**# Análisis cuantitativo de Índice Literario**

# Paso 1

# El primer paso es extraer las reseñas del archivo Índice Literario en formato txt

# Se extraen las reseñas en sí mismas, sin los datos bibliográficos que las acompañan

texto <- readLines("https://raw.githubusercontent.com/NGG-UNIR/Indice-Literario/master/Indice\_Literario.txt")

dir.create("Resenas")

posicion\_datos <- grep("^##N\_DATOSOBRA", texto)

posicion\_resena <- grep("^##N\_RESENAOBRA", texto)

posicion\_finresena <- grep("^##N\_FINRESENA", texto)

for(x in 1:length(posicion\_datos)){

inicio <- posicion\_resena [x]

fin <- posicion\_finresena[x]

resena <- texto[inicio:fin]

resena <- gsub ("^#.\*$", "", resena, perl=T)

write(resena, paste("Resenas/Resena",

"\_",

formatC(x, width = 3, flag = 0),

".txt",

sep = ""))

}

# Paso 2

# Se crean vectores específicos para clasificar las reseñas por sexo, género literario y mes/año de la publicación

sexo <- c(rep("hombre", 12),

rep("mujer", 1), rep ("hombre", 4), rep ("mujer", 1), rep ("hombre", 22), rep ("mujer", 1), rep ("hombre", 11), rep ("mujer", 1), rep ("hombre", 1), rep ("mujer", 1), rep ("hombre", 1), rep ("mujer", 1), rep ("hombre", 84), rep ("mujer", 1), rep ("hombre", 21), rep ("mujer", 1), rep ("hombre", 11), rep ("mujer", 1), rep ("hombre", 31), rep ("mujer", 2), rep ("hombre", 21), rep ("mujer", 1), rep ("hombre", 5), rep ("mujer", 1), rep ("hombre", 26), rep ("mujer", 2), rep ("hombre", 9), rep ("mujer", 1), rep ("hombre", 13), rep ("mujer", 1), rep ("hombre", 13), rep ("mujer", 1), rep ("hombre", 12), rep ("mujer", 1), rep ("hombre", 18), rep ("mujer", 2), rep ("hombre", 2), rep ("mujer", 2), rep ("hombre", 21), rep ("mujer", 2), rep ("hombre", 11))

anno <- c(rep("06\_1932", 31), rep("08\_1932", 36), rep("10\_1932", 23), rep("11\_1932", 26), rep("12\_1932", 21), rep("01\_1933", 19), rep("02\_1933", 29), rep("03\_1933", 19), rep("04\_1933", 27), rep("05\_1933", 26), rep("06\_1933", 26), rep("08\_1933", 29), rep("10\_1933", 21), rep("11\_1933", 20), rep("12\_1933", 21))

genero <- c(rep("estudio introductorio", 1), rep("novela y narraciones", 3), rep("ensayos literarios", 2), rep("dramatica", 10), rep("poesia", 3), rep("literatura historica", 1), rep("literatura historica", 7), rep("literatura historica", 4), rep("estudio introductorio", 1), rep("novela y narraciones", 11), rep("ensayos literarios", 5), rep("antologias", 1), rep("dramatica", 7), rep("poesia", 3), rep("literatura historica", 1), rep("literatura historica", 7), rep("estudio introductorio", 1), rep("novela y narraciones", 6), rep("ensayos literarios", 4), rep("literatura historica", 1), rep("literatura historica", 1), rep("literatura historica", 2), rep("literatura historica", 8), rep("estudio introductorio", 1), rep("novela y narraciones", 7), rep("ensayos literarios", 2), rep("dramatica", 5), rep("literatura historica", 2), rep("literatura historica", 1), rep("literatura historica", 1), rep("literatura historica", 1), rep("literatura historica", 3), rep("literatura historica", 3), rep("estudio introductorio", 1), rep("novela y narraciones", 4), rep("ensayos literarios", 2), rep("antologias", 1), rep("poesia", 2), rep("dramatica", 3), rep("literatura historica", 4), rep("literatura historica", 4), rep("estudio introductorio", 1), rep("novela y narraciones", 7), rep("ensayos literarios", 2), rep("poesia", 1), rep("dramatica", 6), rep("literatura historica", 1), rep("literatura historica", 1), rep("estudio introductorio", 1), rep("novela y narraciones", 4), rep("ensayos literarios", 3), rep("dramatica", 13), rep("poesia", 5), rep("literatura historica", 2), rep("literatura historica", 1), rep("estudio introductorio", 1), rep("novela y narraciones", 5), rep("ensayos literarios", 2), rep("dramatica", 3), rep("literatura historica", 8), rep("estudio introductorio", 1), rep("novela y narraciones", 9), rep("ensayos literarios", 3), rep("poesia", 1), rep("dramatica", 7), rep("literatura historica", 6), rep("estudio introductorio", 1), rep("novela y narraciones", 8), rep("literatura para ninnos", 2), rep("poesia", 1), rep("dramatica", 4), rep("literatura historica", 4), rep("literatura historica", 6), rep("estudio introductorio", 1), rep("novela y narraciones", 11), rep("ensayos literarios", 2), rep("literatura para ninnos", 1), rep("viajes", 1), rep("poesia", 2), rep("literatura historica", 1), rep("literatura historica", 2), rep("literatura historica", 3), rep("literatura historica", 2), rep("estudio introductorio", 1), rep("novela y narraciones", 10), rep ("literatura para ninnos", 4), rep("antologias", 1), rep("poesia", 5), rep("dramatica", 1), rep("literatura historica", 2), rep ("literatura historica", 5), rep("estudio introductorio", 1), rep("novela y narraciones", 6), rep("ensayos literarios", 3), rep("viajes", 1), rep("dramatica", 2), rep("literatura historica", 2), rep("literatura historica", 1), rep("literatura historica", 5), rep("estudio introductorio", 1), rep("novela y narraciones", 7), rep("poesia", 2), rep("dramatica", 3), rep("literatura historica", 2), rep("literatura historica", 5), rep("estudio introductorio", 1), rep("novela y narraciones", 5), rep("poesia", 3), rep("literatura para ninnos", 2), rep("ensayos literarios", 2), rep("dramatica", 2), rep("literatura historica", 2), rep("literatura historica", 4))

# Paso 3

# Crear una tabla con toda la información

# Esta tabla se llamará “Resenas\_tabla” y a ella acudiremos en diversas fases del proyecto cuando necesitemos toda la información original

# 3.1 - Cargamos las librerías

library (tidyverse)

library (tidytext)

# 3.2 Creamos la tabla con las reseñas

ficheros <- list.files(path ="Resenas", pattern = "\*.txt")

Resenas\_tabla <- tibble(anno= character(), genero= character(), sexo = character(), numresena = numeric (), parrafo = numeric (), texto = character())

for (i in 1:length(ficheros)){

resena <- readLines(paste("Resenas", ficheros[i], sep = "/"))

resena <- resena[resena != ""]

temporal <- tibble(anno=anno[i], genero = genero[i], sexo= sexo[i], numresena = i, parrafo = seq\_along (resena), texto = resena)

Resenas\_tabla <- bind\_rows(Resenas\_tabla, temporal)

}

# Paso 4

# Primer análisis: conteo simple de palabras

# 4.1 – dividir en palabras token

# Para ello se crea la tabla “resenas\_tabla\_palabras” a la cual se hará referencia en diversas partes del trabajo

Resenas\_tabla\_palabras <- Resenas\_tabla %>%

unnest\_tokens(palabra, texto)

# 4.2 – conteo de palabras

# Para crear la tabla resumen del corpus, se realiza el conteo activando o desactivando algunos filtros (eg: sexo == “mujer” o genero == “novela y narraciones) y se ejecuta o no la función “count (palabra, sort = TRUE)” en función de si queremos conocer el número de palabras token o el número absoluto de palabras

Resenas\_tabla\_palabras %>%

count(palabra, sort = TRUE)

Resenas\_tabla\_palabras %>%

filter(sexo == "mujer") %>%

filter(genero == "novela y narraciones") %>%

count(palabra, sort = TRUE)

Resenas\_tabla\_palabras %>%

filter(genero == "estudio introductorio") %>%

count(palabra, sort = TRUE)

# Paso 5

# Eliminar palabras vacias

# 5.1 obtener las palabras vacias por defecto del sistema

vacias <- get\_stopwords("es")

vacias <- vacias %>%

rename(palabra = word)

# 5.2 generar otras palabras vacias

# Se generan varias listas, agrupando por tipo de palabras vacias

# En Vacias\_de\_Nani son palabras que se repiten bastante y aportan poca información

# En Vacias\_trato elimino algunos términos asociados al trato, con Sr, Don, etc.

Vacias\_de\_Nani <- tibble(palabra = c("autor", "autora", "autores", "autoras", "libro", "libros", "parte", "si", "sino", "uno", "dos", "tres", "así", "vida", "solo", "cuya", "hace", "obra", "obras", "tan", "etc.", "etc", "toda", "mismo", "si", "ello", "ya", "después", "da", "va", "cada", "ser", "vez", "v","índice", "i", "allí", "veces", "embargo", "sr", "don", "doña", "p", "d", "ª"))

# 5.3 Elimino finalmente las palabras vacias de la tabla de palabras

# Se genera una tabla llamana “Resenas\_tabla\_sinvacias” a la que haremos referencia en otras partes del proyecto

Resenas\_tabla\_sinvacias <- Resenas\_tabla\_palabras %>%

anti\_join(vacias) %>%

anti\_join(Vacias\_de\_Nani)

# Paso 6

# Crear el gráfico con la frecuencia absoluta de aparición de palabras

# Nota: se filtra para obtener solamente las principales categorías, ya que algunas categorías (literatura para niños, antologías, etc…) tienen demasiadas pocas reseñas para poder ser analizadas de forma minimamente representativa

Resenas\_tabla\_sinvacias %>%

filter(genero == "novela y narraciones" | genero == "poesia" | genero == "dramatica" | genero == "ensayos literarios" | genero == "estudio introductorio" | genero == "literatura historica") %>%

group\_by (genero) %>%

count (palabra, sort = T) %>%

top\_n (15) %>%

ungroup %>%

mutate(palabra = reorder\_within(palabra,n,genero)) %>%

ggplot(aes(palabra,

n,

fill = genero)) +

geom\_bar(stat = "identity") +

facet\_wrap(~genero,

scales = "free\_y") +

labs(x = "",

y = "Frecuencia absoluta") +

coord\_flip() +

scale\_x\_reordered() +

theme(legend.position="none")

# Paso 7

# Lematizar

# El objetivo es perfeccionar el gráfico anterior que muestra la frecuencia absoluta de aparición de palabras

# Para ello, se utiliza la librería de UDPIPE

# Es un paso bastante extenso en código porque un fragmento se repite múltiples veces

# 7.1 cargamos el paquete udpipe y el modelo ancora

library(udpipe)

modelo\_ancora <- udpipe\_load\_model(file = 'spanish-ancora-ud-2.4-190531.udpipe')

# 7.2 Crear la tabla en la que se van a cargar las reseñas lematizadas con udpipe

# Se crea la tabla “Resenas\_Udpipe” a la que se hará referencia a lo largo del trabajo

Resenas\_Udpipe <- tibble(parrafo\_id = integer(),

enunciado\_id = integer(),

enunciado = character(),

token\_id = character(),

token = character(),

lema = character(),

upos = character(),

xpos = character(),

rasgos = character(),

genero = character(),

sexo = character())

# 7.3 Es necesario cambiar un parámetro en el lector de texto para que el proceso de lematizar lea bien acentos y caracteres similares

Resenas\_tabla$texto <- iconv(Resenas\_tabla$texto, from = "Latin1", to = "UTF-8")

# 7.4 – Este paso lo vamos a repetir muchas veces

# Se trata de cargar las reseñas, filtrarlas por sexo / género, analizar sus lemas con udpipe, y cargarlas en la tabla “Resenas\_Udpipe” ya mencionada

# 7.4.1 – hombre & novelas y narraciones

temporal <- Resenas\_tabla %>%

filter(sexo == "hombre") %>%

filter(genero == "novela y narraciones") %>%

select(texto)

analisis <- as\_tibble(udpipe\_annotate(modelo\_ancora,

temporal$texto))

analisis <- analisis %>%

add\_column(sexo = "hombre") %>%

add\_column(genero = "novela y narraciones") %>%

select(-(paragraph\_id),

-(deps),

-(misc),

-(head\_token\_id),

-(dep\_rel)) %>%

rename(parrafo\_id = doc\_id,

enunciado\_id = sentence\_id,

enunciado = sentence,

lema = lemma,

rasgos = feats) %>%

mutate(parrafo\_id = as.numeric(str\_extract(parrafo\_id, "\\d+")))

Resenas\_Udpipe <- bind\_rows(Resenas\_Udpipe, analisis)

# 7.4.2 – mujer & novelas y narraciones

temporal <- Resenas\_tabla %>%

filter(sexo == "mujer") %>%

filter(genero == "novela y narraciones") %>%

select(texto)

analisis <- as\_tibble(udpipe\_annotate(modelo\_ancora,

temporal$texto))

analisis <- analisis %>%

add\_column(sexo = "mujer") %>%

add\_column(genero = "novela y narraciones") %>%

select(-(paragraph\_id),

-(deps),

-(misc),

-(head\_token\_id),

-(dep\_rel)) %>%

rename(parrafo\_id = doc\_id,

enunciado\_id = sentence\_id,

enunciado = sentence,

lema = lemma,

rasgos = feats) %>%

mutate(parrafo\_id = as.numeric(str\_extract(parrafo\_id, "\\d+")))

Resenas\_Udpipe <- bind\_rows(Resenas\_Udpipe, analisis)

# 7.4.3 – hombre & estudio introductorio

temporal <- Resenas\_tabla %>%

filter(sexo == "hombre") %>%

filter(genero == "estudio introductorio") %>%

select(texto)

analisis <- as\_tibble(udpipe\_annotate(modelo\_ancora,

temporal$texto))

analisis <- analisis %>%

add\_column(sexo = "hombre") %>%

add\_column(genero = "estudio introductorio") %>%

select(-(paragraph\_id),

-(deps),

-(misc),

-(head\_token\_id),

-(dep\_rel)) %>%

rename(parrafo\_id = doc\_id,

enunciado\_id = sentence\_id,

enunciado = sentence,

lema = lemma,

rasgos = feats) %>%

mutate(parrafo\_id = as.numeric(str\_extract(parrafo\_id, "\\d+")))

Resenas\_Udpipe <- bind\_rows(Resenas\_Udpipe, analisis)

# 7.4.4 – hombre & antologías

temporal <- Resenas\_tabla %>%

filter(sexo == "hombre") %>%

filter(genero == "antologias") %>%

select(texto)

analisis <- as\_tibble(udpipe\_annotate(modelo\_ancora,

temporal$texto))

analisis <- analisis %>%

add\_column(sexo = "hombre") %>%

add\_column(genero = "antologias") %>%

select(-(paragraph\_id),

-(deps),

-(misc),

-(head\_token\_id),

-(dep\_rel)) %>%

rename(parrafo\_id = doc\_id,

enunciado\_id = sentence\_id,

enunciado = sentence,

lema = lemma,

rasgos = feats) %>%

mutate(parrafo\_id = as.numeric(str\_extract(parrafo\_id, "\\d+")))

Resenas\_Udpipe <- bind\_rows(Resenas\_Udpipe, analisis)

# 7.4.5 – hombre & dramatica

temporal <- Resenas\_tabla %>%

filter(sexo == "hombre") %>%

filter(genero == "dramatica") %>%

select(texto)

analisis <- as\_tibble(udpipe\_annotate(modelo\_ancora,

temporal$texto))

analisis <- analisis %>%

add\_column(sexo = "hombre") %>%

add\_column(genero = "dramatica") %>%

select(-(paragraph\_id),

-(deps),

-(misc),

-(head\_token\_id),

-(dep\_rel)) %>%

rename(parrafo\_id = doc\_id,

enunciado\_id = sentence\_id,

enunciado = sentence,

lema = lemma,

rasgos = feats) %>%

mutate(parrafo\_id = as.numeric(str\_extract(parrafo\_id, "\\d+")))

Resenas\_Udpipe <- bind\_rows(Resenas\_Udpipe, analisis)

# 7.4.6 – mujer & dramatica

temporal <- Resenas\_tabla %>%

filter(sexo == "mujer") %>%

filter(genero == "dramatica") %>%

select(texto)

analisis <- as\_tibble(udpipe\_annotate(modelo\_ancora,

temporal$texto))

analisis <- analisis %>%

add\_column(sexo = "mujer") %>%

add\_column(genero = "dramatica") %>%

select(-(paragraph\_id),

-(deps),

-(misc),

-(head\_token\_id),

-(dep\_rel)) %>%

rename(parrafo\_id = doc\_id,

enunciado\_id = sentence\_id,

enunciado = sentence,

lema = lemma,

rasgos = feats) %>%

mutate(parrafo\_id = as.numeric(str\_extract(parrafo\_id, "\\d+")))

Resenas\_Udpipe <- bind\_rows(Resenas\_Udpipe, analisis)

# 7.4.7 – hombre & ensayos literarios

temporal <- Resenas\_tabla %>%

filter(sexo == "hombre") %>%

filter(genero == "ensayos literarios") %>%

select(texto)

analisis <- as\_tibble(udpipe\_annotate(modelo\_ancora,

temporal$texto))

analisis <- analisis %>%

add\_column(sexo = "hombre") %>%

add\_column(genero = "ensayos literarios") %>%

select(-(paragraph\_id),

-(deps),

-(misc),

-(head\_token\_id),

-(dep\_rel)) %>%

rename(parrafo\_id = doc\_id,

enunciado\_id = sentence\_id,

enunciado = sentence,

lema = lemma,

rasgos = feats) %>%

mutate(parrafo\_id = as.numeric(str\_extract(parrafo\_id, "\\d+")))

Resenas\_Udpipe <- bind\_rows(Resenas\_Udpipe, analisis)

# 7.4.8 – mujer & ensayos literarios

temporal <- Resenas\_tabla %>%

filter(sexo == "mujer") %>%

filter(genero == "ensayos literarios") %>%

select(texto)

analisis <- as\_tibble(udpipe\_annotate(modelo\_ancora,

temporal$texto))

analisis <- analisis %>%

add\_column(sexo = "mujer") %>%

add\_column(genero = "ensayos literarios") %>%

select(-(paragraph\_id),

-(deps),

-(misc),

-(head\_token\_id),

-(dep\_rel)) %>%

rename(parrafo\_id = doc\_id,

enunciado\_id = sentence\_id,

enunciado = sentence,

lema = lemma,

rasgos = feats) %>%

mutate(parrafo\_id = as.numeric(str\_extract(parrafo\_id, "\\d+")))

Resenas\_Udpipe <- bind\_rows(Resenas\_Udpipe, analisis)

# 7.4.9 – hombre & literatura histórica

temporal <- Resenas\_tabla %>%

filter(sexo == "hombre") %>%

filter(genero == "literatura historica") %>%

select(texto)

analisis <- as\_tibble(udpipe\_annotate(modelo\_ancora,

temporal$texto))

analisis <- analisis %>%

add\_column(sexo = "hombre") %>%

add\_column(genero = "literatura historica") %>%

select(-(paragraph\_id),

-(deps),

-(misc),

-(head\_token\_id),

-(dep\_rel)) %>%

rename(parrafo\_id = doc\_id,

enunciado\_id = sentence\_id,

enunciado = sentence,

lema = lemma,

rasgos = feats) %>%

mutate(parrafo\_id = as.numeric(str\_extract(parrafo\_id, "\\d+")))

Resenas\_Udpipe <- bind\_rows(Resenas\_Udpipe, analisis)

# 7.4.10 – mujer & literatura histórica

temporal <- Resenas\_tabla %>%

filter(sexo == "mujer") %>%

filter(genero == "literatura historica") %>%

select(texto)

analisis <- as\_tibble(udpipe\_annotate(modelo\_ancora,

temporal$texto))

analisis <- analisis %>%

add\_column(sexo = "mujer") %>%

add\_column(genero = "literatura historica") %>%

select(-(paragraph\_id),

-(deps),

-(misc),

-(head\_token\_id),

-(dep\_rel)) %>%

rename(parrafo\_id = doc\_id,

enunciado\_id = sentence\_id,

enunciado = sentence,

lema = lemma,

rasgos = feats) %>%

mutate(parrafo\_id = as.numeric(str\_extract(parrafo\_id, "\\d+")))

Resenas\_Udpipe <- bind\_rows(Resenas\_Udpipe, analisis)

# 7.4.11 – hombre & literatura para niños

temporal <- Resenas\_tabla %>%

filter(sexo == "hombre") %>%

filter(genero == "literatura para ninnos") %>%

select(texto)

analisis <- as\_tibble(udpipe\_annotate(modelo\_ancora,

temporal$texto))

analisis <- analisis %>%

add\_column(sexo = "hombre") %>%

add\_column(genero = "literatura para ninnos") %>%

select(-(paragraph\_id),

-(deps),

-(misc),

-(head\_token\_id),

-(dep\_rel)) %>%

rename(parrafo\_id = doc\_id,

enunciado\_id = sentence\_id,

enunciado = sentence,

lema = lemma,

rasgos = feats) %>%

mutate(parrafo\_id = as.numeric(str\_extract(parrafo\_id, "\\d+")))

Resenas\_Udpipe <- bind\_rows(Resenas\_Udpipe, analisis)

# 7.4.12 – mujer & literatura para niños

temporal <- Resenas\_tabla %>%

filter(sexo == "mujer") %>%

filter(genero == "literatura para ninnos") %>%

select(texto)

analisis <- as\_tibble(udpipe\_annotate(modelo\_ancora,

temporal$texto))

analisis <- analisis %>%

add\_column(sexo = "mujer") %>%

add\_column(genero = "literatura para ninnos") %>%

select(-(paragraph\_id),

-(deps),

-(misc),

-(head\_token\_id),

-(dep\_rel)) %>%

rename(parrafo\_id = doc\_id,

enunciado\_id = sentence\_id,

enunciado = sentence,

lema = lemma,

rasgos = feats) %>%

mutate(parrafo\_id = as.numeric(str\_extract(parrafo\_id, "\\d+")))

Resenas\_Udpipe <- bind\_rows(Resenas\_Udpipe, analisis)

# 7.4.13 – hombre & poesía

temporal <- Resenas\_tabla %>%

filter(sexo == "hombre") %>%

filter(genero == "poesia") %>%

select(texto)

analisis <- as\_tibble(udpipe\_annotate(modelo\_ancora,

temporal$texto))

analisis <- analisis %>%

add\_column(sexo = "hombre") %>%

add\_column(genero = "poesia") %>%

select(-(paragraph\_id),

-(deps),

-(misc),

-(head\_token\_id),

-(dep\_rel)) %>%

rename(parrafo\_id = doc\_id,

enunciado\_id = sentence\_id,

enunciado = sentence,

lema = lemma,

rasgos = feats) %>%

mutate(parrafo\_id = as.numeric(str\_extract(parrafo\_id, "\\d+")))

Resenas\_Udpipe <- bind\_rows(Resenas\_Udpipe, analisis)

# 7.4.14 – mujer & poesía

temporal <- Resenas\_tabla %>%

filter(sexo == "mujer") %>%

filter(genero == "poesia") %>%

select(texto)

analisis <- as\_tibble(udpipe\_annotate(modelo\_ancora,

temporal$texto))

analisis <- analisis %>%

add\_column(sexo = "mujer") %>%

add\_column(genero = "poesia") %>%

select(-(paragraph\_id),

-(deps),

-(misc),

-(head\_token\_id),

-(dep\_rel)) %>%

rename(parrafo\_id = doc\_id,

enunciado\_id = sentence\_id,

enunciado = sentence,

lema = lemma,

rasgos = feats) %>%

mutate(parrafo\_id = as.numeric(str\_extract(parrafo\_id, "\\d+")))

Resenas\_Udpipe <- bind\_rows(Resenas\_Udpipe, analisis)

# 7.4.15 – hombre & viajes

temporal <- Resenas\_tabla %>%

filter(sexo == "hombre") %>%

filter(genero == "viajes") %>%

select(texto)

analisis <- as\_tibble(udpipe\_annotate(modelo\_ancora,

temporal$texto))

analisis <- analisis %>%

add\_column(sexo = "hombre") %>%

add\_column(genero = "viajes") %>%

select(-(paragraph\_id),

-(deps),

-(misc),

-(head\_token\_id),

-(dep\_rel)) %>%

rename(parrafo\_id = doc\_id,

enunciado\_id = sentence\_id,

enunciado = sentence,

lema = lemma,

rasgos = feats) %>%

mutate(parrafo\_id = as.numeric(str\_extract(parrafo\_id, "\\d+")))

Resenas\_Udpipe <- bind\_rows(Resenas\_Udpipe, analisis)

# 7.5 Palabras vacías

# Este proceso ya se hizo en la tabla de palabras, y ahora debe hacerse en la tabla de lemas

# Vuelvo a generar las listas de palabras vacias para llamar a esa variable lema en lugar de palabra

# Adicionalmente, elimino nombres propios

# En el proceso udpipe / ancora aparecen nuevos lemas sin apenas valor que debo eliminar, por ejemplo \u0097 que resulta ser el símbolo de fin de línea.

vacias\_udpipe <- get\_stopwords("es")

vacias\_udpipe <- vacias\_udpipe %>%

rename(lema = word)

Vacias\_de\_Nani\_udpipe <- tibble(lema = c("autor", "autora", "autores", "autoras", "libro", "libros", "parte", "si", "sino", "uno", "dos", "tres", "así", "vida", "solo", "cuya", "hace", "obra", "obras", "tan", "etc.", "etc", "toda", "mismo", "si", "ello", "ya", "después", "da", "va", "cada", "v", "ser", "vez", "allí", "veces", "índice", "i", "ª", "embargo", "Sr", "Don", "Doña", "P", "D", "don", "doña", "", " ", "\u0097", ".\u0097"))

Vacias\_personajes <- tibble(lema = c("María", "Julián", "Pedro", "Juan", "Jorge", "Cristina", "Eva", "Germán", "Dolores", "Leonardo", "Francisco", "Fedor", "Pepe", "Javier", "Mariano", "Fulgencia", "Alejo", "José", "Antonio", "Ramón", "Ignacio", "Teresa", "Elvira", "Eduardo", "Miguel", "Julio", "Juan", "Pablo", "Javier", "Isabel", "Carlos", "Olga", "Marta", "Carmen", "Aurelia", "Almudena", "Luisa", "Susana", "Paloma", "Rubén"))

Resenas\_Udpipe\_sinvacias <- Resenas\_Udpipe %>%

anti\_join(vacias\_udpipe) %>%

anti\_join(Vacias\_de\_Nani\_udpipe) %>%

anti\_join(Vacias\_personajes)

# 7.6 Gráfica de frecuencia absoluta de lemas

# Finalmente, ya puedo replicar la gráfica de frecuencia absoluta, con lemas en vez de con palabras

# Aplico el filtro NOUN y PROPN para que solamente muestre lemas que sean sustantivos o nombres propios

Resenas\_Udpipe\_sinvacias %>%

filter(upos == "NOUN" | upos == "PROPN") %>%

filter(genero == "novela y narraciones" | genero == "poesia" | genero == "dramatica" | genero == "ensayos literarios" | genero == "estudio introductorio" | genero == "literatura historica") %>%

group\_by (genero) %>%

count (lema, sort = T) %>%

top\_n (15) %>%

ungroup %>%

mutate (lema = reorder\_within(lema, n, genero)) %>%

ggplot(aes(lema,

n,

fill = genero)) +

geom\_bar(stat = "identity") +

facet\_wrap(~genero,

scales = "free\_y") +

labs(x = "",

y = "Frecuencia absoluta") +

coord\_flip() +

scale\_x\_reordered() +

theme(legend.position="none")

# 7.7 Gráfica de frecuencia absoluta de nombres propios

# Esta gráfica es interesante para intentar conocer los autores que más se mencionaban

# 7.7.1 palabras vacias (otra vez)

# Resulta que aparecen como nombre propio algunos términos que no lo son.

# Los elimino del análisis usando el método de palabras vacias

Vacias\_nombrespropios <- tibble(lema = c("d", "d.", "San", "genio",".","espectador","poesía","margen","V","III","Antología", "Verso", "novia", "novio", "V.", "palabras", "Madre", "literario", "índice"))

Resenas\_Udpipe\_sinvacias2 <- Resenas\_Udpipe %>%

anti\_join(vacias\_udpipe) %>%

anti\_join(Vacias\_de\_Nani\_udpipe) %>%

anti\_join(Vacias\_personajes) %>%

anti\_join(Vacias\_nombrespropios)

# 7.7.2 Pinto la gráfica de frecuencia absoluta de nombres propios

Resenas\_Udpipe\_sinvacias2 %>%

filter(upos == "PROPN") %>%

filter(genero == "novela y narraciones" | genero == "poesia" | genero == "dramatica" | genero == "ensayos literarios" | genero == "estudio introductorio" | genero == "literatura historica") %>%

group\_by (genero) %>%

count (lema, sort = T) %>%

top\_n (8) %>%

ungroup %>%

mutate (lema = reorder\_within(lema, n, genero)) %>%

ggplot(aes(lema,

n,

fill = genero)) +

geom\_bar(stat = "identity") +

facet\_wrap(~genero,

scales = "free\_y") +

labs(x = "",

y = "Frecuencia absoluta") +

coord\_flip() +

scale\_x\_reordered() +

theme(legend.position="none")

# Paso 8 – Bigramas

# El objetivo es dividir las reseñas en pares de palabras (bigramas)

# 8.1 – Crear la tabla de bigramas a partir de la tabla original de reseñas

Resenas\_bigramas <- Resenas\_tabla %>%

unnest\_tokens(bigrama,

texto,

token = "ngrams",

n = 2)

# 8.2 Separar los bigramas en pares de palabras

Resenas\_bigramas\_separados <- Resenas\_bigramas %>%

separate(bigrama,

c("palabra1", "palabra2"),

sep = " ")

# 8.3 Eliminar las palabras vacias

# 8.3.2 Eliminar las palabras vacias de la tabla de bigramas separados:

Resenas\_bigramas\_sinvacias <- Resenas\_bigramas\_separados %>%

filter(!palabra1 %in% vacias$palabra,

!palabra2 %in% vacias$palabra) %>%

filter(!palabra1 %in% Vacias\_de\_Nani$palabra,

!palabra2 %in% Vacias\_de\_Nani$palabra)

# 8.3.4

# En la instrucción anterior han aparecido líneas con NA, que debo eliminar:

Resenas\_bigramas\_sinvacias <- Resenas\_bigramas\_sinvacias %>%

drop\_na()

# 8.4

# Unir de nuevo los bigramas (ya sin palabras vacías) y filtrar por género del autor reseñado

Resenas\_bigramas\_unidos <- Resenas\_bigramas\_sinvacias %>%

unite(bigrama, palabra1, palabra2, sep = " ")

# 8.5

# Y ya por fin: Hacer la gráfica con los bigramas más presentes en las reseñas, por cada uno de los principales géneros

Resenas\_bigramas\_unidos %>%

filter(genero == "novela y narraciones" | genero == "poesia" | genero == "dramatica" | genero == "ensayos literarios" | genero == "estudio introductorio" | genero == "literatura historica") %>%

count(genero, bigrama, sort = T) %>%

group\_by(genero) %>%

top\_n(7) %>%

ungroup %>%

mutate (bigrama = reorder\_within(bigrama, n, genero)) %>%

ggplot() +

geom\_col(aes(y = n , x = bigrama),

fill = "maroon") +

coord\_flip() +

facet\_wrap(~ genero, ncol = 2, scales = "free") +

theme\_linedraw() +

scale\_x\_reordered() +

labs(x = "Bigramas", y = "Frecuencia")

# Paso 9 - Coocurrencias

# Analizar las coocurrencias es analizar palabras que suelen ir juntas.

# Se pretenden analizar las coocurrencias de forma separada por sexo y por género

# 9.1 cargamos la librería

library("widyr")

# 9.2 cargar las reseñas en tablas, sin palabras vacias, para preparar el análisis

# Este fragmento de código lo vamos a repetir varias veces

# 9.2.1 cargamos en una tabla todas las reseñas (sin palabras vacias)

# esto es para preparar análisis posteriores

Resenas\_todas <- Resenas\_tabla %>%

mutate(seccion = row\_number()) %>%

unnest\_tokens(palabra, texto) %>%

filter(!palabra %in% vacias$palabra) %>%

filter(!palabra %in% Vacias\_de\_Nani$palabra)

# 9.2.2 cargamos en una tabla todas las reseñas de mujeres (sin palabras vacias)

Resenas\_mujer <- Resenas\_tabla %>%

filter(sexo =="mujer") %>%

mutate(seccion = row\_number()) %>%

unnest\_tokens(palabra, texto) %>%

filter(!palabra %in% vacias$palabra) %>%

filter(!palabra %in% Vacias\_de\_Nani$palabra)

# 9.2.3 cargamos en una tabla todas las reseñas dramaticas (sin palabras vacias)

Resenas\_dramaticas <- Resenas\_tabla %>%

filter(genero =="dramatica") %>%

mutate(seccion = row\_number()) %>%

unnest\_tokens(palabra, texto) %>%

filter(!palabra %in% vacias$palabra) %>%

filter(!palabra %in% Vacias\_de\_Nani$palabra)

# 9.2.4 cargamos en una tabla todas las reseñas del estudio introductorio (sin palabras vacias)

Resenas\_estudio\_introductorio <- Resenas\_tabla %>%

filter(genero =="estudio introductorio") %>%

mutate(seccion = row\_number()) %>%

unnest\_tokens(palabra, texto) %>%

filter(!palabra %in% vacias$palabra) %>%

filter(!palabra %in% Vacias\_de\_Nani$palabra)

# 9.2.4 cargamos en una tabla todas las reseñas de novelas y narraciones (sin palabras vacias)

Resenas\_novela\_narraciones <- Resenas\_tabla %>%

filter(genero =="novela y narraciones") %>%

mutate(seccion = row\_number()) %>%

unnest\_tokens(palabra, texto) %>%

filter(!palabra %in% vacias$palabra) %>%

filter(!palabra %in% Vacias\_de\_Nani$palabra)

# 9.2.5 cargamos en una tabla todas las reseñas de ensayos literarios (sin palabras vacias)

Resenas\_ensayos\_literarios <- Resenas\_tabla %>%

filter(genero =="ensayos literarios") %>%

mutate(seccion = row\_number()) %>%

unnest\_tokens(palabra, texto) %>%

filter(!palabra %in% vacias$palabra) %>%

filter(!palabra %in% Vacias\_de\_Nani$palabra)

# 9.2.6 cargamos en una tabla todas las reseñas de ensayos literarios (sin palabras vacias)

Resenas\_literatura\_historica <- Resenas\_tabla %>%

filter(genero =="literatura historica") %>%

mutate(seccion = row\_number()) %>%

unnest\_tokens(palabra, texto) %>%

filter(!palabra %in% vacias$palabra) %>%

filter(!palabra %in% Vacias\_de\_Nani$palabra)

# 9.3 dividimos las tablas de palabras en pares de palabras (es decir, palabras que suelen coincidir en el mismo párrafo)

pares\_palabras\_todas <- Resenas\_todas %>%

pairwise\_count(palabra,

numresena,

sort = T)

pares\_palabras\_mujer <- Resenas\_mujer %>%

pairwise\_count(palabra,

numresena,

sort = T)

# 9.4

# Queremos analizar la correlación entre palabras, es decir la probabilidad de que dos palabras aparezcan ambas en el mismo párrafo

# Se trata de determinar los pares de palabras que están más correlacionados en el texto (es decir, que suelen ir juntos)

# Esto luego se puede filtrar por sexo, por género o para palabras concretas.

#9.4.1 Obtener tablas con las correlaciones, para cada grupo de reseñas creado anteriormente (todas, mujeres, por género…)

# esto es para preparar análisis posteriores

palabras\_correlacion\_todas <- Resenas\_todas %>%

group\_by(palabra) %>%

filter(n() >= 10) %>%

pairwise\_cor(palabra,

numresena,

sort = TRUE)

palabras\_correlacion\_mujer <- Resenas\_mujer %>%

group\_by(palabra) %>%

filter(n() >= 3) %>%

pairwise\_cor(palabra,

numresena,

sort = TRUE)

palabras\_correlacion\_dramatica <- Resenas\_dramaticas %>%

group\_by(palabra) %>%

filter(n() >= 3) %>%

pairwise\_cor(palabra,

numresena,

sort = TRUE)

palabras\_correlacion\_novela\_narraciones <- Resenas\_novela\_narraciones %>%

group\_by(palabra) %>%

filter(n() >= 3) %>%

pairwise\_cor(palabra,

numresena,

sort = TRUE)

palabras\_correlacion\_estudio\_introductorio <- Resenas\_estudio\_introductorio %>%

group\_by(palabra) %>%

filter(n() >= 3) %>%

pairwise\_cor(palabra,

numresena,

sort = TRUE)

palabras\_correlacion\_literatura\_historica <- Resenas\_literatura\_historica %>%

group\_by(palabra) %>%

filter(n() >= 3) %>%

pairwise\_cor(palabra,

numresena,

sort = TRUE)

palabras\_correlacion\_ensayos\_literarios <- Resenas\_ensayos\_literarios %>%

group\_by(palabra) %>%

filter(n() >= 3) %>%

pairwise\_cor(palabra,

numresena,

sort = TRUE)

# 9.5 – PRIMER ANÁLISIS CON COOCURRENCIAS

# TÉRMINOS ASOCIADOS A ESCRITORES

# Se trata de obtener las palabras que más se asocian (están correlacionadas con) a una lista de escritores concretos

palabras\_correlacion\_todas %>%

filter(item1 %in% c("valleinclán",

"benavente",

"ortega",

"machado",

"baroja",

"jiménez")) %>%

group\_by(item1) %>%

top\_n(10) %>%

ungroup() %>%

mutate(item2 = reorder\_within(item2, correlation,item1)) %>%

ggplot(aes(item2, correlation)) +

geom\_bar(stat = "identity") +

facet\_wrap(~ item1, scales = "free") +

scale\_x\_reordered() +

coord\_flip()

palabras\_correlacion\_todas %>%

filter(item1 %in% c("marañón",

"camba",

"bergamín",

"galdós",

"giménez",

"unamuno")) %>%

group\_by(item1) %>%

top\_n(10) %>%

ungroup() %>%

mutate(item2 = reorder\_within(item2, correlation,item1)) %>%

ggplot(aes(item2, correlation)) +

geom\_bar(stat = "identity") +

facet\_wrap(~ item1, scales = "free") +

scale\_x\_reordered() +

coord\_flip()

# 9.5 – SEGUNDO ANÁLISIS CON COOCURRENCIAS

# TÉRMINOS ASOCIADOS A LA PALABRA “HOMBRE” Y A LA PALABRA “MUJER”

# 9.5.1

# Lo primero es confirmar si la palabra “hombre” y la palabra “mujer” se mencionan con suficiente frecuencia en cada una de las principales secciones

Resenas\_tabla\_sinvacias %>%

filter (palabra == "hombre" | palabra == "mujer") %>%

filter(genero == "novela y narraciones" | genero == "poesia" | genero == "dramatica" | genero == "ensayos literarios" | genero == "estudio introductorio" | genero == "literatura historica") %>%

group\_by (genero) %>%

count (palabra, sort = T) %>%

top\_n (3) %>%

ggplot(aes(reorder(palabra, n),

n,

fill = genero)) +

geom\_bar(stat = "identity") +

facet\_wrap(~genero,

scales = "free\_y") +

labs(x = "",

y = "Frecuencia absoluta") +

coord\_flip() +

theme(legend.position="none")

# 9.5.2

# Una vez confirmado que tenemos un número razonable de apariciones de estas palabras, realizamos el análisis de coocurrencias para “hombre” y “mujer” en cada una de las principales secciones

# el fragmento de código se repite

palabras\_correlacion\_dramatica %>%

filter(item1 %in% c("hombre", "mujer")) %>%

group\_by(item1) %>%

top\_n(5) %>%

ungroup() %>%

mutate(item2 = reorder(item2, correlation)) %>%

ggplot(aes(item2, correlation)) +

geom\_bar(stat = "identity") +

facet\_wrap(~ item1, scales = "free") +

coord\_flip()

palabras\_correlacion\_novela\_narraciones %>%

filter(item1 %in% c("hombre", "mujer")) %>%

group\_by(item1) %>%

top\_n(5) %>%

ungroup() %>%

mutate(item2 = reorder(item2, correlation)) %>%

ggplot(aes(item2, correlation)) +

geom\_bar(stat = "identity") +

facet\_wrap(~ item1, scales = "free") +

coord\_flip()

palabras\_correlacion\_estudio\_introductorio %>%

filter(item1 %in% c("hombre", "mujer")) %>%

group\_by(item1) %>%

top\_n(5) %>%

ungroup() %>%

mutate(item2 = reorder(item2, correlation)) %>%

ggplot(aes(item2, correlation)) +

geom\_bar(stat = "identity") +

facet\_wrap(~ item1, scales = "free") +

coord\_flip()

palabras\_correlacion\_ensayos\_literarios %>%

filter(item1 %in% c("hombre", "mujer")) %>%

group\_by(item1) %>%

top\_n(5) %>%

ungroup() %>%

mutate(item2 = reorder(item2, correlation)) %>%

ggplot(aes(item2, correlation)) +

geom\_bar(stat = "identity") +

facet\_wrap(~ item1, scales = "free") +

coord\_flip()

palabras\_correlacion\_literatura\_historica %>%

filter(item1 %in% c("hombre", "mujer")) %>%

group\_by(item1) %>%

top\_n(5) %>%

ungroup() %>%

mutate(item2 = reorder(item2, correlation)) %>%

ggplot(aes(item2, correlation)) +

geom\_bar(stat = "identity") +

facet\_wrap(~ item1, scales = "free") +

coord\_flip()

# 9.6 – TERCER ANÁLISIS CON COOCURRENCIAS

# TÉRMINOS ASOCIADOS A PALABRAS DE CARACTERÍSTICAS MÁS POLÍTICAS

# ES UN ANÁLISIS MUY PARECIDO AL REALIZADO EN 9.5

# 9.6.1 Primero comprobamos que las palabras que queremos analizar se repiten con suficiente frecuencia en las secciones de índice literario

Resenas\_tabla\_sinvacias %>%

filter (palabra == "monarquía" | palabra == "república" | palabra == "dictadura" | palabra == "pueblo" | palabra == "revolución") %>%

filter(genero == "novela y narraciones" | genero == "poesia" | genero == "dramatica" | genero == "ensayos literarios" | genero == "estudio introductorio" | genero == "literatura historica") %>%

group\_by (genero) %>%

count (palabra, sort = T) %>%

top\_n (8) %>%

ggplot(aes(reorder(palabra, n),

n,

fill = genero)) +

geom\_bar(stat = "identity") +

facet\_wrap(~genero,

scales = "free\_y") +

labs(x = "",

y = "Frecuencia absoluta") +

coord\_flip() +

theme(legend.position="none")

Resenas\_tabla\_sinvacias %>%

filter (palabra == "religión" | palabra == "dios" | palabra == "iglesia") %>%

filter(genero == "novela y narraciones" | genero == "poesia" | genero == "dramatica" | genero == "ensayos literarios" | genero == "estudio introductorio" | genero == "literatura historica") %>%

group\_by (genero) %>%

count (palabra, sort = T) %>%

top\_n (8) %>%

ggplot(aes(reorder(palabra, n),

n,

fill = genero)) +

geom\_bar(stat = "identity") +

facet\_wrap(~genero,

scales = "free\_y") +

labs(x = "",

y = "Frecuencia absoluta") +

coord\_flip() +

theme(legend.position="none")

# 9.6.1b En el caso de divorcio y matrimonio, nos fijamos en las frecuencias relativas en lugar de las frecuencias absolutas

Frecuencia\_resenas\_sinvacias <- Resenas\_tabla\_sinvacias %>%

group\_by (sexo) %>%

count (palabra, sort = T) %>%

mutate (relativa = n / sum(n)) %>%

filter(palabra == "divorcio" | palabra == "matrimonio") %>%

ungroup ()

Frecuencia\_resenas\_sinvacias %>%

ggplot(aes(reorder(palabra, relativa),

relativa,

fill = sexo)) +

geom\_bar(stat = "identity") +

facet\_wrap(~sexo,

scales = "free\_y") +

labs(x = "",

y = "Frecuencia relativa") +

coord\_flip() +

theme(legend.position="none")

# 9.6.2 Y a continuación analizamos esas palabras

palabras\_correlacion\_todas %>%

filter(item1 %in% c("república", "monarquía", "dictadura", "pueblo", "revolución")) %>%

group\_by(item1) %>%

top\_n(5) %>%

ungroup() %>%

mutate(item2 = reorder\_within(item2, correlation, item1)) %>%

ggplot(aes(item2, correlation)) +

geom\_bar(stat = "identity") +

facet\_wrap(~ item1, scales = "free") +

scale\_x\_reordered() +

coord\_flip()

palabras\_correlacion\_todas %>%

filter(item1 %in% c("religión", "dios", "iglesia")) %>% group\_by(item1) %>%

top\_n(5) %>%

ungroup() %>%

mutate(item2 = reorder\_within(item2, correlation, item1)) %>%

ggplot(aes(item2, correlation)) +

geom\_bar(stat = "identity") +

facet\_wrap(~ item1, scales = "free") +

scale\_x\_reordered() +

coord\_flip()

palabras\_correlacion\_todas %>%

filter(item1 %in% c("divorcio", "matrimonio")) %>%

group\_by(item1) %>%

top\_n(5) %>%

ungroup() %>%

mutate(item2 = reorder\_within(item2, correlation, item1)) %>%

ggplot(aes(item2, correlation)) +

geom\_bar(stat = "identity") +

facet\_wrap(~ item1, scales = "free") +

scale\_x\_reordered() +

coord\_flip()

# PASO 10 - Topic Modeling

# Trataremos de identificar algunos de los topics (conjuntos de palabras que aparecen juntas con frecuencia) presentes en este número de la revista Índice y señalar los más presentes en las reseñas de obras escritas por hombres y mujeres para establecer diferencias y en los distintos géneros / secciones

# 10.1 Cargar las librerías

library(tm)

library(topicmodels)

library(scales)

# 10.2

# Contar las palabras por reseña (sin tener en cuenta las palabras vacías) y juntar sexo, género y número de reseña en un único campo.

palabra\_conteo\_por\_resena <- Resenas\_tabla\_sinvacias %>%

unite (sexo\_genero\_numresena, sexo, genero, numresena) %>%

count(sexo\_genero\_numresena, palabra, sort = TRUE) %>%

ungroup()

# 10.3 Obtener la matriz DTM

resenas\_dtm <- palabra\_conteo\_por\_resena %>%

cast\_dtm(sexo\_genero\_numresena, palabra, n)

# 10.4 Construir el modelo LDA para obtener los Topics

# IMPORTANTE: SE OPTA POR UN MODELO DE 6 TOPICS (parece dar resultados más claros que modelos con más o menos topics)

resenas\_lda <- LDA (resenas\_dtm, k=6, control = list (seed = 1234))

resenas\_lda\_td <- tidy(resenas\_lda, matrix = "beta")

options(scipen=999)

# 10.5

# obtenemos los 8 términos más frecuentes para cada uno de los topics

terminos\_frecuentes <- resenas\_lda\_td %>%

group\_by(topic) %>%

top\_n(8, beta) %>%

ungroup() %>%

arrange(topic, -beta)

# 10.6

# pinto los términos más frecuentes de cada topic

terminos\_frecuentes %>%

mutate(term = reorder\_within(term, beta, topic)) %>%

ggplot(aes(term, beta, fill = factor(topic))) +

geom\_col(show.legend = FALSE) +

facet\_wrap(~ topic, scales = "free") +

scale\_x\_reordered() +

coord\_flip()

#10.7

# Análisis para ver cuanto tiene de cada topic cada reseña

# Para ello vuelvo a separar los campos sexo, género y número de reseña

# Estamos preparando el análisis

resenas\_lda\_gamma <- tidy(resenas\_lda, matrix = "gamma")

resenas\_lda\_gamma <- resenas\_lda\_gamma %>%

separate(document,

c("sexo", "genero", "numresena"),

sep = "\_", convert = TRUE)

#10.8 Miramos el consenso de cada reseña, es decir, el topic dominante de cada reseña, y lo pintamos

resenas\_por\_topico\_principal <- resenas\_lda\_gamma %>%

group\_by(sexo, genero, numresena) %>%

top\_n(1, gamma) %>%

ungroup() %>%

arrange(gamma)

resenas\_por\_topico\_principal %>%

filter(genero == "novela y narraciones" | genero == "poesia" | genero == "dramatica" | genero == "ensayos literarios" | genero == "estudio introductorio" | genero == "literatura historica") %>%

group\_by (genero) %>%

count (topic, sort = T) %>%

top\_n (6) %>%

ungroup () %>%

mutate(topic = reorder\_within(topic, n,genero)) %>%

ggplot(aes(topic,

n,

fill = genero)) +

geom\_bar(stat = "identity") +

facet\_wrap(~genero,

scales = "free\_y") +

labs(x = "",

y = "Frecuencia absoluta") +

coord\_flip() +

scale\_x\_reordered() +

theme(legend.position="none")

#10.9 Profundizo en el análisis anterior (tópico dominante por reseña) y me fijo solamente en reseñas de “novelas y narraciones”, diferenciando obras escritas por hombres y obras escritas por mujeres

resenas\_por\_topico\_principal %>%

filter(genero == "novela y narraciones") %>%

group\_by (sexo) %>%

count (topic, sort = T) %>%

top\_n (16) %>%

ungroup () %>%

mutate(topic = reorder\_within(topic, n, sexo)) %>%

ggplot(aes(topic,

n,

fill = sexo)) +

geom\_bar(stat = "identity") +

facet\_wrap(~sexo,

scales = "free\_y") +

labs(x = "",

y = "Frecuencia absoluta") +

coord\_flip() +

scale\_x\_reordered() +

theme(legend.position="none")

resenas\_por\_topico\_principal %>%

filter(genero == "novela y narraciones") %>%

group\_by (sexo) %>%

count (topic, sort = T) %>%

mutate (relativa = n / sum(n)) %>%

ungroup () %>%

ggplot(aes(reorder(topic,relativa),

relativa,

fill = sexo)) +

geom\_bar(stat = "identity") +

facet\_wrap(~sexo,

scales = "free\_y") +

labs(x = "",

y = "Frecuencia relativa") +

coord\_flip() +

theme(legend.position="none")

#10.10 A continuación extraigo las palabras más frecuentes de los tópicos 2 y 3, para ver si se aprecian diferencias relevantes

terminos\_frecuentes <- resenas\_lda\_td %>%

group\_by(topic) %>%

top\_n(13, beta) %>%

ungroup() %>%

arrange(topic, -beta)

terminos\_frecuentes %>%

filter (topic == "3" | topic == "1") %>%

mutate(term = reorder(term, beta)) %>%

ggplot(aes(term, beta, fill = factor(topic))) +

geom\_col(show.legend = FALSE) +

facet\_wrap(~ topic, scales = "free") +

coord\_flip()

# 10.11

# Para profundizar más en el análisis, me fijo en el porcentaje de palabras de cada reseña que van a cada tópico

# Hago el análisis diferenciando reseñas de hombre y de mujer

asignaciones <- augment(resenas\_lda, data = resenas\_dtm)

asignaciones <- asignaciones %>%

separate(document, c("sexo","genero",

"numresena"),

convert = TRUE)

asignaciones %>%

count(sexo, .topic, wt = count) %>%

group\_by(sexo) %>%

mutate(porcentaje = n / sum(n)) %>%

ggplot(aes(.topic, sexo, fill = porcentaje)) +

geom\_tile() +

scale\_fill\_gradient2(high = "blue", label = percent\_format()) +

scale\_x\_discrete() +

scale\_x\_discrete(limits=1:6) +

theme\_minimal() +

theme(axis.text.x = element\_text(angle = 90, hjust = 1),

panel.grid = element\_blank()) +

labs(x = "Temas de las reseñas (modelado de tópicos)",

y = "Género del autor reseñado",

title = "Descomposición de las reseñas por tópicos",

fill = "% de asignaciones")