

Cours 5 : Analyse de Texte

Nolan

November 8, 2024

Fonction	Usage	Exemple/Résultat
<code>read_csv("file.csv")</code>	Lit un fichier CSV et le convertit en DataFrame.	<code>review_data <- read_csv("Roomba Reviews.csv")</code>
<code>filter(data, condition)</code>	Filtre les lignes en fonction d'une condition.	<code>filter(review_data, Product == "iRobot Roomba 650 for Pets")</code>
<code>group_by(data, column)</code>	Regroupe les données par une colonne spécifiée.	<code>review_data %>% group_by(Product)</code>
<code>summarize(data, new_col = fun(col))</code>	Résume les données groupées, e.g., calcul de moyennes.	<code>summarize(stars_mean = mean(Stars))</code>
<code>count(data, col)</code>	Compte les occurrences de chaque valeur d'une colonne.	<code>count(review_data, Product)</code>
<code>unnest_tokens(word, col)</code>	Tokenise le texte en mots.	<code>unnest_tokens(word, Review)</code>
<code>anti_join(data, stop_words)</code>	Supprime les mots vides (stop words) du texte.	<code>anti_join(tidy_review, stop_words)</code>
<code>mutate(data, new_col = fun())</code>	Ajoute une colonne ou modifie une colonne existante.	<code>mutate(id = row_number())</code>
<code>arrange(data, desc(col))</code>	Trie les données en ordre décroissant.	<code>arrange(desc(n))</code> pour trier par nombre de mots.
<code>ggplot(data) + aes(x, y)</code>	Configure un graphique ggplot avec des esthétiques spécifiques.	<code>ggplot(word_counts, aes(x = word, y = n)) + geom_col()</code>
<code>geom_col()</code>	Ajoute un graphique en barres.	<code>+ geom_col()</code> pour visualiser les comptages de mots
<code>coord_flip()</code>	Inverse les axes X et Y pour un affichage horizontal.	<code>+ coord_flip()</code> pour améliorer la lisibilité des étiquettes
<code>fct_reorder(col, var)</code>	Réorganise les niveaux d'un facteur selon une variable.	<code>mutate(word2 = fct_reorder(word, n))</code>
<code>facet_wrap(~col)</code>	Crée des facettes pour afficher des sous-ensembles de données.	<code>facet_wrap(~Product, scales="free")</code>
<code>tribble(~col1, ~col2, ...)</code>	Crée un tibble rapide.	<code>tribble(~word, ~lexicon, "roomba", "CUSTOM")</code>
<code>bind_rows(data1, data2)</code>	Fusionne des données par ligne.	<code>stop_words2 <- bind_rows(stop_words, custom_stop_words)</code>
<code>inner_join(data1, data2)</code>	Joint les données avec des correspondances entre les colonnes spécifiées.	<code>inner_join(tidy_review, get_sentiments("bing"))</code>
<code>get_sentiments("bing")</code>	Charge un dictionnaire de sentiments.	<code>get_sentiments("bing")</code> pour ajouter des sentiments positifs/négatifs
<code>spread(col1, col2)</code>	Réorganise les données en colonnes distinctes.	<code>spread(sentiment, n)</code> pour répartir les sentiments
<code>mutate(overall_sentiment = positive - negative)</code>	Crée une colonne pour le sentiment global.	<code>mutate(overall_sentiment = positive - negative)</code>
<code>wordcloud()</code>	Crée un nuage de mots pour visualiser la fréquence des mots.	<code>wordcloud(words = word_counts\$word, freq = word_counts\$n)</code>