Analysis of Kakuzi Annual Report

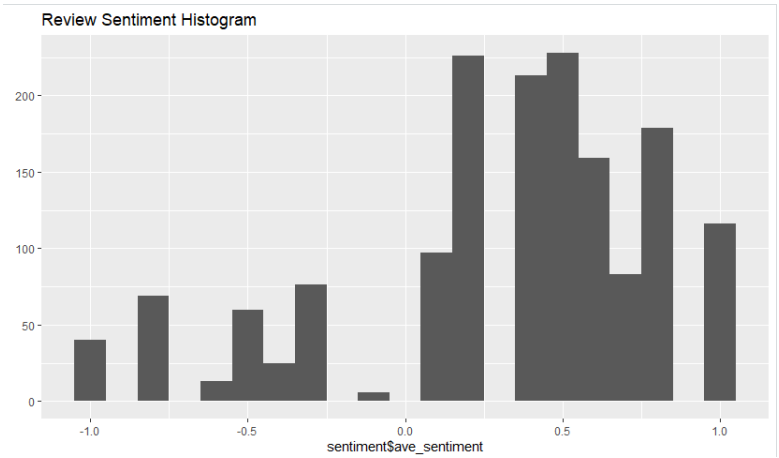
David Mathea Mithiga
Adm: 034915

Executive Summary

This report provides a comprehensive analysis of the Kakuzi Annual Report using sentiment analysis and word cloud visualization. The key takeaways from this analysis include:

1. **Word Cloud Visualization** : A word cloud was generated to highlight the most frequently occurring words in the comments, providing insights into the primary themes discussed in the report.

2. **Sentiment Analysis Summary** : Sentiment scores were calculated for each comment, enabling classification into positive, negative, or neutral categories. This helps in understanding the overall tone and mood of the report.



3. **Key Metrics** :
    a) Total number of comments analyzed.
    b) Frequency distribution of sentiments (positive, negative, neutral).
    c) Most frequently mentioned words and their relevance to the report.

The findings indicate that the majority of comments are positive, with some areas requiring attention based on negative sentiment.

1. Setup and Installation

Before performing any analysis, we installed and loaded several R packages necessary for text mining, sentiment analysis, and visualization. These packages include **tidytext**, **SentimentAnalysis**, **wordcloud**, and **RColorBrewer**. Each package serves a specific purpose:

- **tidytext**: Facilitates text processing and tokenization.
- **SentimentAnalysis**: Provides tools for calculating sentiment scores.
- **wordcloud**: Generates visual representations of word frequencies.
- **RColorBrewer**: Offers color palettes for enhancing visualizations.

2. Data Import and Exploration

We imported the dataset containing comments from the Kakuzi Annual Report and performed initial exploratory data analysis (EDA) to understand its structure and contents.

- The dataset was imported using the **read_excel** function, ensuring compatibility with Excel files. The original .CSV data could not be loaded with read_csv. To resolved, the data was converted to .xlsx with excel, and uploaded to r using the package read_excel.
- A new column **comment_count** was added to track the number of comments.
- The **str**, **summary**, and **class** functions provided insights into the dataset's structure, helping identify potential issues such as missing values or incorrect data types.

```
CBL00 <- read_excel("C:\\Users\\Administrator\\OneDrive\\Documents\\Strathmore\\Principles-of-Data-Science\\Kakuzi\\KakuziAnnual.xlsx")
View(CBL00)
CBL01 <- CBL00
View(CBL01)
```

3. Tokenization and Sentence Splitting

To analyze the comments, we tokenized the text into individual words and split sentences for sentiment analysis.

```
# Separate out words with sentences
CBL02b <- CBL01 %>%
```

	No.	QnNo	Question	CommentText	Response
1	1	2021	Annual Report	KAKUZI PLC ANNUAL REPORT AND AUDITED CONSOLIDATE...	32759
2	2	2020	Annual Report	1 KAKUZI PLC ANNUAL REPORT AND AUDITED CONSOLIDA...	32759
3	3	2019	Annual Report	1 KAKUZI PLC ANNUAL REPORT AND AUDITED CONSOLIDA...	32759

```
select(CommentText) %>%
  mutate(sentence_id = row_number()) %>%
  unnest_tokens(word, CommentText)

View(CBL02b)
```

	sentence_id	word
1	1	kakuzi
2	1	plc
3	1	annual
4	1	report
5	1	and
6	1	audited
7	1	consolidated
8	1	and
9	1	separate
10	1	financial

- The **unnest_tokens** function split the comments into individual words, assigning a unique **sentence_id** to each sentence.
- This step is crucial for preparing the data for sentiment analysis and word frequency calculations.

4. Sentiment Analysis

We performed sentiment analysis to calculate sentiment scores for each sentence and classified them into positive, negative, or neutral categories.

```
# Add back sentences and sentiments
CBL02d <- CBL02b %>%
  get_sentences(text) %>%
  sentiment() %>%
  drop_na() %>%   # Remove empty lines
```

```
  mutate(sentence_id = row_number())

# Exclude sentiments with '0' value
CBL02e <- subset(CBL02d, CBL02d$sentiment != 0.00)
CBL02e1 <- subset(CBL02e, CBL02e$sentiment > 0.00)  # Positive sentiments
CBL02e2 <- subset(CBL02e, CBL02e$sentiment < 0.00)  # Negative sentiments
CBL02e3 <- subset(CBL02e, CBL02e$sentiment == 0.00) # Neutral sentiments

View(CBL02e)
```

-

	sentence_id	word	element_id	word_count	sentiment	ave_sentiment	sd_sentiment
1	38	information	43	1	0.40	0.40	NA
2	42	general	48	1	0.40	0.40	NA
3	46	general	54	1	0.40	0.40	NA
4	57	general	67	1	0.40	0.40	NA
5	61	chairman	73	1	0.25	0.25	NA
6	96	loss	119	1	-0.75	-0.75	NA
7	99	comprehensive	122	1	0.75	0.75	NA
8	116	equity	142	1	1.00	1.00	NA
9	122	equity	149	1	1.00	1.00	NA
10	128	cash	156	1	0.40	0.40	NA

The **get_sentences** and **sentiment** functions assigned sentiment scores to each sentence.

- Sentences with a sentiment score of zero were excluded, as they do not contribute meaningful information.
- The dataset was further divided into positive, negative, and neutral subsets for detailed analysis.

5. Word Frequency Analysis

We calculated the frequency of each word in the comments to identify the most commonly mentioned terms.

```
# Find frequency of words
CBL02f <- CBL02e %>%
  group_by(word) %>%
  summarise(freq = sum(word_count))

View(CBL02f)
```

	word	freq
1	abandoned	2
2	ability	1
3	absolute	1
4	abuse	1
5	accept	3
6	acceptable	1
7	accountability	1
8	accountant	4
9	accredited	2
10	action	2

- Words were grouped, and their frequencies were calculated using the **group_by** and **summarise** functions.
- This step helps in identifying key themes and topics discussed in the comments.

6. Word Cloud Visualization

- A word cloud was generated to visually represent the most frequently occurring words. The **wordcloud** function created a visual representation of word frequencies, with larger font sizes indicating higher frequency.

  This visualization aids in quickly identifying dominant themes in the comments.

```
# Create word cloud
set.seed(1234)
wordcloud(words = CBL02f$word, freq = CBL02f$freq, min.freq = 1,
          max.words=200, random.order=FALSE, rot.per=0.35,
          colors=brewer.pal(8, "Dark2"))
```

7. Sentiment Summary

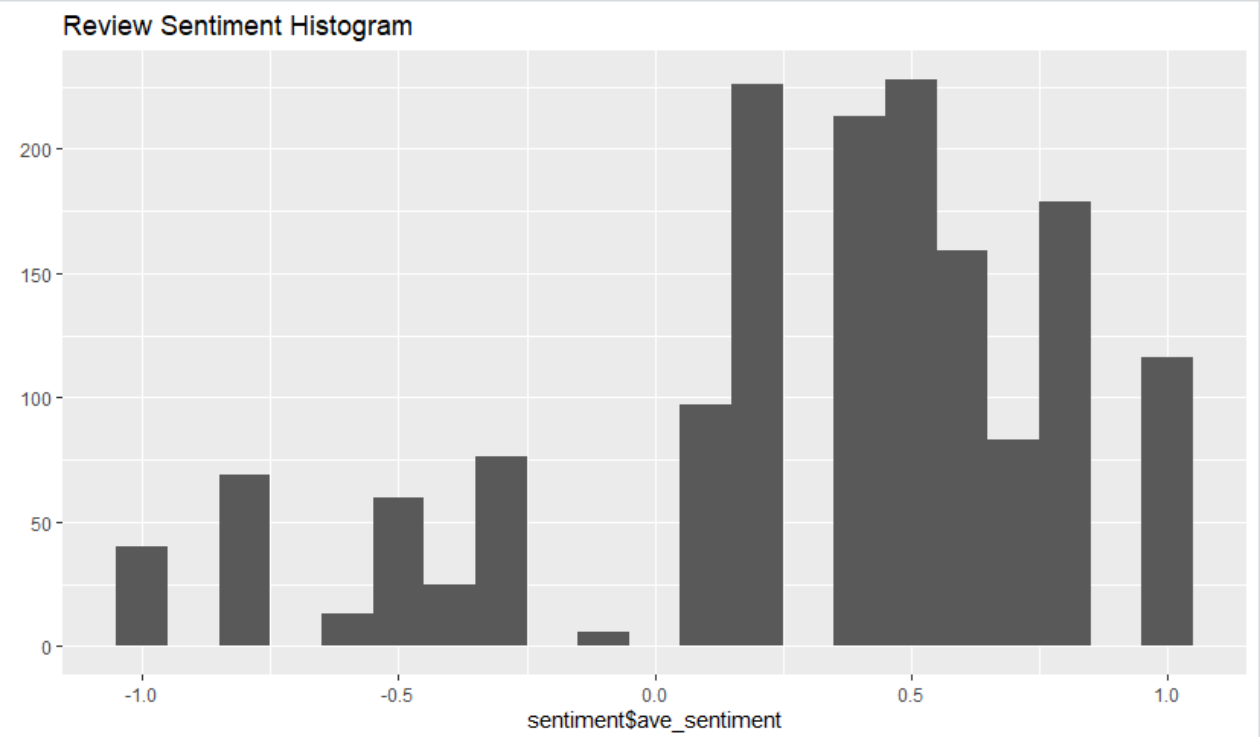
We summarized the sentiment scores and visualized their distribution using a histogram.

```
# Summarize sentiments
summary(CBL02e$sentiment)

# Histogram of sentiment scores
sentiment <- sentiment_by(CBL02e$word)
summary(sentiment$ave_sentiment)

CBL02e$ave_sentiment <- sentiment$ave_sentiment
CBL02e$sd_sentiment <- sentiment$sd

library(ggplot2)
pplt <- qplot(sentiment$ave_sentiment, geom="histogram", binwidth=0.1, main="Review Sentiment Histogram")
pplt
```



- The **summary** function provided statistical insights into the sentiment scores, including mean, median, and range.
- A histogram was generated to visualize the distribution of sentiment scores, highlighting the prevalence of positive, negative, or neutral sentiments.