Servicio de vigilancia y detección de insights y oportunidades para la oferta exportable de alimentos con soporte en la gastronomía y tendencias post-COVID19'

Lima, Enero 2021

Perú

TEXT MINING & SENTIMENT ANALYSIS

Promperú - 2021


1. Software utilizado
2. Modelos analíticos implementados
3. Metodología
4. Proceso de análisis de Insights

SOFTWARE

CARACTERÍSTICAS

1. **Procesador: 3.2 GHz Core I7**
2. **Memoria RAM: 32 GB**
3. **R version: 3.6.2**
4. **R studio version: 1.3**

Razones para usar R:

1. R es gratuito
2. Es intuitivo
3. Esta avalado por una comunidad científica mundial.
4. Constantemente aparecen nuevos paquetes gratuitos que expande la capacidad de r
5. Crea gráficos de calidad superior a otros paquetes
6. Es compatible con 'todos' los formatos de datos

Razones para usar Power BI:

1. Es intuitivo
2. Compatible con el modelo de datos actual
3. Control y seguridad de información
4. Actualizaciones periódicas
5. Múltiple visibilidad (PC, móvil)


1. Software utilizado
2. Modelos analíticos implementados
3. Metodología
4. Proceso de análisis de Insights

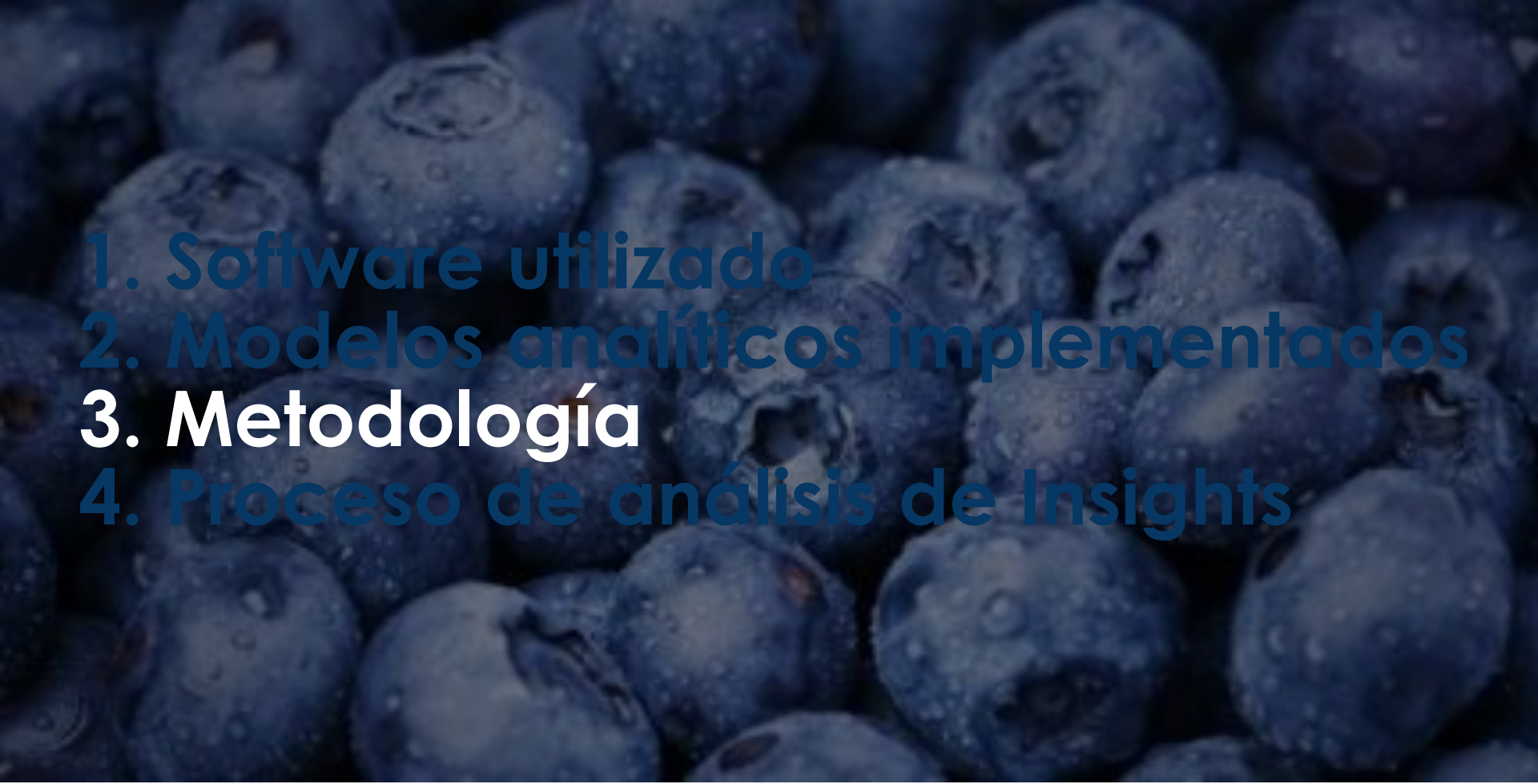
1. Evolución de las menciones en las semanas de estudio
   Las líneas de tiempo que permiten ver la estacionalidad del producto en el periodo de estudio

2. Keyword Cloud/ frequency
   Una nube de palabras que permite ver cuáles se citan más, qué otras palabras acompañan, etc.

3. Análisis de bigramas
   Grafo de palabras que se combinan de dos en dos para dar significado a los mensajes.

4. Topic Modeling
   Cluster de términos, para ver cómo se agrupan, qué resúmenes se podrían hacer, etc.

5. Sentiment Analysis
   Análisis del sentimiento, tanto positivo y negativos, así como por emociones básicas de la persona (alegría, tristeza, miedo, rechazo, etc.).


1. Software utilizado
2. Modelos analíticos implementados
3. Metodología
4. Proceso de análisis de Insights

Proceso Promperú

Análisis y Exploración

- Construcción de matriz de datos
- Filtro de negocio.
- Estudio de variables

Limpieza de datos y tokenización

- Limpieza de textos.
- Filtro de stopwords
- Tokenización
- Lematizado de palabras

Se realiza la limpieza de la data conservando textos relevantes para el estudio

Construcción de marco de datos Tidy

- Construcción de marco de datos considerando palabras tokenizadas.

Modelamiento

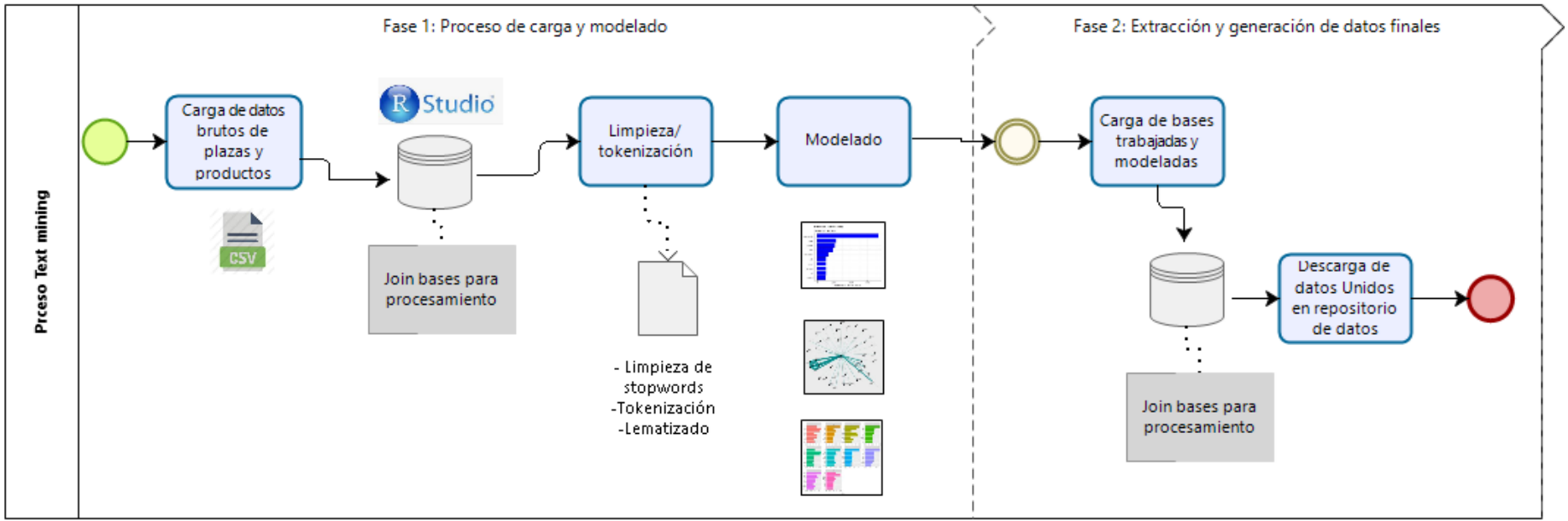
- Análisis de tendencias
- Frecuencia de palabras mas recurrentes
- Análisis clustering
- Análisis de bigramas
- Análisis de sentimientos
- Generación de tablas de consumo

Visualización de datos

- Carga de tablas de consumo.
- Creación de variables adicionales.
- Actualización y publicación de reporte.

Retroalimentación

Proceso Promperú



1. Software utilizado
2. Modelos analíticos implementados
3. Metodología
4. Proceso de análisis de Insights

A. Diccionario de variables requeridas

Variable	Descripción
url	La URL donde está el comentario de texto.
sentimiento	Obtenido por la herramienta de escucha (quizás nos aporte algo, lo he dejado por si las moscas, aunque prefiero hacerlo a partir del texto disponible)
autor	Quién lo escribió, por si quisieran localizar algún blog o identidad de redes sociales de referencia (influencia)
texto	El contenido de texto en sí. El campos más relevante, lógicamente.
hashtags	Quizás sea interesante sacar cuáles son los más citados, para ver tendencias, modas, etc.
impacto	Cuánto gente ha reaccionado al texto (mide a ver el éxito de una publicación)
impresiones	Cuánto gente lo ha llegado a leer (es lógicamente una cifra mayor al impacto)
localización	Desde dónde se ha escrito en el mercado de referencia, por si quisiera analizar también por ubicaciones.
dia	Día del periodo de estudio (octubre a diciembre de 2020)
hora	hora
plaza	País en análisis
producto	Producto en análisis

A. Preparando el entorno de trabajo

Limpieza del entorno de trabajo

```
# clean the workspace
rm(list = ls())
cat("\014")
# fijamos a UTF-8
options(encoding = "utf-8")
```

Carga de librerías necesarias para trabajar

```
# Cargamos las librerías  que vamos a necesitar
library(readxl) #  Para leer ficheros excel
library(tidyverse) #  Para las operaciones con datos
library(syuzhet) #  Libreria para emociones
library(tidytext)
library(stringr) #  Para operar con datos de tipo St
library(stopwords) #  Para poder quitar las stopword
library(ggplot2) #  Librería de visualización gráfic
library(lubridate) #  Para el formateo de fechas y s
library(scales) # Para trabajar con datos de coma
```

Carga de funciones diseñadas

Functions	
calcular…	function (dtm)
cvLDA	function (Ntopics, dtm, …

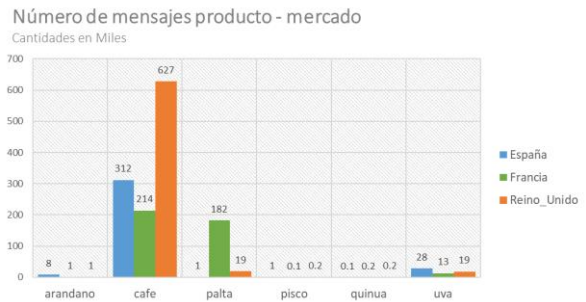
B. ANÁLISIS Y EXPLORACIÓN

Carga y concatenación de datos según productos y plazas

```
206 arandano_esp <- read.csv("2. Datos brutos - Data 2020/productos/arandano_esp.csv",encoding = "Latin1",sep = ";")
207 arandano_fr <- read.csv("2. Datos brutos - Data 2020/productos/arandano_fr.csv",encoding = "Latin1",sep = ";")
208 arandano_uk <- read.csv("2. Datos brutos - Data 2020/productos/arandano_uk.csv",encoding = "Latin1",sep = ";")
```

Construcción de variables

```
210 # Creando variables plaza y  producto
211 arandano_esp$plaza <- "Espana"
212 arandano_fr$plaza <- "Francia"
213 arandano_uk$plaza <- "Reino_Unido"
214 arandano <- rbind(arandano_esp,arandano_fr,arandano_uk)
215
216 rm(arandano_esp,arandano_fr,arandano_uk)
217 arandano$producto <- "arandano"
218
219 names(arandano) <- c("X","url","sentimiento","autor","texto","hashtags","impacto",
220                      "impresiones","localizacion","fecha","hora","plaza","producto")
```



Joins bases

```
390 dfpromperu2020 <- rbind(arandano,cafe,palta,pisco,quinua,uva,superfood)
391 dfpromperu2020$X <- NULL
392 names(dfpromperu2020)
```

C. LIMPIEZA DE DATOS

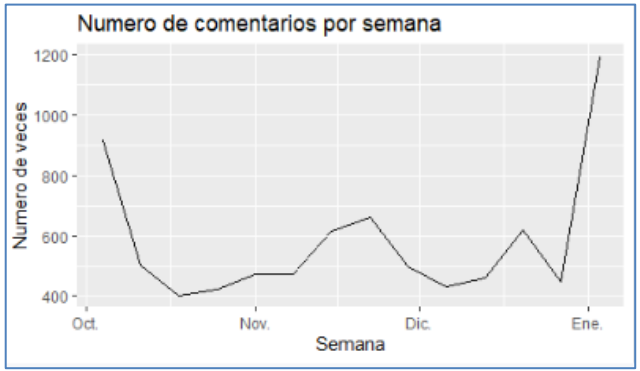
1 *Seleccionar el producto y plaza que se quiere analizar*

```
# corremos considerando base genérica
# ------------------------------------
# Carga de datos para un producto en específico (todos los paises)
datos <- arandano
# datos <- cafe
# datos <- palta
# datos <- pisco
# datos <- quinua
# datos <- uva
# datos <- superfood

# Filtro por plaza (país específico)
# ------------------------------------
# ------------------------------------
datos <- datos %>% filter(plaza == 'Espana')
# datos <- datos %>% filter(plaza == 'Francia')
# datos <- datos %>% filter(plaza == 'Reino_Unido')

table(datos$producto,datos$plaza)
# ------------------------------------
```

2 *Limpieza de textos*

```
# Vamos a hacer un poco de limpieza de texto
# ------------------------------------------
# espana$texto<-gsub("#[A-Za-z0-9]+|@[A-Za-z0-9]+|\\w+(?:\\.\\w+)*/\\S+", "",
datos$texto<-gsub("@[A-Za-z0-9]+|\\w+(?:\\.\\w+)*/\\S+", "", datos$texto)
datos$texto <- gsub("RT", "", datos$texto)
datos$texto <- gsub("https://", "", datos$texto)
datos$texto <- gsub("http://", "", datos$texto)

datos$texto <- tolower(datos$texto) # Se convierte todo el texto a min?sculas
datos$texto <- str_replace_all(datos$texto,"http\\S*", "") # Eliminaci?n de p
datos$texto <- str_replace_all(datos$texto,"[[:punct:]]", " ") # Eliminaci?n
datos$texto <- str_replace_all(datos$texto,"[[:digit:]]", " ") # Eliminaci?n
datos$texto <- str_replace_all(datos$texto,"[\\s]+", " ") # Eliminaci?n de es
datos$texto <- chartr('áéíóúñ','aeioun', datos$texto)
```

(1) Evolución de las menciones en las semanas de estudio



E. MODELAMIENTO DE DATOS

(2) Keyword Cloud/ Frequency

```
661 # Representacion grafica de las frecuencias
662 df_tidy %>% group_by(token) %>% summarise(n1 = n()) %>%
663   arrange(desc(n1)) %>% slice_max(order_by = n1, n = 10) %>%
664   ggplot(aes(x = reorder(token,n1), y = n1)) +
665   geom_col(show.legend = TRUE,fill = "blue") +
666   theme_bw() +
667   labs(y = "", x = "") +
668   theme(legend.position = "none") +
669   coord_flip() +
670   labs(title = "Palabras mas usadas",
671        subtitle = "Stopwords retiradas",
672        x = "Palabra",
673        y = "Número de veces usada")
```

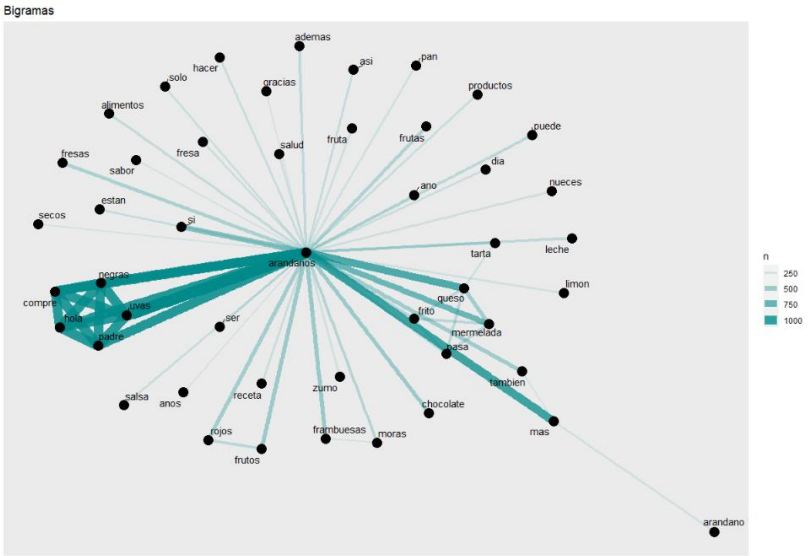
Backup de base de datos

```
677 # Para guardar la bases
678 # ---------------------
679 df_backup <- df_tidy %>% group_by(token) %>% summarise(n1 = n()) %>%
680   arrange(desc(n1)) %>% slice_max(order_by = n1, n = 10)
681
682 df_backup$producto <- "arandano"
683 df_backup$plaza <- "españa"
684
685 df_backup <- df_backup[c(3,4,1,2)]
686 write.csv(df_backup,'3. Datos Procesados - 2020/df_tidy(uva_UK).csv',row.names = F)
```

E. MODELAMIENTO DE DATOS

(3) Análisis de bigramas

```
# A veces nos interesa entender la relacion entre palabras en una opini?n.
review_bigrams <- df_bigrams %>%
  unnest_tokens(bigram, frase, token = "ngrams", n = 2) # separamos token 2 - grams
bigrams_separated <- review_bigrams %>%
  separate(bigram, c("word1", "word2"), sep = " ") # separamos word por bigrama
bigrams_filtered <- bigrams_separated %>%
  filter(!word1 %in% lexiconSW$word) %>%
  filter(!word2 %in% lexiconSW$word) # eliminamos  stop words por bigrama
bigram_counts <- bigrams_filtered %>%
  dplyr::count(word1, word2, sort = TRUE) # contamos la cantidad de words por bigra
bigrams_united <- bigrams_filtered %>%
  unite(bigram, word1, word2, sep = " ") # count bigrams cleaning
bigrams_united %>%
  dplyr::count(bigram, sort = TRUE)
```



Backup de base de datos

```
analisis_bigrams <- title_word_pairs %>% filter(n >= 500)

analisis_bigrams$producto <- "arandano"
analisis_bigrams$plaza <- "España"

analisis_bigrams <- analisis_bigrams[c(4,5,1,2,3)]
write.csv(analisis_bigrams,'3. Datos Procesados - 2020/analisis_bigrams(uva_uk).csv
```

E. MODELAMIENTO DE DATOS

(4) Topic Modeling

```
849 corpus <- corpus(datos$texto)
850 cdfm <- dfm(corpus, remove=c(lista_stopwords),
851             verbose=TRUE, remove_punct=TRUE, remove_numbers=TRUE)
852
853 # Quitamos palabras que solo salgan 1 vez
854 cdfm <- dfm_trim(cdfm, min_docfreq = 2, verbose=TRUE)
855
856 # Ahora lo exportamos a un formato para procesar los Topic Models.
857 dtm <- convert(cdfm, to="topicmodels")
858
859 # Calculamos ahora los topcis óptimos
860 # calcular topics(dtm)
861 # Estimamos el LDA con el número óptimo de topics que nos haya salido
862 lda <- LDA(dtm, k = 10, method = "Gibbs",
863            control = list(verbose=25L, seed = 123, burnin = 100, iter = 500))
```

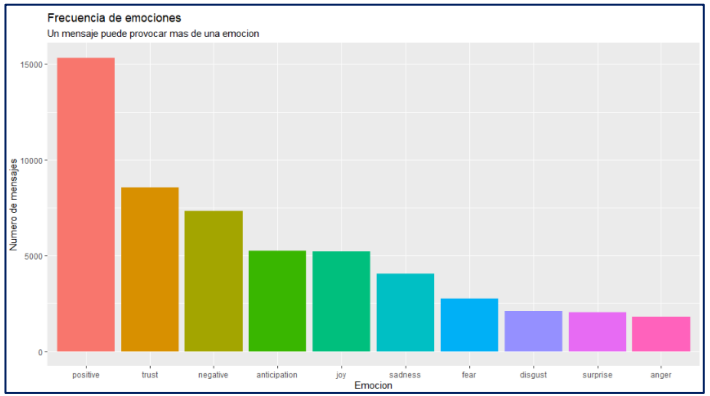
Backup de base de datos

```
889 terminos$producto <- "arandano"
890 terminos$plaza <- "españa"
891 terminos <- terminos[c(4,5,1,2,3)]
892
893 write.csv(terminos,'3. Datos Procesados - 2020/cluster_terminos(uva_uk).csv',row.names = F)
```

E. MODELAMIENTO DE DATOS

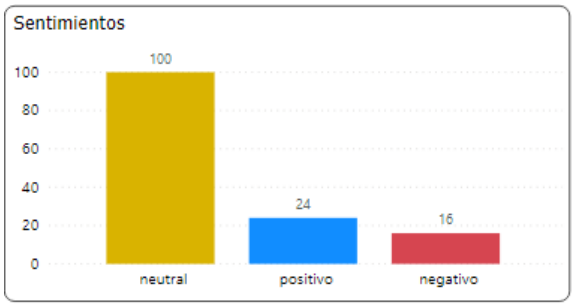
(4) Topic Modeling

```
920 # Vamos a coger las palabras en español del diccionario NRC
921 nrc<- get_sentiment(datos$texto, method="nrc",lang="spanish")
922 #nrc<- get_sentiment(datos$texto, method="nrc",lang="french")
923 #nrc<- get_sentiment(datos$texto, method="nrc",lang="english")
924
925 # Obtenemos las emociones
926 emotions <- get_nrc_sentiment(datos$texto,lang="spanish")
927 #emotions <- get_nrc_sentiment(datos$texto,lang="french")
928 #emotions <- get_nrc_sentiment(datos$texto,lang="english")
929
930 emo_bar = colSums(emotions)
```



Backup de base de datos

```
949 emo_sum$producto <- "quinua"
950 emo_sum$plaza <- "Reino_Unido"
951
952 emo_sum <- emo_sum[c(3,4,1,2)]
953
954 write.csv(emo_sum,'3. Datos Procesados - 2020/sentimientos (quinua_uk).csv',row.names = F)
```



Servicio de vigilancia y detección de insights y oportunidades para la oferta exportable de alimentos con soporte en la gastronomía y tendencias post-COVID19'

Lima, Enero 2021

Perú