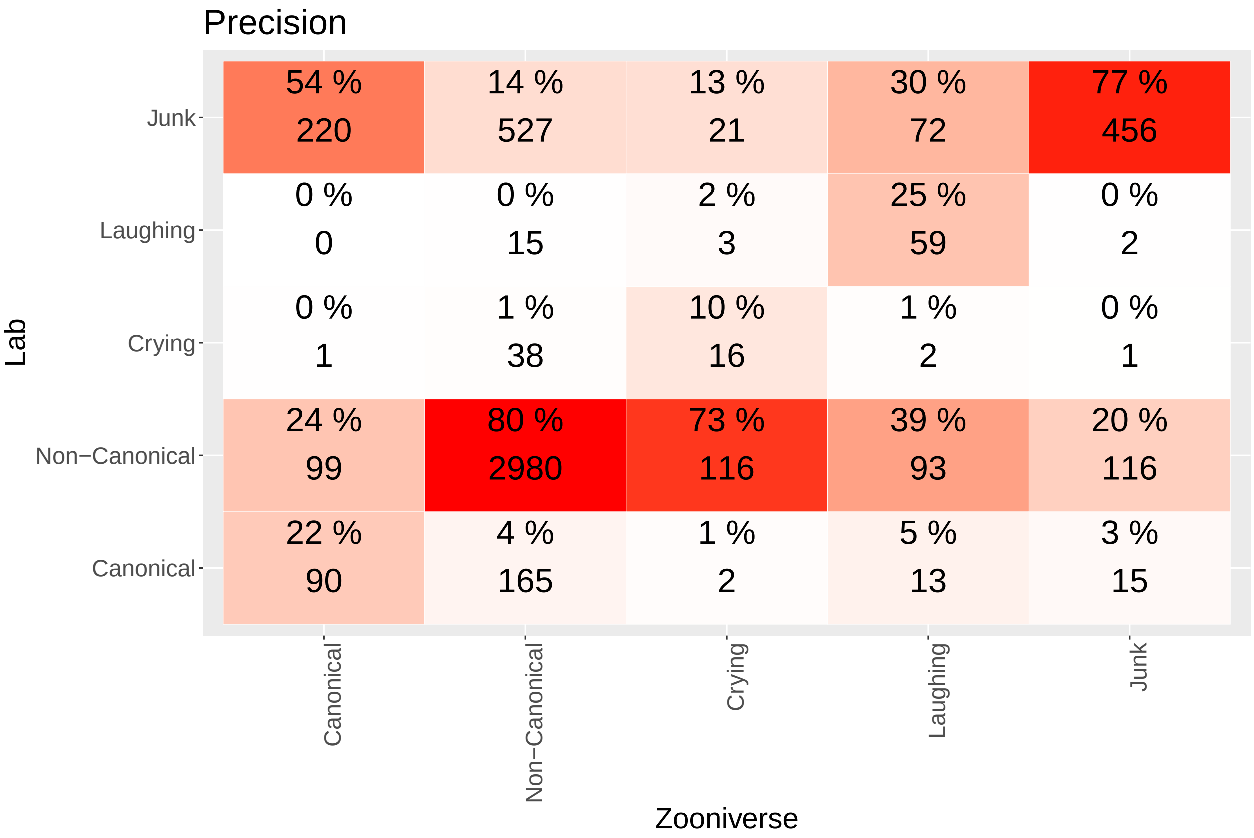
```
## Confusion Matrix and Statistics
##
##                Zooniverse
## Lab             Canonical Non-Canonical Crying Laughing Junk
##   Canonical            90           165      2       13   15
##   Non-Canonical        99          2980    116       93  116
##   Crying                1            38     16        2    1
##   Laughing              0            15      3       59    2
##   Junk                220           527     21       72  456
```

```
##
## Overall Statistics
##
##                Accuracy : 0.703
##                  95% CI : (0.6903, 0.7155)
##     No Information Rate : 0.7273
##     P-Value [Acc > NIR] : 0.9999
##
##                   Kappa : 0.3839
##
##  Mcnemar's Test P-Value : <2e-16
##
## Statistics by Class:
##
##                      Class: Canonical Class: Non-Canonical Class: Crying
## Sensitivity                   0.21951               0.8000      0.101266
## Specificity                   0.95862               0.6965      0.991539
## Pos Pred Value                0.31579               0.8754      0.275862
## Neg Pred Value                0.93384               0.5664      0.971959
## Prevalence                    0.08005               0.7273      0.030847
## Detection Rate                0.01757               0.5818      0.003124
## Detection Prevalence          0.05564               0.6646      0.011324
## Balanced Accuracy             0.58906               0.7482      0.546402
##                      Class: Laughing Class: Junk
## Sensitivity                  0.24686     0.77288
## Specificity                  0.99590     0.81465
## Pos Pred Value               0.74684     0.35185
## Neg Pred Value               0.96431     0.96498
## Prevalence                   0.04666     0.11519
## Detection Rate               0.01152     0.08903
## Detection Prevalence         0.01542     0.25303
## Balanced Accuracy            0.62138     0.79377
```

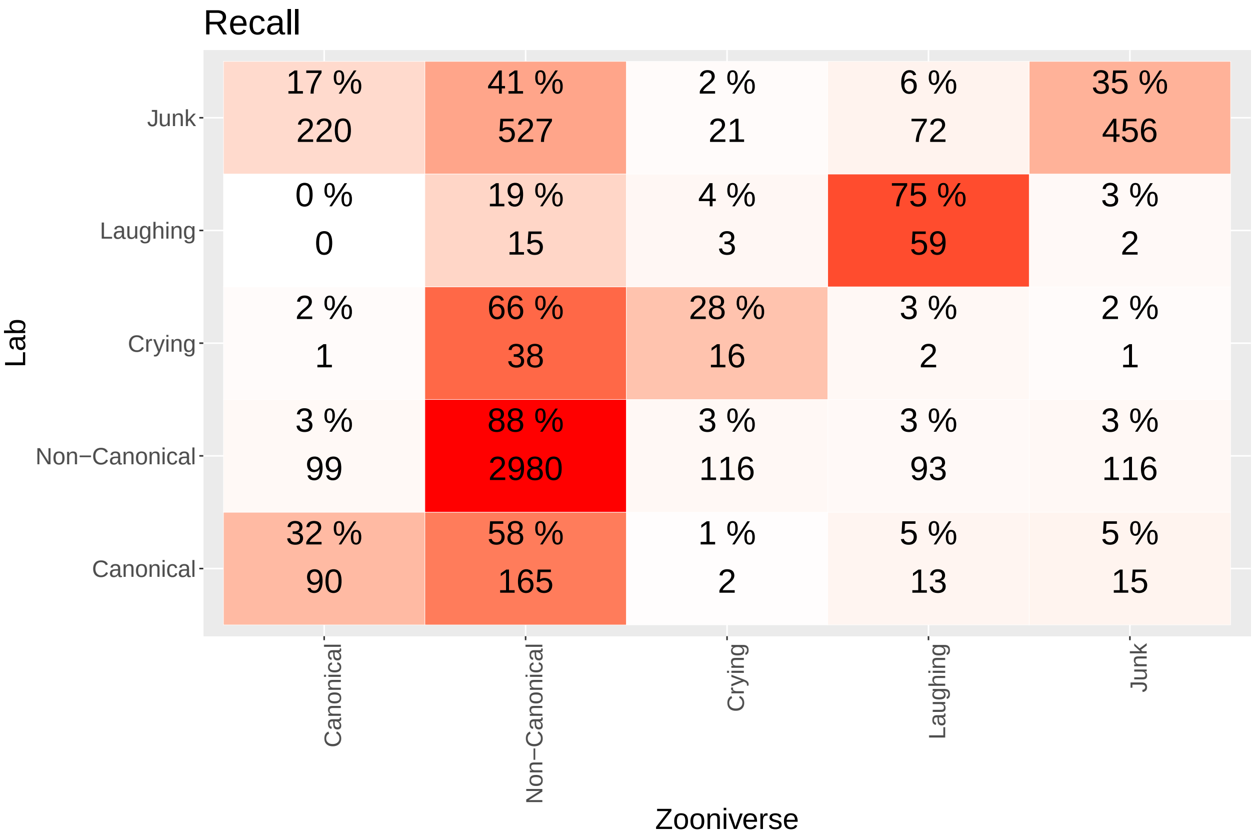
```r
colsums=colSums(conf_tab)
my_conf_tab=conf_tab
for(i in 1:5) my_conf_tab[,i]=my_conf_tab[,i]/colsums[i]
colSums(my_conf_tab)
```

```
##     Canonical Non-Canonical        Crying      Laughing          Junk
##             1             1             1             1             1
```

```r
prop_cat=data.frame(my_conf_tab*100) #generates precision because columns
prop_cat$id=paste(prop_cat$Lab,prop_cat$Zooniverse)
colnames(prop_cat)[3]<-"pr"
data.frame(conf_tab)->stall
stall$id=paste(stall$Lab,stall$Zooniverse)
stall=merge(stall,prop_cat[c("id","pr")])
ggplot(data = stall, mapping = aes(y = Lab, x=Zooniverse)) +
 geom_tile(aes(fill= rescale(pr)), colour = "white") +
  geom_text(aes(label = paste(round(pr),"%")), vjust = -1,size=8) +
  geom_text(aes(label = Freq), vjust = 1,size=8) +
  scale_fill_gradient(low = "white", high = "red", name = "Proportion") +
     theme(legend.position = "none") +
  xlab("Zooniverse") + ylab("Lab") +
  ggtitle("Precision")+theme(text = element_text(size=20),
        axis.text.x = element_text(angle=90, hjust=1))
```



```
prop_cat=data.frame(conf_tab/rowSums(conf_tab)*100)  #generates recall because rows
prop_cat$id=paste(prop_cat$Lab,prop_cat$Zooniverse)
colnames(prop_cat)[3]<-"rec"
data.frame(conf_tab)->stall
stall$id=paste(stall$Lab,stall$Zooniverse)
stall=merge(stall,prop_cat[c("id","rec")])
ggplot(data = stall, mapping = aes(y = Lab, x=Zooniverse)) +
 geom_tile(aes(fill= rescale(rec)), colour = "white") +
  geom_text(aes(label = paste(round(rec),"%")), vjust = -1,size=8) +
  geom_text(aes(label = Freq), vjust = 1,size=8) +
  scale_fill_gradient(low = "white", high = "red", name = "Proportion") +
     theme(legend.position = "none") +
  xlab("Zooniverse") + ylab("Lab") +
  ggtitle("Recall")+theme(text = element_text(size=20),
        axis.text.x = element_text(angle=90, hjust=1))
```


Recall

Separate confusion matrices with just the low risk controls

```r
# CM with just TD kids
data_TD<-subset(data_all, Diagnosis=="Low-RiskControl")
mycf=confusionMatrix(data_TD$lab, data_TD$Zoon_classif, dnn = c("Lab","Zooniverse"))
conf_tab=mycf$table
mycf
```

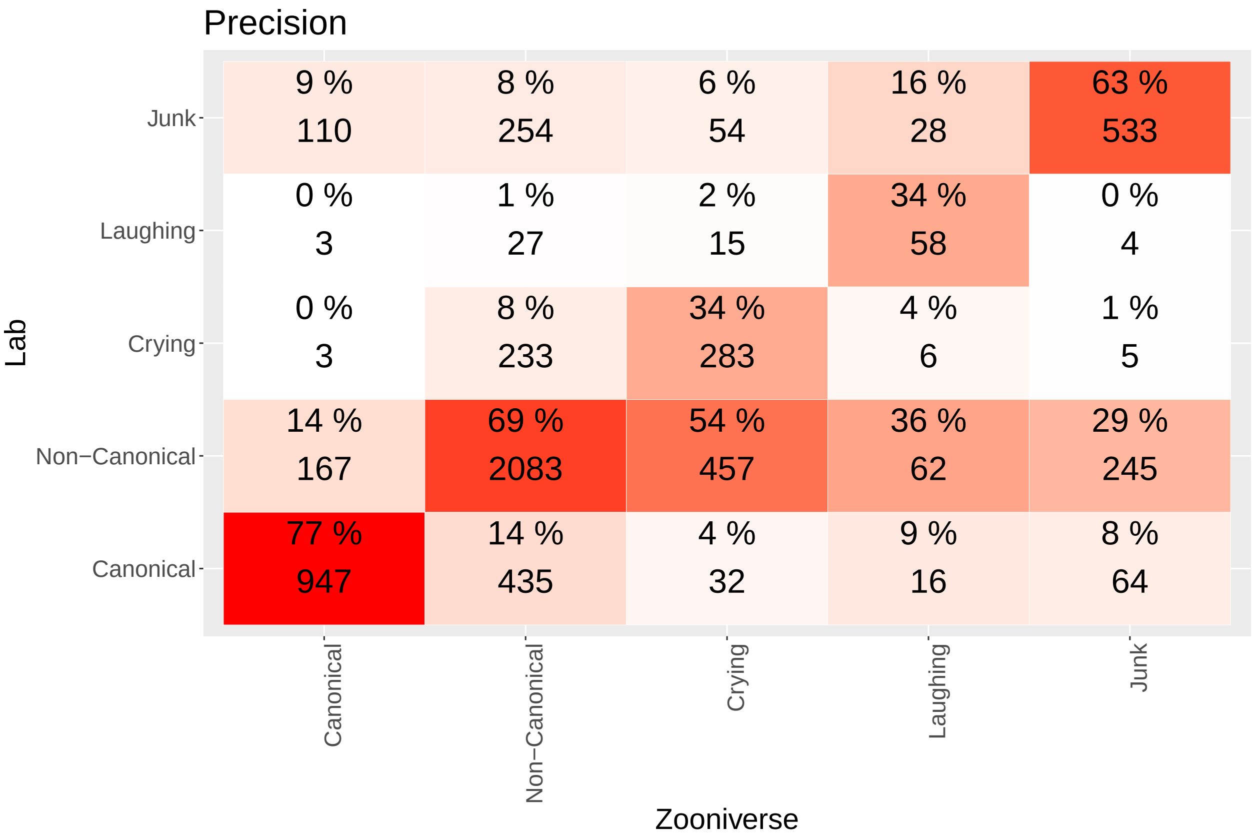
```
## Confusion Matrix and Statistics
## 
##                 Zooniverse
## Lab             Canonical Non-Canonical Crying Laughing Junk
##   Canonical           947           435     32       16   64
##   Non-Canonical       167          2083    457       62  245
##   Crying                3           233    283        6    5
##   Laughing              3            27     15       58    4
##   Junk                110           254     54       28  533
## 
## Overall Statistics
## 
##                 Accuracy : 0.6375
##                   95% CI : (0.6253, 0.6495)
##      No Information Rate : 0.4951
##      P-Value [Acc > NIR] : < 2.2e-16
## 
##                    Kappa : 0.4612
```

```
##
##  Mcnemar's Test P-Value : < 2.2e-16
##
## Statistics by Class:
##
##                      Class: Canonical Class: Non-Canonical Class: Crying
## Sensitivity                    0.7699                0.6870       0.33650
## Specificity                    0.8882                0.6989       0.95325
## Pos Pred Value                 0.6339                0.6911       0.53396
## Neg Pred Value                 0.9389                0.6949       0.90025
## Prevalence                     0.2008                0.4951       0.13733
## Detection Rate                 0.1546                0.3401       0.04621
## Detection Prevalence           0.2440                0.4922       0.08654
## Balanced Accuracy              0.8291                0.6930       0.64488
##                      Class: Laughing Class: Junk
## Sensitivity                 0.341176     0.62632
## Specificity                 0.991770     0.91542
## Pos Pred Value              0.542056     0.54443
## Neg Pred Value              0.981386     0.93819
## Prevalence                  0.027760     0.13896
## Detection Rate              0.009471     0.08703
## Detection Prevalence        0.017472     0.15986
## Balanced Accuracy           0.666473     0.77087
```

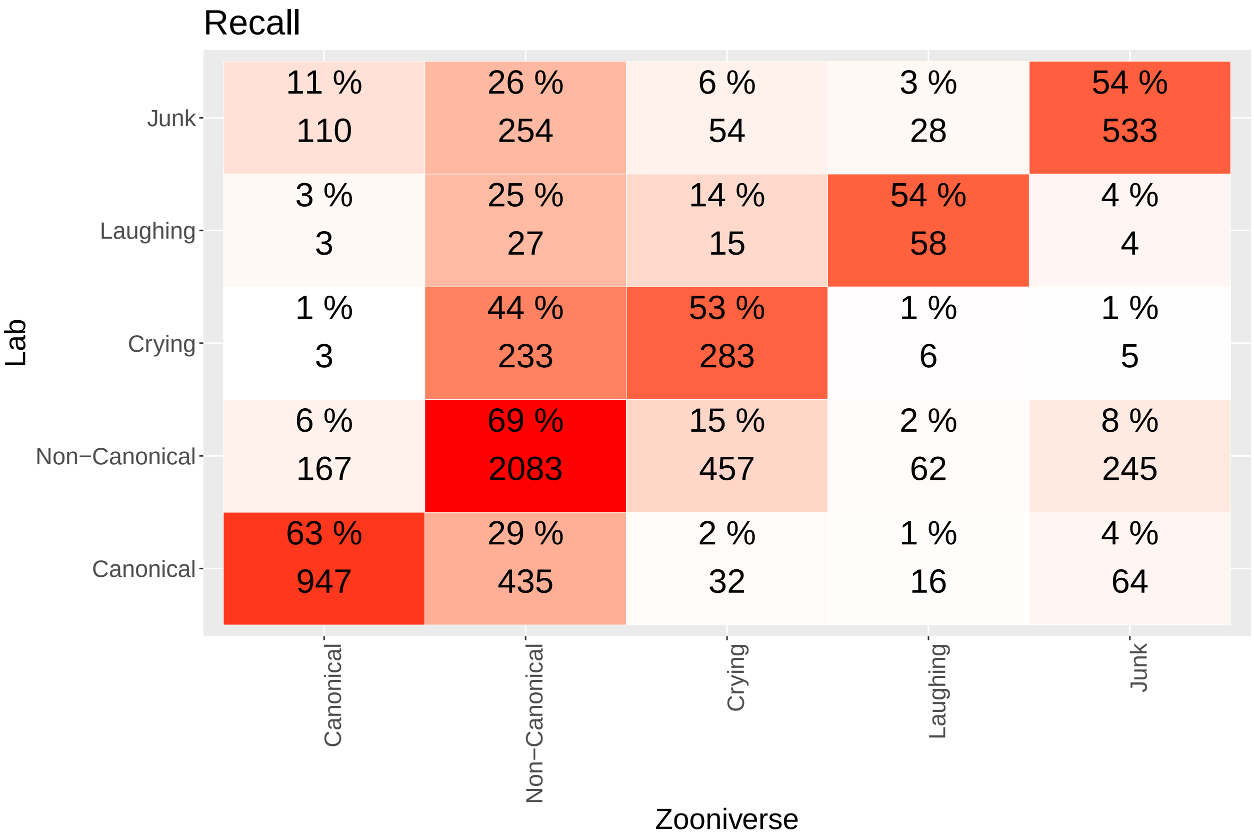
```
colsums=colSums(conf_tab)
my_conf_tab=conf_tab
for(i in 1:5) my_conf_tab[,i]=my_conf_tab[,i]/colsums[i]
colSums(my_conf_tab)
```

```
##     Canonical Non-Canonical        Crying      Laughing          Junk
##             1             1             1             1             1
```

```
prop_cat=data.frame(my_conf_tab*100) #generates precision because columns
prop_cat$id=paste(prop_cat$Lab,prop_cat$Zooniverse)
colnames(prop_cat)[3]<-"pr"
data.frame(conf_tab)->stall
stall$id=paste(stall$Lab,stall$Zooniverse)
stall=merge(stall,prop_cat[c("id","pr")])
ggplot(data = stall, mapping = aes(y = Lab, x=Zooniverse)) +
 geom_tile(aes(fill= rescale(pr)), colour = "white") +
  geom_text(aes(label = paste(round(pr),"%")), vjust = -1,size=8) +
  geom_text(aes(label = Freq), vjust = 1,size=8) +
  scale_fill_gradient(low = "white", high = "red", name = "Proportion") +
     theme(legend.position = "none") +
  xlab("Zooniverse") + ylab("Lab") +
  ggtitle("Precision")+theme(text = element_text(size=20),
        axis.text.x = element_text(angle=90, hjust=1))
```



```
prop_cat=data.frame(conf_tab/rowSums(conf_tab)*100)  #generates recall because rows
prop_cat$id=paste(prop_cat$Lab,prop_cat$Zooniverse)
colnames(prop_cat)[3]<-"rec"
data.frame(conf_tab)->stall
stall$id=paste(stall$Lab,stall$Zooniverse)
stall=merge(stall,prop_cat[c("id","rec")])
ggplot(data = stall, mapping = aes(y = Lab, x=Zooniverse)) +
 geom_tile(aes(fill= rescale(rec)), colour = "white") +
  geom_text(aes(label = paste(round(rec),"%")), vjust = -1,size=8) +
  geom_text(aes(label = Freq), vjust = 1,size=8) +
  scale_fill_gradient(low = "white", high = "red", name = "Proportion") +
     theme(legend.position = "none") +
  xlab("Zooniverse") + ylab("Lab") +
  ggtitle("Recall")+theme(text = element_text(size=20),
        axis.text.x = element_text(angle=90, hjust=1))
```



Recall

Lab
Junk
Laughing
Crying
Non−Canonical
Canonical
11 %
110
26 %
254
6 %
54
3 %
28
54 %
533
3 %
3
25 %
27
14 %
15
54 %
58
4 %
4
1 %
3
44 %
233
53 %
283
1 %
6
1 %
5
6 %
167
69 %
2083
15 %
457
2 %
62
8 %
245
63 %
947
29 %
435
2 %
32
1 %
16
4 %
64
Canonical
Non−Canonical
Crying
Laughing
Junk
Zooniverse