Introducción al Análisis de texto

Text Mining

Daniel Jiménez M.

Universidad Nacional de Colombia

02 -11 -2020

Bibliográfia

- Ingo Feinerer, **Introduction to the tm Package**, 2019, CRAN.
- Julia Silge & David Robinson, **Text Mining with R**, 2020, O'Reilly.
- Christopher D. Manning, **Foundations os Statistical Natuale Lenguaje Processing**, 1999, MIT.
- Ted Kwartler, **Text Mining in Practice with R**, 2017, Wiley.

Requisitos para la clase

Para esta clase es necesario que tengan instaladas y funcionales las siguientes liberías:

```
library(tidyverse) # Manipulación de datos
library(tidytext) # Manipulación de textos
library(tm) #  Procesamiento de mega-data y creación de corpus
library(topicmodels) # Agrupación de textos por tema
library(wordcloud) # Nube de palabras
```

¿Qué es el Text Mining?

Es un área de la ciencia de datos que estudia los documentos de naturaleza texto (Datos no estructurados), en el cual se explora su naturaleza, patrones, contenido y sentido.

¿Qué es el Text Mining?

Los objetivos del `text mining` son :

- Comprender la naturaleza de los documentos;
- Esta comprensión se basa en patrones estadísticos;
- Con base a los elementos estadísticos se deben encontran patrones no evidentes entre los datos.

¿Qué es el Text Mining?

Para poder desarrollar Text Mining hay que comprender que este proceso se divide generalmente en tres capas:

- Análisis de documentos;
- Topic Modelling;
- Machine Learning

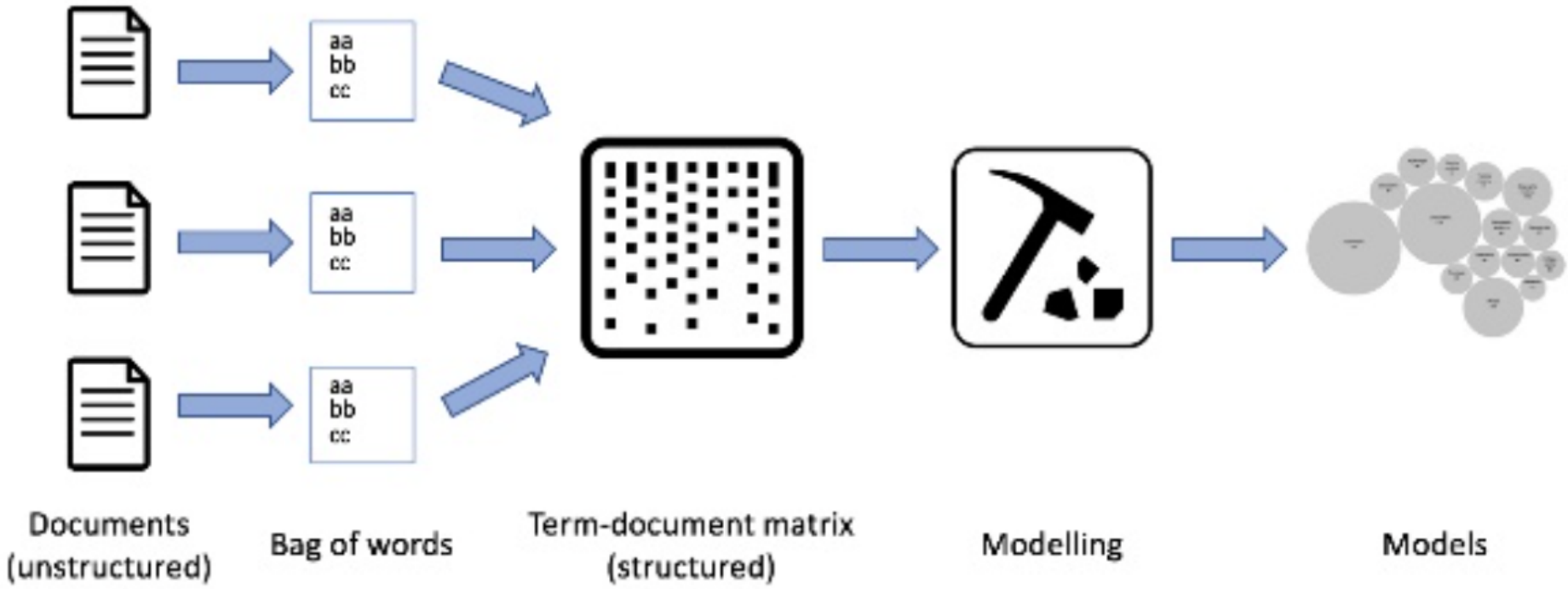
Análisis de documentos

Para entender un poco esta parte, se desarrollará un ejemplo y sobre el definiremos los pasos que se implementaron para llegar a esta conclusión.

El ejemplo será analizar la canción *Hawai* de Maluma, en el repositorio, en la sección `Bases de Datos`

Análisis de documentos

El proceso natural del `text mining` es el siguiente :



Hawai en Text Mining

Lo primero a trabajar es el cargue de la base de datos

```
hawai<-read_csv('../Bases_de_datos/hawai.txt',col_names = FALS
  rename(Letra=X1)
```

```
## # A tibble: 6 x 1
##   Letra
##   <chr>
## 1 Deja de mentirte (ah)
## 2 La foto que subiste con él diciendo que era tu cielo
## 3 Bebé
## 4 No te diré quién
## 5 Por mí te vieron
## 6 Déjame decirte
```

Hawai en Text Mining

Paso seguido hay que convertirlo en una base de datos y eso se hace asignandole un # de lineas y dandole el formato de chr a las frases.

```
hawai_df<-hawai%>%
  tibble(line=1:50,texto=Letra)
```

```
## # A tibble: 6 x 3
##   Letra                                  line texto
##   <chr>                                 <int> <chr>
## 1 Deja de mentirte (ah)                     1 Deja de mentir
## 2 La foto que subiste con él diciendo~      2 La foto que su
## 3 Bebé                                      3 Bebé
## 4 No te diré quién                          4 No te diré qui
## 5 Por mí te vieron                          5 Por mí te vier
## 6 Déjame decirte                            6 Déjame decirte
```

Contexto

Cuando trabajamos con bases de datos textuales, hay que hacer es generar un formato en el cual se pueda trabajar y es ahí cuando entra la teoría de las bases de datos, puntualmente sobre estructuras de datos.

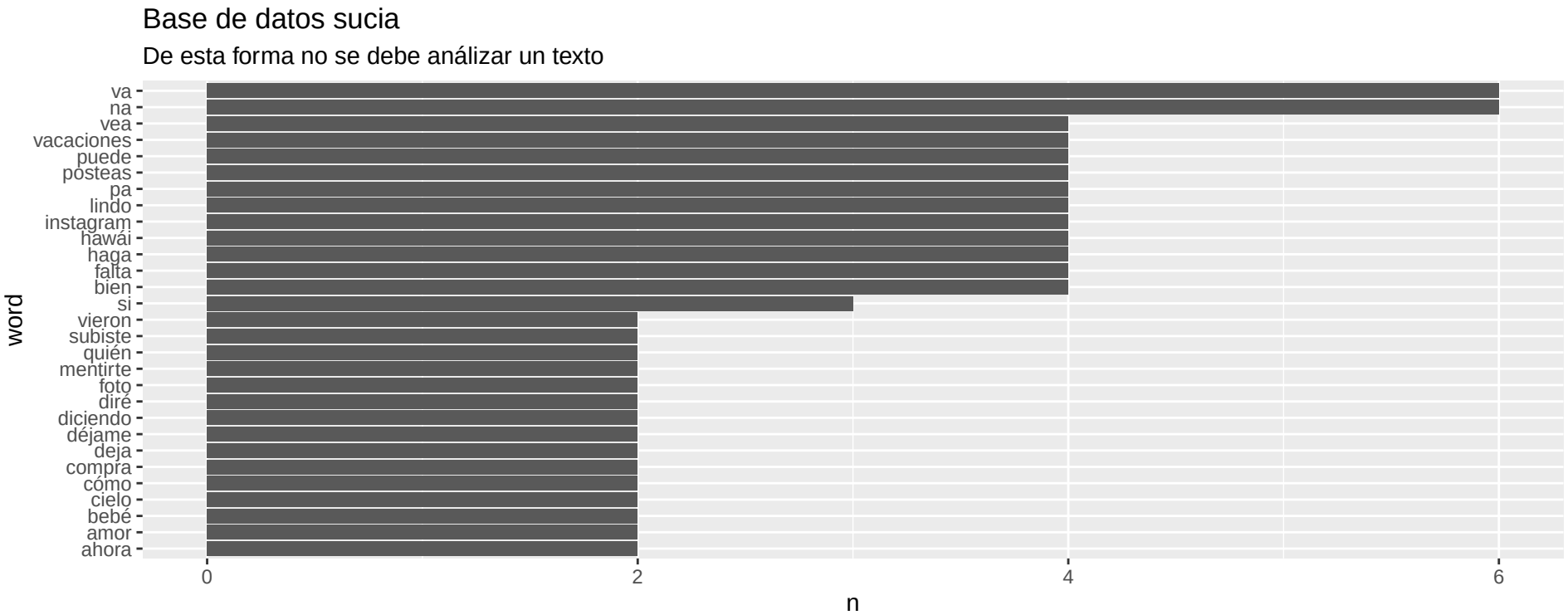
Hawai en Text Mining

Ahora se tokeniza la canción para poder a organizar la data y desarrollar el análisis

```
hawai_df%>%
  unnest_tokens(word,Letra)%>%
  filter(!word %in% stop_words_es)%>%
  count(word,sort=TRUE)
```

```
## # A tibble: 76 x 2
##    word          n
##    <chr>     <int>
##  1 na            6
##  2 va            6
##  3 bien          4
##  4 falta         4
##  5 haga          4
##  6 havái         4
##  7 instagram     4
```

Hawai en Text Mining



Contexto

En los pasos anteriores se incurrio en un proceso donde se desarrollo

- El orden de la base de datos ;
- Se genero los string de la manera como se deseaba, en este caso por palabras

Contexto

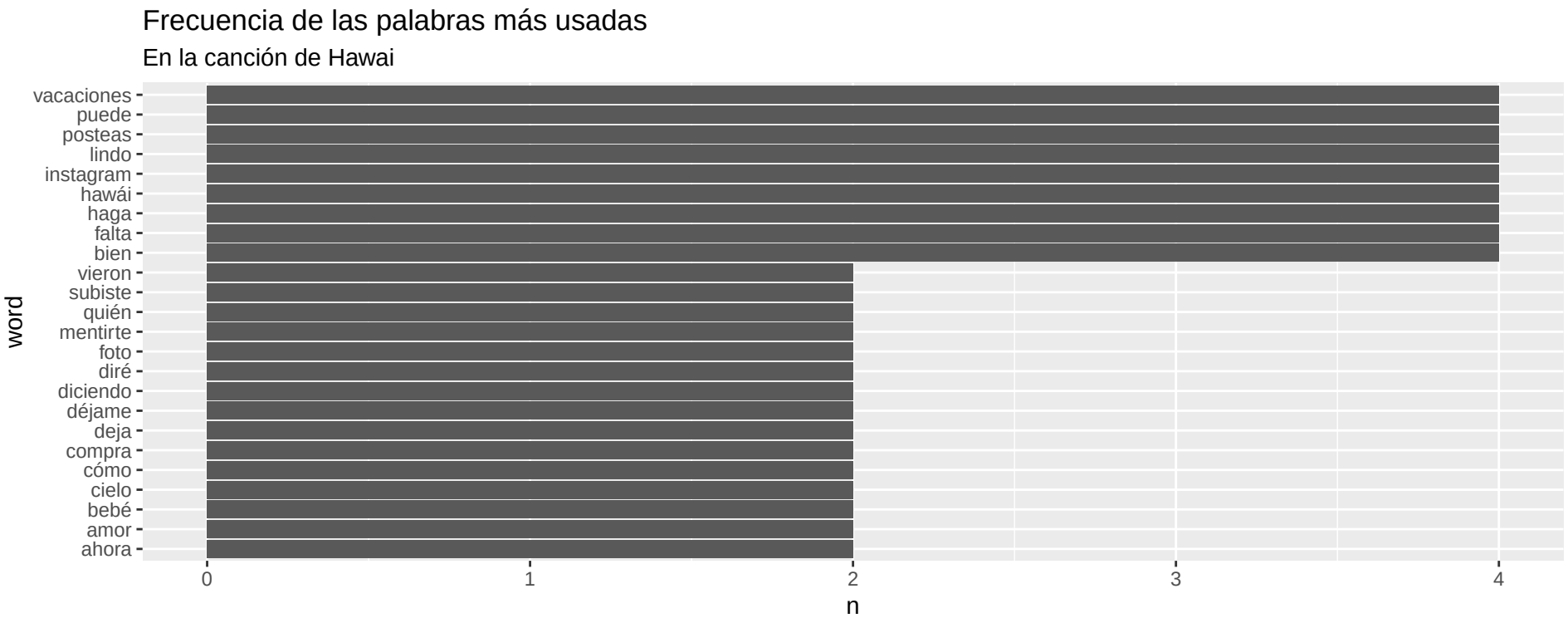
Finalmente se desarrollo un proceso de tokenizar, el cual consiste en divir el documento en un token el cual es una unidad de medida de partición que se hace al antojo, pero puede ser :

- Palabras
- Regex
- Sentencias
- Párrafos

Y de esta manera se le da una estructura de la base de datos, lo cual nos asegura que se puede empezar a trabajar con este tipo de data.

Hawai en Text Mining

Se remueven los stop words

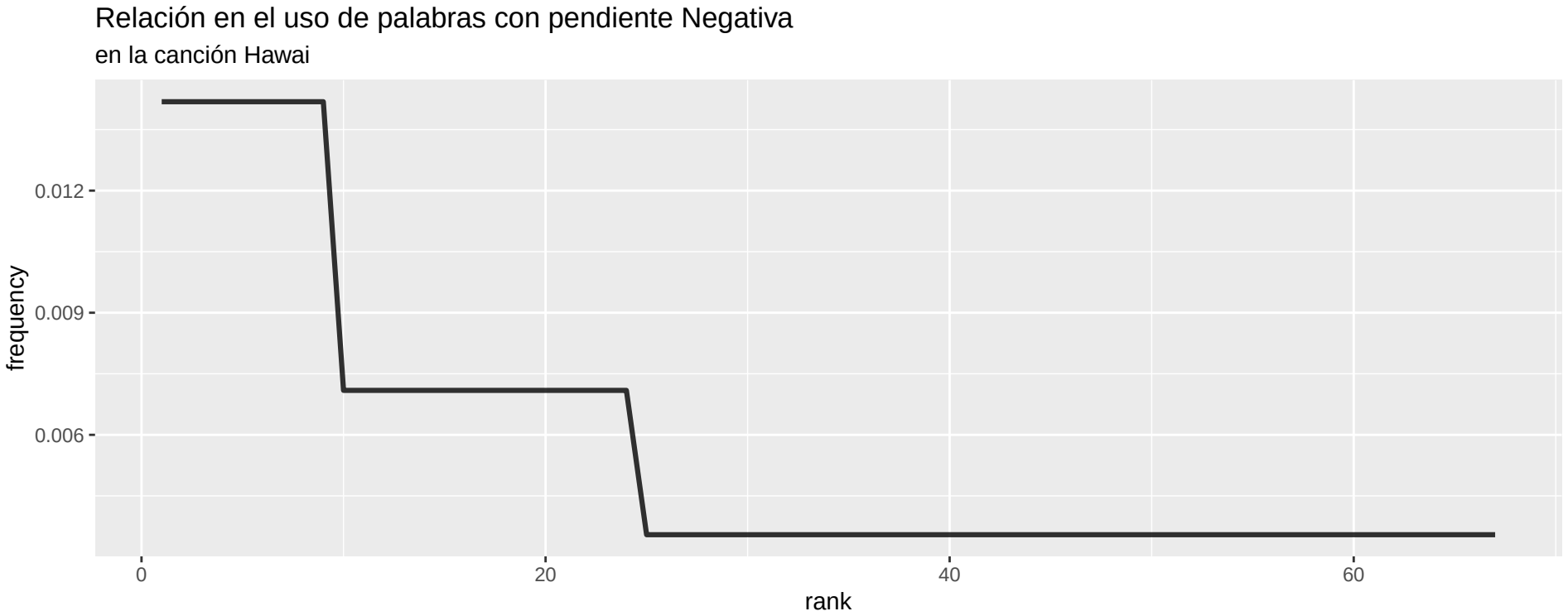


Contexto

Paso seguido se trabaja en la limpieza de los datos y es a través de los stop words, palabras que son contectores y no le dan sentido al contexto del análisis .

Hawai en Text Mining

Ahora se evalua la relación de frecuencia del uso de palabras en el contexto del párrafo a través de la ley de Zipf.



Contexto

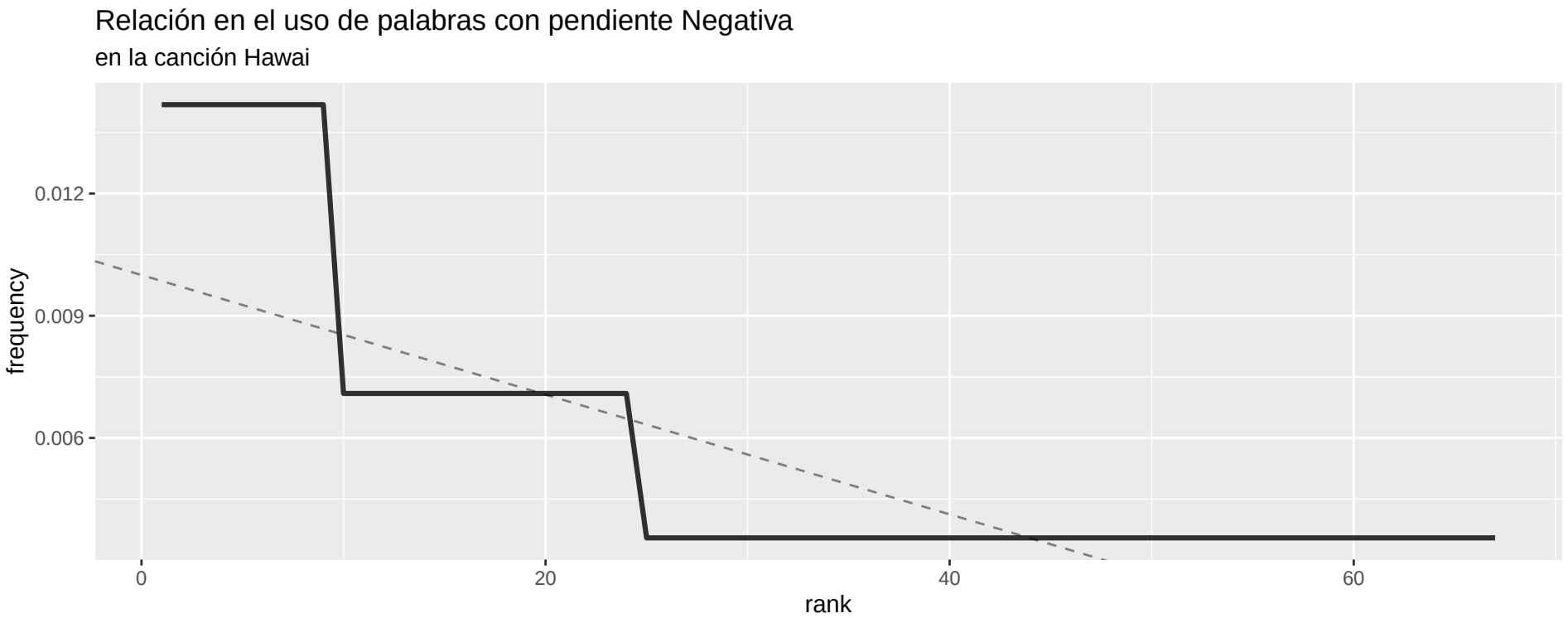
La ley de Zipf consiste en la relación matemática del uso frecuente de palabras a través de su patron de uso.

En 1940, George Zipf encontro que la palabra más usadas en un documento tiende a ser el doble de la segunda más usada y la tercera más usada un tercio de la segunda y así entendio la sucesión.

Por lo tanto la relación que encontro Zipf es la siguiente: $>$ La frecuencia de aparición de una palabra es proporcional al inversio de la prosición que ocupa.

$$\text{freq} =\infty \frac{1}{ranking}$$

Hawai en Text Mining



Gracias a esta relación se puede determinar la inversa se cumple y con ello se puede determinar lo siguiente

Hawai en Text Mining

El modelo matemático de la canción es el siguiente

```
fit<-lm(frequency~rank,data=freq_words)
```

$$\text{frequency} = 0.01 + 0(\text{rank}) + \epsilon$$

Con esto ya se tiene construido el modelo matemático de la linguistica de la canción.

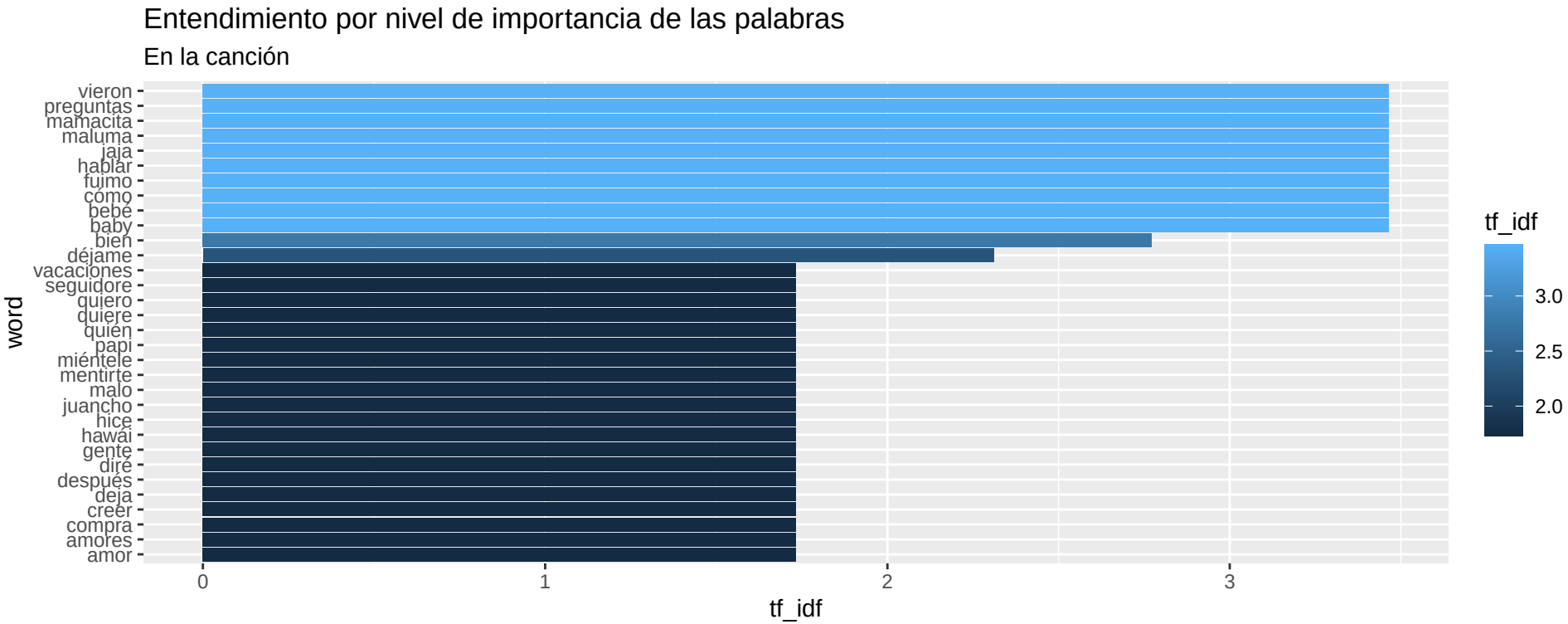
Hawai en Text Mining

Contexto

La relación anterior es el constructo matemático de la lingüística de la canción, gracias a ello podemos entender el contexto del wordcloud. Esta última visualización describe a las palabras más frecuentes dentro de un texto, pero si se le suma la ley de Zipf, el patron que se encuentra es que después de estas palabras, el resto de la canción son los conductores del contexto.

Hawai en Text Mining

Ahora dejamos de lado la frecuencia de las palabras y desarollamos un análisis en el cual se resalta la importancia de las palabras y con base a ello se le da un peso



Contexto

Para entender de que trata un texto es necesario cuantificar ciertos aspectos :

- tf : Frecuencia de la palabra;
- tf-idf: Frecuencia inversa de un termino, permite reducir el peso de las palabras que más se usan y darle peso a las que más generan contexto

Contexto

Por lo tanto

$$tf = \frac{f(t,d)}{max\{f(t,d) \in d\}}$$

Mientras que

$$idf(t,D) = log\frac{|D|}{\{d \in D : t \in d\}}$$

Donde `d` es la probabilidad de ocurrencia de una palabra dentro del texto, `D` es la cardinalidad de una palabra dentro del documento y $\{d \in D : t \in d\}$ Número de documentos donde aparece el termino `t`.

Hawai en Text Mining

Ahora se trabaja en el desarrollo de n-grams

```
hawai_ngram<-hawai_df%>%
  unnest_tokens(ngram,Letra,token = 'ngrams',n=2)

hawai_ngram%>%
  count(ngram,sort=TRUE)%>%
  filter(!is.na(ngram))
```

Hawai en Text Mining

```
## # A tibble: 4 x 2
##   ngram              n
##   <chr>          <int>
## 1 no te              7
## 2 que yo             5
## 3 te va              5
## 4 de vacaciones      4
```

Hawai en Text Mining

Ahora se separan los n-grams para poder preparar un análisis

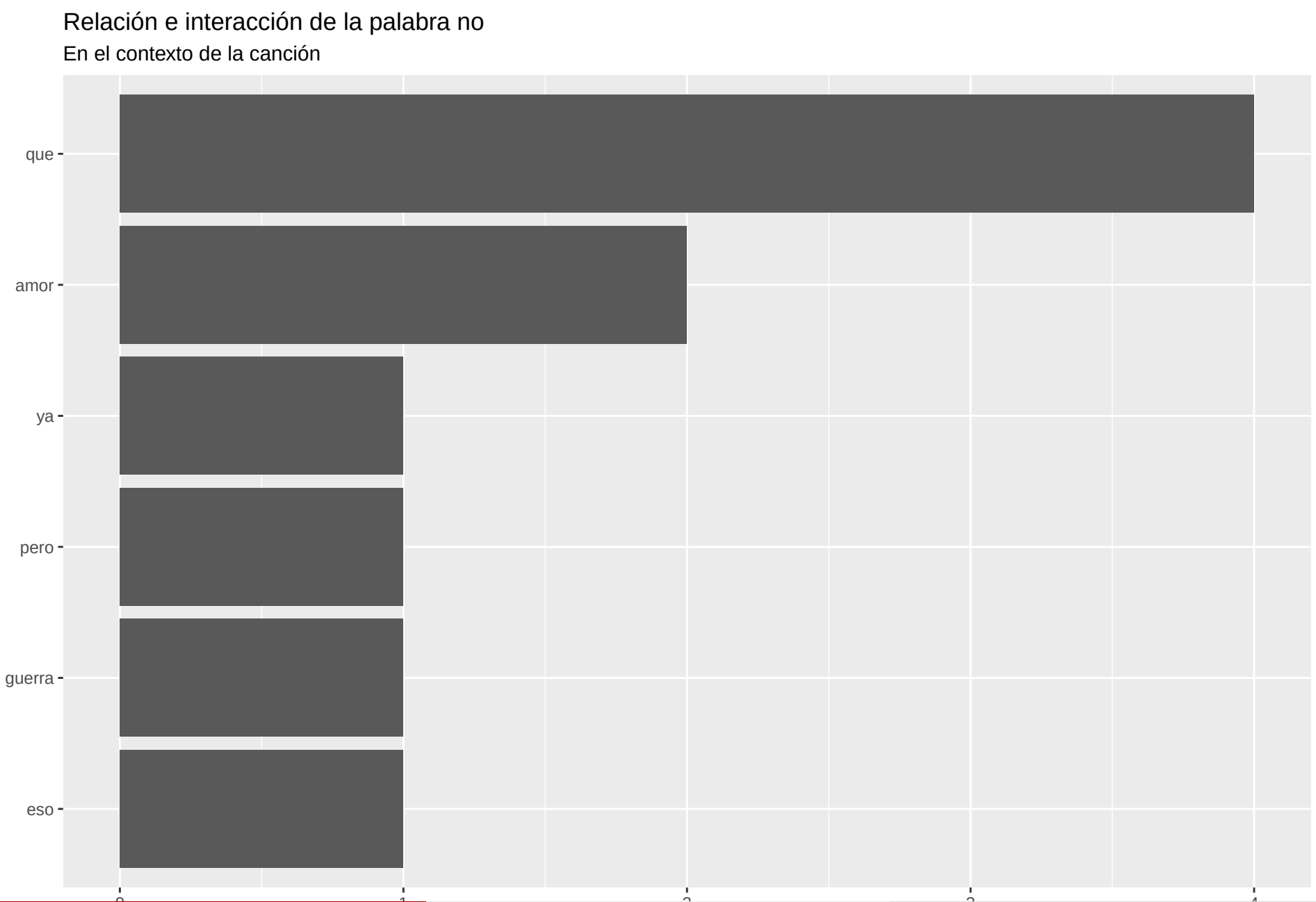
```
## # A tibble: 6 x 3
##   word1  word2        n
##   <chr>  <chr>    <int>
## 1 haga   falta        4
## 2 diré   quién        2
## 3 vea    cómo         2
## 4 déjame decirte      1
## 5 déjame hablar       1
## 6 gana   ninguno      1
```

Hawai en Text Mining

Al separar el bigram podemos hacer el conteo y de paso con ello la unión se puede encontrar un nuevo patron.

```
## # A tibble: 6 x 2
##    line bigram
##   <int> <chr>
## 1     1 mentirte ah
## 2     4 diré quién
## 3     6 déjame decirte
## 4     7 trata bien
## 5     8 llegué primero
## 6     9 va ir
```

Hawai en Text Mining



Contexto

Con los bigrams podemos decontruir una oración y con base a ello poder comprender un poco más el contexto de lo que estamos evaluando.

Hawai en Text Mining

Para profundizar sobre el entendimiento de la canción se trabaja sobre un análisis de redes.

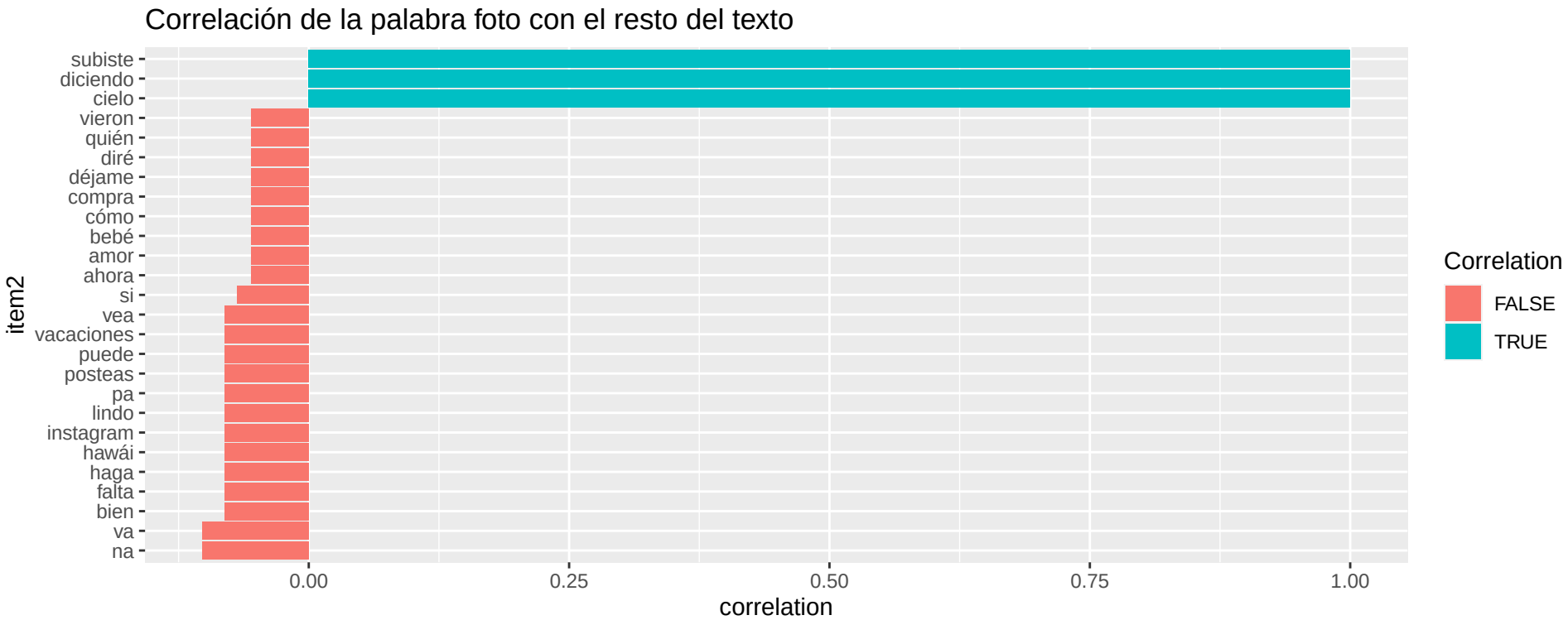
```
## # A tibble: 6 x 3
##   item1   item2    correlation
##   <chr>   <chr>          <dbl>
## 1 deja    mentirte           1.
## 2 foto    subiste            1.
## 3 foto    diciendo           1.
## 4 subiste diciendo           1.
## 5 foto    cielo              1.
## 6 subiste cielo              1.
```

Contexto

Para comprender el contexto de la canción es necesario trabajar una capa matemática más profunda y para ello sera necesario llamar a la librería `widyr` con la cual se podrá hacer de manera ordenada permutaciones entre las palabras y con base a ello hacer correlaciones.

Hawai en Text Mining

Suponga que quiere entender en que contexto se uso la palabra `foto`



Contexto

Si se da cuenta en este contexto lo rojo significa algo inversamente proporcional, eso quiere decir que cada vez que se usan esas palabras, pierde frecuencia o sentido la palabra `foto`, mientras lo verde es algo que genera contexto con la misma palabra.

El paso anterior es el desarrollo de un `PCA`, que en si consite en la creación de una pseudo-variable que almacena la mayor cantidad de información **varianza** en el contexto de los datos.

Hawai en Text Mining

Ahora hagamos un análisis de redes para poder crear un contexto de la canción.

```
words_count<-hawai_words%>%
  count(word,sort = TRUE)

hawai_words%>%
  pairwise_cor(word,line,sort=TRUE)%>%
  head(500)%>%
  as_tbl_graph()%>%
  left_join(words_count,by=c(name='word'))%>%
  ggraph(layout = 'fr')+
  geom_edge_link(aes(edge_alpha=correlation))+
  geom_node_point(aes(size=n))+
  geom_node_text(aes(label=name),
                 check_overlap=TRUE,
                 vjust=2,
                 hjust=1
```

Hawai en Text Mining

Esta parte es un bonus! y es que usaremos una técnica con algunas modificaciones para crear un tag sobre la canción.

```
library(stm)
hawai_matrix<-hawai_words%>%
  group_by(word)%>%
  filter(total_words>1)%>%
  cast_sparse(line, word, total_words)

topic_model_1 <- stm(hawai_matrix,
                     K = 2,
                     verbose = TRUE,
                     init.type = "Spectral",
                     emtol = 5)
```

```
## Beginning Spectral Initialization
##   Calculating the gram matrix...
##   Finding anchor words...
```

Hawai en Text Mining

Ahora veremos las palabras que más representan en contexto a la canción.

Hawai en Text Mining

Construimos un corpus del documento

```
hawai_dtm<-hawai_words%>%
  group_by(word)%>%
  filter(total_words>1)%>%
  cast_dtm(line, word, total_words)
hawai_dtm
```

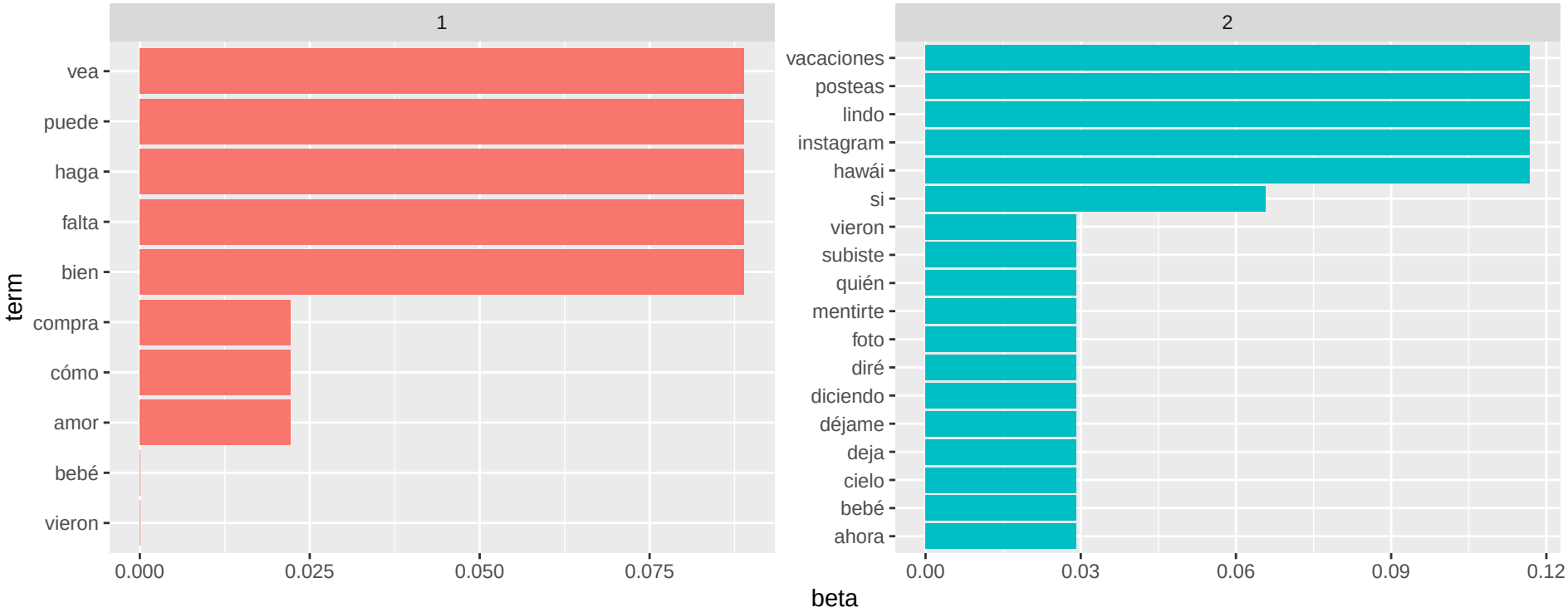
```
## <<DocumentTermMatrix (documents: 38, terms: 29)>>
## Non-/sparse entries: 89/1013
## Sparsity           : 92%
## Maximal term length: 10
## Weighting          : term frequency (tf)
```

Contexto

Corpus es la forma de desarrollar en un solo documento mega-data.

Hawai en Text Mining

Ahora implementaremos un LDA para poder generar topics en la canción.



Contexto

Latent Dirichlet allocation, es una técnica que permite agrupar los datos en grupos en grupos que tienen una explicación no observable, en este caso tópicos.

Hawai en Text Mining

Evaluamos los topics candidatos

```
## # A tibble: 2 x 4
##   term      topic1 topic2 log_ratio
##   <chr>      <dbl>  <dbl>     <dbl>
## 1 bebé   8.19e-276 0.0292      909.
## 2 vieron 7.78e-277 0.0292      912.
```

Hawai en Text Mining

Y ahora tenemos el tagging gracias a beta y gamma

```
## # A tibble: 1 x 5
## # Groups:   topic [1]
##   document topic gamma total_words tagging
##   <chr>    <int> <dbl>       <int> <chr>
## 1 10           1 0.999           4 haga falta
```

Contexto

- Beta en text mining : La probabilidad de que una palabra pertenezca a un topic
- Gamma en text mining : La probabilidad de cada documento por tema.