Analysis of Available Data

Load the corpora

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.1     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.1
## v purrr     1.0.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(tidymodels)
```

```
## -- Attaching packages --------------------------------------- tidymodels 1.2.0 --
## v broom        1.0.5     v rsample      1.2.1
## v dials        1.3.0     v tune         1.2.1
## v infer        1.0.7     v workflows    1.1.4
## v modeldata    1.4.0     v workflowsets 1.1.0
## v parsnip      1.2.1     v yardstick    1.3.2
## v recipes      1.1.0
## -- Conflicts ------------------------------------------ tidymodels_conflicts() --
## x scales::discard() masks purrr::discard()
## x dplyr::filter()   masks stats::filter()
## x recipes::fixed()  masks stringr::fixed()
## x dplyr::lag()      masks stats::lag()
## x yardstick::spec() masks readr::spec()
## x recipes::step()   masks stats::step()
## * Search for functions across packages at https://www.tidymodels.org/find/
```

```
library(jsonlite)
```

```
##
## Attaching package: 'jsonlite'
##
## The following object is masked from 'package:purrr':
##
##     flatten
```

```
set.seed(42)

load_kuk_subcorpus_metadata <- function(crp) {
  read_tsv(paste(c(
    "../corpora/KUK_1.0/metadata/", crp, "_DocumentFileFormat.tsv"
  ), collapse = "")) %>%
```

```
    filter(FileFormat == "TXT") %>%
    full_join(
      read_tsv(paste(c(
        "../corpora/KUK_1.0/metadata/",
        crp,
        "_DocumentIdentificationGenreProperties.tsv"
      ), collapse = "")),
      by = "KUK_ID"
    ) %>%
    mutate(across(where(is.numeric), as.character)) %>%
    mutate(subcorpus = crp) %>%
    select(KUK_ID, FileName, FileFormat, FolderPath, subcorpus, everything())
}

kuky_orig <- fromJSON("../corpora/KUKY/argumentative.json")$documents %>%
  as_tibble() %>%
  bind_rows(
    fromJSON("../corpora/KUKY/normative.json")$documents %>% as_tibble()
  ) %>%
  rename(KUK_ID = doc_id) %>%
  select(!c(plainText, doc_name)) %>%
  select(KUK_ID, everything())

kuky_kuk <- load_kuk_subcorpus_metadata("KUKY") %>%
  filter(FolderPath == "data/KUKY/TXT")
```

```
## Rows: 448 Columns: 4
## -- Column specification --------------------------------------------------------
## Delimiter: "\t"
## chr (4): KUK_ID, FileName, FileFormat, FolderPath
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
## Rows: 224 Columns: 12
## -- Column specification --------------------------------------------------------
## Delimiter: "\t"
## chr (8): KUK_ID, SourceDB, Anonymized, RecipientType, RecipientIndividuation...
## lgl (4): SourceID, DocumentTitle, ClarityPursuit, Bindingness
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
kuky <- kuky_kuk %>% full_join(kuky_orig, by = "KUK_ID")
czcdc <- load_kuk_subcorpus_metadata("CzCDC")
```

```
## Rows: 237723 Columns: 4
## -- Column specification --------------------------------------------------------
## Delimiter: "\t"
## chr (4): KUK_ID, FileName, FileFormat, FolderPath
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
## Rows: 237723 Columns: 12
## -- Column specification --------------------------------------------------------
```

```
## Delimiter: "\t"
## chr (10): KUK_ID, SourceDB, SourceID, DocumentTitle, Anonymized, RecipientTy...
## lgl  (2): ClarityPursuit, Bindingness
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
eso <- load_kuk_subcorpus_metadata("ESO")
```

```
## Rows: 11230 Columns: 4
## -- Column specification --------------------------------------------------------
## Delimiter: "\t"
## chr (3): KUK_ID, FileFormat, FolderPath
## dbl (1): FileName
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
## Rows: 5615 Columns: 12
## -- Column specification --------------------------------------------------------
## Delimiter: "\t"
## chr (10): KUK_ID, SourceDB, SourceID, DocumentTitle, Anonymized, RecipientTy...
## lgl  (2): ClarityPursuit, Bindingness
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
frbo <- load_kuk_subcorpus_metadata("FrBo") %>%
  # load metadata for FrBo updated with Quality (=Readability)
  bind_rows(
    read_csv("../corpora/FrBo_contents.csv") %>%
      mutate(Readability = str_to_lower(Quality)) %>%
      select(!Quality)
  ) %>%
  # and move the Quality values to the original rows
  arrange(KUK_ID) %>%
  group_by(KUK_ID) %>%
  fill(Readability, .direction = "up") %>%
  ungroup() %>%
  filter(!is.na(FileName))
```

```
## Rows: 638 Columns: 4
## -- Column specification --------------------------------------------------------
## Delimiter: "\t"
## chr (4): KUK_ID, FileName, FileFormat, FolderPath
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
## Rows: 319 Columns: 12
## -- Column specification --------------------------------------------------------
## Delimiter: "\t"
## chr (10): KUK_ID, SourceDB, SourceID, DocumentTitle, Anonymized, RecipientTy...
## lgl  (2): ClarityPursuit, Bindingness
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
## Rows: 310 Columns: 13
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (11): KUK_ID, SourceDB, SourceID, DocumentTitle, Quality, Anonymized, Re...
## lgl  (2): ClarityPursuit, Bindingness
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
lifrlaw <- load_kuk_subcorpus_metadata("LiFRLaw")
```

```
## Rows: 36 Columns: 4
## -- Column specification --------------------------------------------------------
## Delimiter: "\t"
## chr (4): KUK_ID, FileName, FileFormat, FolderPath
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
## Rows: 18 Columns: 11
## -- Column specification --------------------------------------------------------
## Delimiter: "\t"
## chr (9): KUK_ID, SourceDB, SourceID, DocumentTitle, Anonymized, Recipient Ty...
## lgl (2): ClarityPursuit, Bindingness
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
ombuflyers <- load_kuk_subcorpus_metadata("OmbuFlyers")
```

```
## Rows: 234 Columns: 4
## -- Column specification --------------------------------------------------------
## Delimiter: "\t"
## chr (4): KUK_ID, FileName, FileFormat, FolderPath
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
## Rows: 117 Columns: 12
## -- Column specification --------------------------------------------------------
## Delimiter: "\t"
## chr (8): KUK_ID, DocumentTitle, Anonymized, RecipientType, RecipientIndividu...
## lgl (4): SourceDB, SourceID, ClarityPursuit, Bindingness
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
df <- kuky %>%
  bind_rows(czcdc) %>%
  bind_rows(eso) %>%
  bind_rows(frbo) %>%
  bind_rows(lifrlaw) %>%
  bind_rows(ombuflyers)

str(df)
```

```
## tibble [244,016 x 35] (S3: tbl_df/tbl/data.frame)
##  $ KUK_ID                  : chr [1:244016] "671918e2c6537d54ff0626db" "671918e2c6537d54ff0626dc" "67
```

```
##  $ FileName                : chr [1:244016] "orig_Certifikáty autorizovaných inspektorů" "red_Co je
##  $ FileFormat              : chr [1:244016] "TXT" "TXT" "TXT" "TXT" ...
##  $ FolderPath              : chr [1:244016] "data/KUKY/TXT" "data/KUKY/TXT" "data/KUKY/TXT" "data/KU
##  $ subcorpus               : chr [1:244016] "KUKY" "KUKY" "KUKY" "KUKY" ...
##  $ SourceDB                : chr [1:244016] "SourceDB" "SourceDB" "SourceDB" "SourceDB" ...
##  $ SourceID                : chr [1:244016] NA NA NA NA ...
##  $ DocumentTitle           : chr [1:244016] NA NA NA NA ...
##  $ ClarityPursuit          : logi [1:244016] NA NA NA NA NA NA ...
##  $ Anonymized.x            : chr [1:244016] "No" "No" "No" "No" ...
##  $ RecipientType.x         : chr [1:244016] "natural person" "natural person" "natural person" "natu
##  $ RecipientIndividuation.x: chr [1:244016] "public" "public" "public" "public" ...
##  $ AuthorType.x            : chr [1:244016] "individual" "individual" "individual" "authority" ...
##  $ Objectivity.x           : chr [1:244016] "quasiobjective" "quasiobjective" "quasiobjective" "quasi
##  $ LegalActType.x          : chr [1:244016] "normative" "normative" "normative" "normative" ...
##  $ Bindingness.x           : logi [1:244016] FALSE FALSE FALSE FALSE FALSE FALSE ...
##  $ Readability             : chr [1:244016] "low" "high" "low" "low" ...
##  $ SyllogismBased          : chr [1:244016] "false" "false" "false" "false" ...
##  $ DocumentVersion         : chr [1:244016] "Original" "Redesign" "Original" "Original" ...
##  $ ParentDocumentID        : chr [1:244016] NA NA NA NA ...
##  $ LegalActType.y          : chr [1:244016] "normative" "normative" "normative" "normative" ...
##  $ Objectivity.y           : chr [1:244016] "quasiobjective" "quasiobjective" "quasiobjective" "quasi
##  $ Bindingness.y           : logi [1:244016] FALSE FALSE FALSE FALSE FALSE FALSE ...
##  $ AuthorType.y            : chr [1:244016] "individual" "individual" "individual" "authority" ...
##  $ RecipientType.y         : chr [1:244016] "natural person" "natural person" "natural person" "natu
##  $ RecipientIndividuation.y: chr [1:244016] "public" "public" "public" "public" ...
##  $ Anonymized.y            : chr [1:244016] "No" "No" "No" "No" ...
##  $ Anonymized              : chr [1:244016] NA NA NA NA ...
##  $ RecipientType           : chr [1:244016] NA NA NA NA ...
##  $ RecipientIndividuation  : chr [1:244016] NA NA NA NA ...
##  $ AuthorType              : chr [1:244016] NA NA NA NA ...
##  $ Objectivity             : chr [1:244016] NA NA NA NA ...
##  $ LegalActType            : chr [1:244016] NA NA NA NA ...
##  $ Bindingness             : logi [1:244016] NA NA NA NA NA NA ...
##  $ Recipient Type          : chr [1:244016] NA NA NA NA ...
```

Properties of KUKY

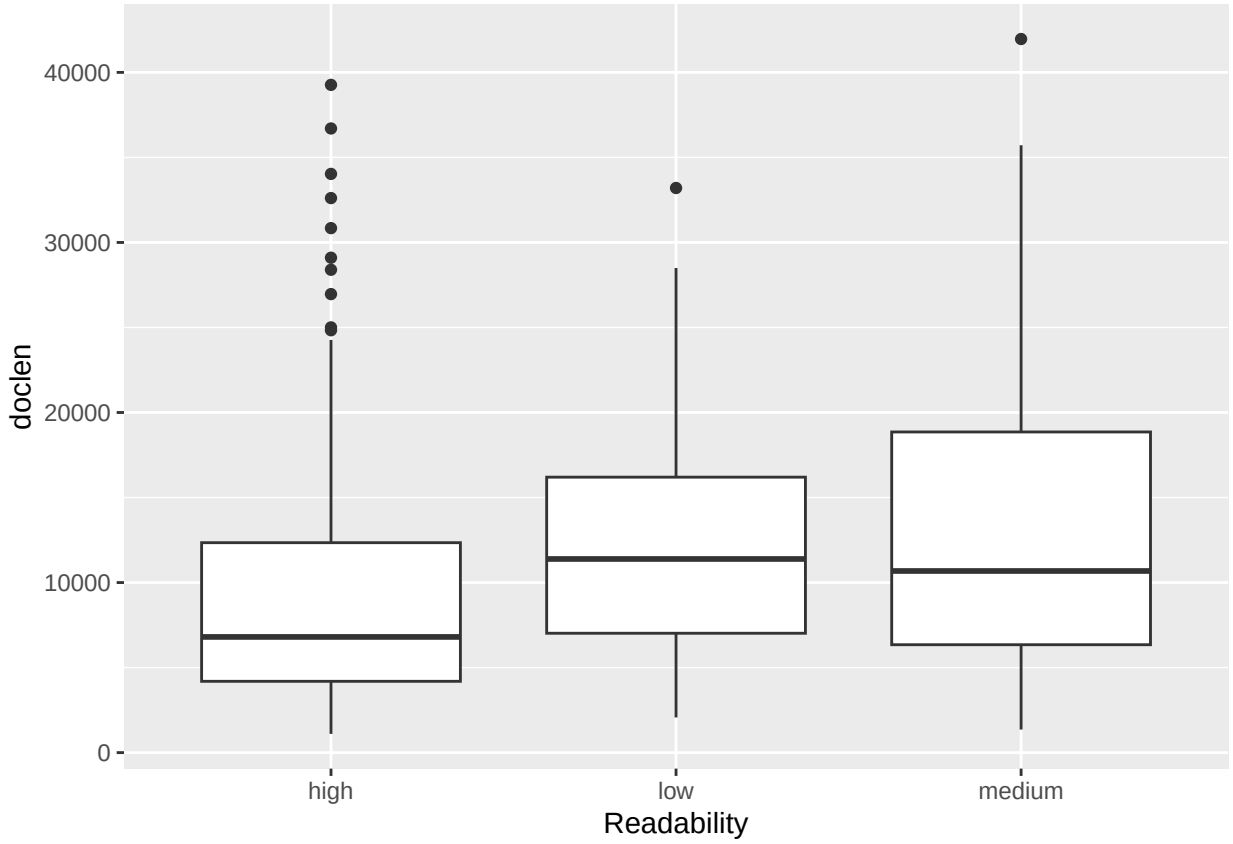
```
kuky_properties_df <- fromJSON(
  "../corpora/KUKY/argumentative.json"
)$documents %>%
  as_tibble() %>%
  bind_rows(
    fromJSON("../corpora/KUKY/normative.json")$documents %>% as_tibble()
  ) %>%
  rename(KUK_ID = doc_id) %>%
  mutate(doclen = str_length(plainText))

print(kuky_properties_df %>% group_by(Readability) %>% count())
```

```
## # A tibble: 3 x 2
## # Groups:   Readability [3]
##   Readability     n
##   <chr>       <int>
## 1 high          125
```

```
## 2 low            38
## 3 medium         61
```

```
kuky_properties_df %>% ggplot(aes(x = Readability, y = doclen)) +
  geom_boxplot()
```



Quick peek into other parts of the data set:

Subcorpus	Low # of chars	High # of chars
CzCDC/ConCo	2.000	18.000
CzCDC/SupAdmCo	3.000	30.000
CzCDC/SupCo	3.000	10.000
ESO	7.000	40.000
FrBo/articles	4.000	15.000

Filter out duplicates

Some subcorpora overlap (*FrBo* with *ESO*, and multiple subcorpora with *KUKY*).

The usage of documents with ClarityPursuit == NA is questionable, let's exclude such documents. This effectively comes with a price of excluding the whole *ESO* subcorpus.

The usage of documents with ClarityPursuit == TRUE is also questionable as they're not reviewed in the same manner as the documents from KUKY, yet at the same time they are less likely to be as "unreadable" as the documents with ClarityPursuit == FALSE. Such documents could very well be readable, interfering with the training process. This effectively comes with a price of excluding the whole *FrBo/analyses* subcorpus.

After filtering ClarityPursuit == NA out, the only remaining overlaps are with *KUKY*. Let's keep the documents from *KUKY* as they are associated with a more careful readability evaluation.

```
# display duplicate file entries
df %>%
  group_by(FileName) %>%
  mutate(n = n()) %>%
  filter(n > 1) %>%
  select(FileName, subcorpus, Readability, ClarityPursuit) %>%
  arrange(FileName) %>%
  print(n = 80)
```

```
## # A tibble: 80 x 4
## # Groups:   FileName [40]
##    FileName                                subcorpus Readability ClarityPursuit
##    <chr>                                   <chr>     <chr>       <lgl>
##  1 100                                     ESO       <NA>        NA
##  2 100                                     FrBo      good        TRUE
##  3 102                                     ESO       <NA>        NA
##  4 102                                     FrBo      good        TRUE
##  5 110                                     ESO       <NA>        NA
##  6 110                                     FrBo      medium      TRUE
##  7 14                                      ESO       <NA>        NA
##  8 14                                      FrBo      good        TRUE
##  9 142                                     ESO       <NA>        NA
## 10 142                                     FrBo      medium      TRUE
## 11 148                                     ESO       <NA>        NA
## 12 148                                     FrBo      good        TRUE
## 13 152                                     ESO       <NA>        NA
## 14 152                                     FrBo      good        TRUE
## 15 154                                     ESO       <NA>        NA
## 16 154                                     FrBo      medium      TRUE
## 17 156                                     ESO       <NA>        NA
## 18 156                                     FrBo      good        TRUE
## 19 158                                     ESO       <NA>        NA
## 20 158                                     FrBo      good        TRUE
## 21 16                                      ESO       <NA>        NA
## 22 16                                      FrBo      good        TRUE
## 23 170                                     ESO       <NA>        NA
## 24 170                                     FrBo      medium      TRUE
## 25 176                                     ESO       <NA>        NA
## 26 176                                     FrBo      medium      TRUE
## 27 18                                      ESO       <NA>        NA
## 28 18                                      FrBo      good        TRUE
## 29 190                                     ESO       <NA>        NA
## 30 190                                     FrBo      good        TRUE
## 31 200                                     ESO       <NA>        NA
## 32 200                                     FrBo      good        TRUE
## 33 202                                     ESO       <NA>        NA
## 34 202                                     FrBo      good        TRUE
## 35 204                                     ESO       <NA>        NA
## 36 204                                     FrBo      good        TRUE
## 37 206                                     ESO       <NA>        NA
## 38 206                                     FrBo      good        TRUE
## 39 208                                     ESO       <NA>        NA
```

```
## 40 208                                          FrBo      good        TRUE
## 41 24                                           ESO       <NA>        NA
## 42 24                                           FrBo      good        TRUE
## 43 28                                           ESO       <NA>        NA
## 44 28                                           FrBo      medium      TRUE
## 45 30                                           ESO       <NA>        NA
## 46 30                                           FrBo      good        TRUE
## 47 42                                           ESO       <NA>        NA
## 48 42                                           FrBo      medium      TRUE
## 49 44                                           ESO       <NA>        NA
## 50 44                                           FrBo      good        TRUE
## 51 54                                           ESO       <NA>        NA
## 52 54                                           FrBo      good        TRUE
## 53 68                                           ESO       <NA>        NA
## 54 68                                           FrBo      medium      TRUE
## 55 70                                           ESO       <NA>        NA
## 56 70                                           FrBo      good        TRUE
## 57 76                                           ESO       <NA>        NA
## 58 76                                           FrBo      good        TRUE
## 59 Duchody                                      KUKY      low         NA
## 60 Duchody                                      OmbuFlye~ <NA>        FALSE
## 61 Odpadni-vody                                 KUKY      low         NA
## 62 Odpadni-vody                                 OmbuFlye~ <NA>        FALSE
## 63 ockovani-1_kusv                              KUKY      high        NA
## 64 ockovani-1_kusv                              LiFRLaw   <NA>        TRUE
## 65 ockovani-3_orig                              KUKY      low         NA
## 66 ockovani-3_orig                              LiFRLaw   <NA>        FALSE
## 67 orig_Certifikáty autorizovaných inspekt~     KUKY      low         NA
## 68 orig_Certifikáty autorizovaných inspekt~     FrBo      medium      FALSE
## 69 orig_financovani_politickych_stran           KUKY      low         NA
## 70 orig_financovani_politickych_stran           FrBo      medium      FALSE
## 71 red_Co je to územní plánování_final_při~     KUKY      high        NA
## 72 red_Co je to územní plánování_final_při~     FrBo      good        TRUE
## 73 stavarska-1_kusv                             KUKY      high        NA
## 74 stavarska-1_kusv                             LiFRLaw   <NA>        TRUE
## 75 stavarska-2_orig                             KUKY      low         NA
## 76 stavarska-2_orig                             LiFRLaw   <NA>        FALSE
## 77 zaloba-1_orig                                KUKY      medium      NA
## 78 zaloba-1_orig                                LiFRLaw   <NA>        FALSE
## 79 zaloba-2_kusv                                KUKY      high        NA
## 80 zaloba-2_kusv                                LiFRLaw   <NA>        TRUE
```

```r
# keep only rows where either Readability or ClarityPursuit isn't NA
# and exclude ClarityPursuit == TRUE
df <- df %>%
  filter(!is.na(Readability) | !is.na(ClarityPursuit)) %>%
  filter(ClarityPursuit == FALSE | is.na(ClarityPursuit))

# 7 duplicates remaining
# keep the ones with Readability assessment
df <- df %>%
  group_by(FileName) %>%
  mutate(n = n()) %>%
  ungroup() %>%
```

```
  filter(n == 1 | !is.na(Readability)) %>%
  select(!n)
```

The dataset is now free of overlaps.

Prepare for ML

Classes

```
df <- df %>%
  mutate(class = if_else(Readability %in% c("high", "medium"), "good", "bad"))
```

Data set parameters

```
.split_prop <- 4 / 5 # proportion of testing data in the dataset
.no_folds <- 10 # no. of folds in v-fold cross-validation
.balance <- 3 / 10 # proportion of positive samples in the target dataset

dssize_positive <- count(df %>% filter(class == "good"))[[1, 1]]
dssize_total <- dssize_positive / .balance
dssize_negative <- dssize_total - dssize_positive

cat(c(
  paste(c(
    "Data set size: ", dssize_total, "\n"
  ), collapse = ""),
  paste(c(
    "Positive class size: ", dssize_positive, "\n"
  ), collapse = ""),
  paste(c(
    "Negative class size: ", dssize_negative, "\n"
  ), collapse = ""),
  paste(c(
    "Training data set size: ", dssize_total * .split_prop, "\n"
  ), collapse = ""),
  paste(c(
    "Training positive class size: ", dssize_positive * .split_prop, "\n"
  ), collapse = ""),
  paste(c(
    "Training negative class size: ", dssize_negative * .split_prop, "\n"
  ), collapse = ""),
  paste(c(
    "One fold size: ", (dssize_total * .split_prop) / .no_folds, "\n"
  ), collapse = ""),
  paste(c(
    "One fold positive class size: ", (dssize_positive * .split_prop) / .no_folds, "\n"
  ), collapse = ""),
  paste(c(
    "One fold negative class size: ", (dssize_negative * .split_prop) / .no_folds, "\n"
  ), collapse = ""),
  paste(c(
    "Evaluation data set size: ", dssize_total * (1 - .split_prop), "\n"
```

```
  ), collapse = ""),
  paste(c(
    "Evaluation positive class size: ", dssize_positive * (1 - .split_prop), "\n"
  ), collapse = ""),
  paste(c(
    "Evaluation negative class size: ", dssize_negative * (1 - .split_prop), "\n"
  ), collapse = "")
), quote = FALSE)
```

```
## Data set size: 800
##  Positive class size: 240
##  Negative class size: 560
##  Training data set size: 640
##  Training positive class size: 192
##  Training negative class size: 448
##  One fold size: 64
##  One fold positive class size: 19.2
##  One fold negative class size: 44.8
##  Evaluation data set size: 160
##  Evaluation positive class size: 48
##  Evaluation negative class size: 112
##  FALSE
```

Data set undersampling and split

```
table(df$subcorpus, df$class)
```

```
##
##                  bad   good
##   CzCDC       237723      0
##   FrBo            60     54
##   KUKY            38    186
##   LiFRLaw          3      0
##   OmbuFlyers      50      0
```

```
bads <- df %>%
  filter(class == "bad") %>%
  group_by(subcorpus) %>%
  mutate(subcorpus_size = n()) %>%
  ungroup()

max_negative_subcorpus <- bads %>%
  arrange(-subcorpus_size) %>%
  head(n = 1)

mns_name <- max_negative_subcorpus %>% pull(subcorpus)
mns_size <- max_negative_subcorpus %>% pull(subcorpus_size)
orig_negative_class_size <- bads %>%
  count() %>%
  pull(n)

# target undersample of MNS = target neg. size - other-negative-subcorpora-size
mns_target_size <- dssize_negative - (orig_negative_class_size - mns_size)
```

```
mns_sample <- sample(
  bads %>% filter(subcorpus == mns_name) %>% pull(KUK_ID), mns_target_size
)

df <- df %>% filter(
  class == "good" |
    subcorpus != mns_name |
    KUK_ID %in% mns_sample
)

table(df$subcorpus, df$class)
```

```
##
##              bad good
##   CzCDC      409    0
##   FrBo        60   54
##   KUKY        38  186
##   LiFRLaw      3    0
##   OmbuFlyers  50    0
```

```
df_split <- df %>% initial_split(prop = .split_prop)
training_set <- training(df_split)
evaluation_set <- testing(df_split)

folds <- vfold_cv(training_set, v = .no_folds, strata = class)

print(df_split)
```

```
## <Training/Testing/Total>
## <640/160/800>
```

```
print(folds)
```

```
## #  10-fold cross-validation using stratification
## # A tibble: 10 x 2
##    splits           id
##    <list>           <chr>
##  1 <split [576/64]> Fold01
##  2 <split [576/64]> Fold02
##  3 <split [576/64]> Fold03
##  4 <split [576/64]> Fold04
##  5 <split [576/64]> Fold05
##  6 <split [576/64]> Fold06
##  7 <split [576/64]> Fold07
##  8 <split [576/64]> Fold08
##  9 <split [576/64]> Fold09
## 10 <split [576/64]> Fold10
```