Report - Reddit Data Analysis

Task 1: Topic Modeling

1. Introduction:

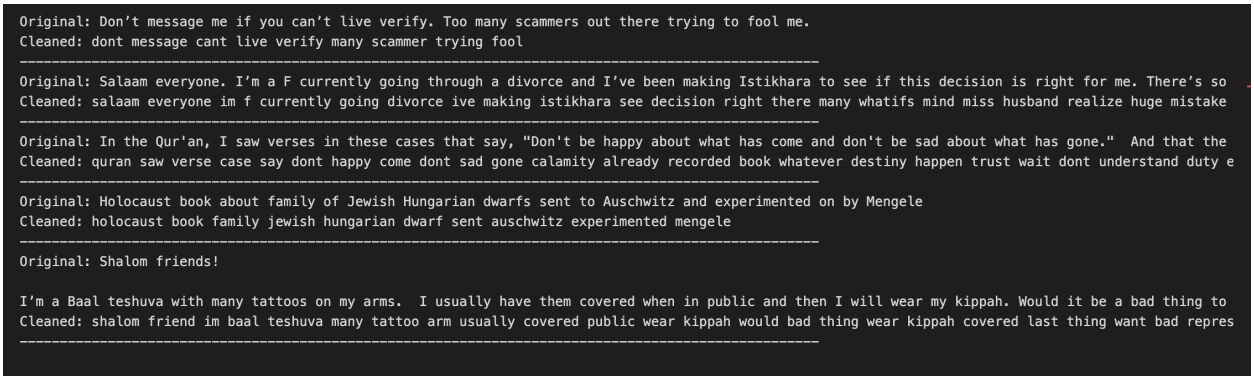
This section of the analysis focuses on discovering underlying themes in Reddit discussions about the Israel-Hamas conflict using Latent Dirichlet Allocation (LDA). The primary goal is to extract meaningful topics from the posts, label these topics, and visualize them using word clouds.

2. Preprocessing the Data:

Before applying LDA, the raw textual data was preprocessed to ensure cleaner input and better topic modeling. The following steps were taken:

1. **Converting to Lowercase:**
    - Ensures uniformity in text representation.
2. **Removing Mentions and URLs:**
    - Stripped out user mentions and URLs to focus on textual content.
3. **Removing Special Characters and Numbers:**
    - Removed hashtags, special characters, and numbers to clean the data further.
4. **Lemmatization:**
    - Normalized words to their root forms (e.g., "running" → "run").
5. **Stopword Removal:**
    - Removed common stopwords (e.g., "the", "is") to focus on meaningful terms.

Sample Results of Preprocessing:



3. Topic Modeling Using LDA:

To identify themes in the dataset using LDA. This involves:

- Experimenting with different numbers of topics.
- Selecting the best model using coherence scores.

LDA Model Steps:

1. **Vectorizing Text:**
    - Used CountVectorizer to convert the cleaned text into a document-term matrix.
2. **Applying LDA:**
    - Tested different topic counts (1–10).
    - Evaluated coherence scores to find the optimal number of topics.
3. **Coherence Scores:**
    - Coherence scores were used to evaluate topic quality. Below are the results:

Number of Topics	Coherence Score
1	0.4173
2	0.6500 (Best)
3	0.6253
4	0.6021
5	0.6201
6	0.6315

The optimal number of topics selected was **2**, as it had the highest coherence score of **0.6500**.

Topics and Their Interpretations:

- **Topic 1: Personal Experiences and Feelings**
  Key Terms: "im", "feel", "time", "people", "want", "allah"
- **Topic 2: Conflict and Politics**
  Key Terms: "israel", "hamas", "palestine", "war", "gaza", "jewish"

4. Labeling and Visualization

Topic Labeling:

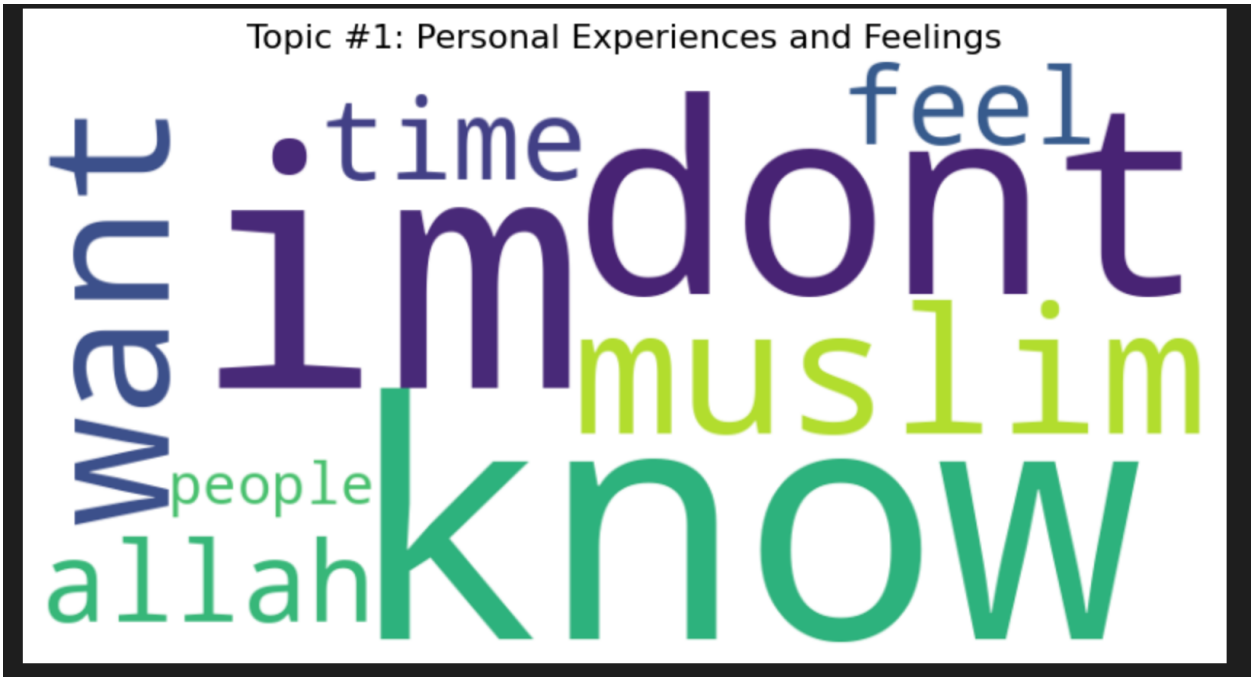
Based on the key terms for each topic, the following labels were assigned:

1. Topic 1: Personal Experiences and Feelings
2. Topic 2: Conflict and Politics

Word Cloud Visualizations:

Below are the word clouds for each topic:

Topic 1: Personal Experiences and Feelings



Topic 2: Conflict and Politics

The LDA model identified two distinct themes:

- Topic 1: Personal anecdotes and emotions around religious and life decisions.
- Topic 2: Political and conflict-related discussions about the Israel-Hamas conflict.

The word clouds highlight the most frequently discussed terms, providing further evidence of these themes.

The analysis successfully captured meaningful themes from the Reddit dataset, providing insights into user discussions around this significant geopolitical event.

Task 2 - Named Entity Recognition (NER)

The goal of this task was to analyze how specific entities related to the Israel-Hamas conflict are portrayed in Reddit discussions. By using Named Entity Recognition (NER) and sentiment analysis, we aimed to determine whether mentions of these entities carried a positive, negative, or neutral sentiment. Additionally, by extracting parts of speech (verbs, nouns, and adjectives) from sentences containing these entities, we could further explore the context in which these entities were discussed.

1) NER Extraction

The spaCy library was used to identify sentences mentioning key entities such as:

- Israel
- Hamas
- Antony Blinken
- Benjamin Netanyahu
- IDF (Israeli Defense Forces)

2) Sentiment Analysis

VADER was employed to compute sentiment scores for sentences mentioning the target entities. It provides a compound score that ranges from -1 (most negative) to 1 (most positive).

3) Part-of-Speech (POS) Tagging

For each sentence mentioning a target entity, nouns, verbs, and adjectives were extracted using NLTK's POS tagging.

4) Sentiment Scores

The following table summarizes the average sentiment scores for each entity:

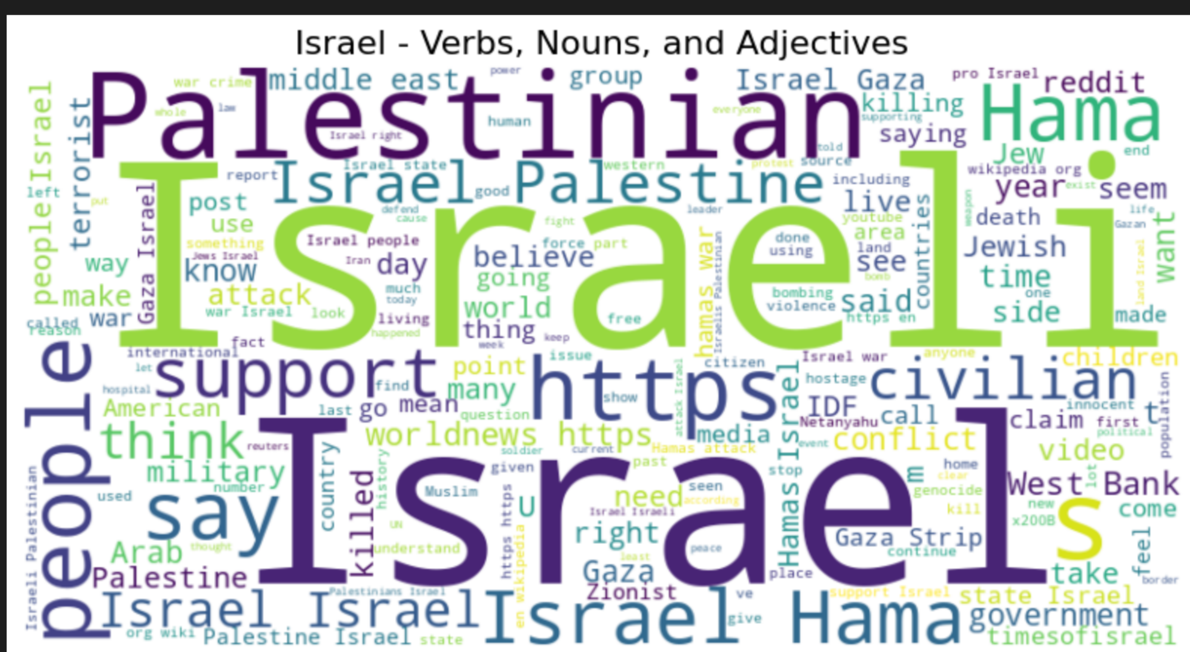
=== Entity Sentiment Summary ===

Entity	Average Sentiment Score
Israel	-0.1126
Netanyahu	-0.1003
IDF	-0.1987
Hamas	-0.1906
Benjamin Netanyahu	-0.1550
Israeli Defense Forces	-0.1534
Antony Blinken	-0.1039

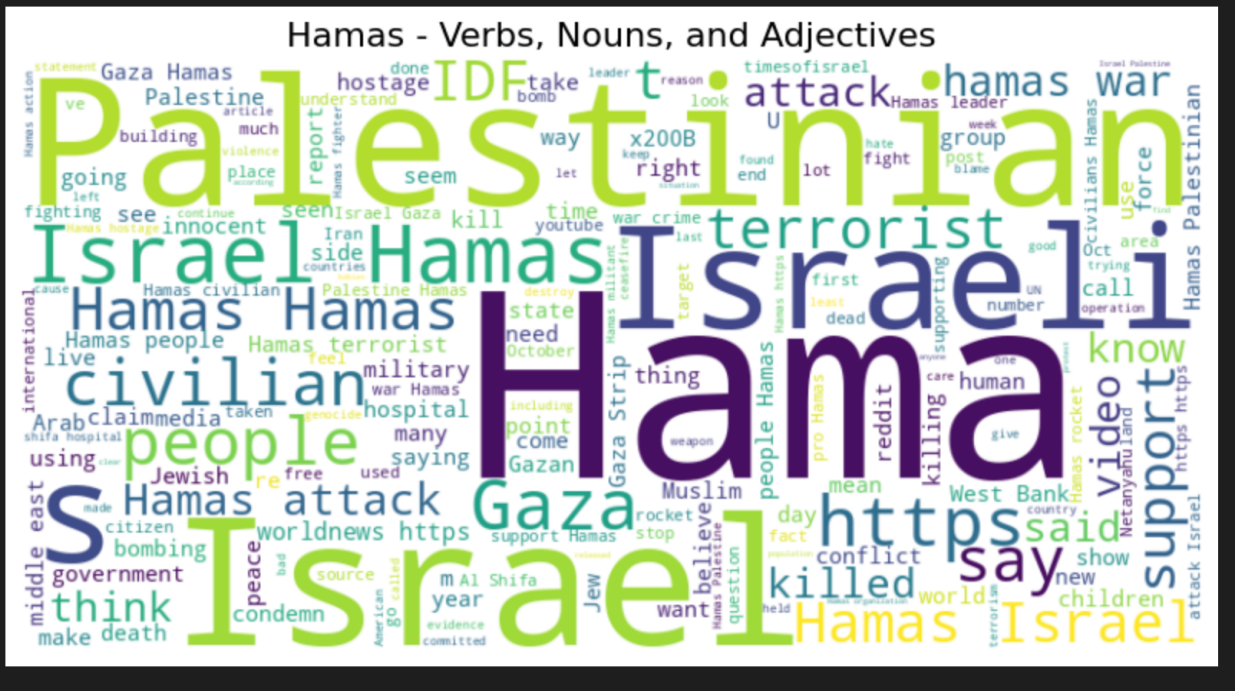
5) Visualizations

Word clouds were generated to visualize the most frequently used nouns, verbs, and adjectives in the context of each entity.

```
Generating Visualizations:
Word Cloud for Israel
```

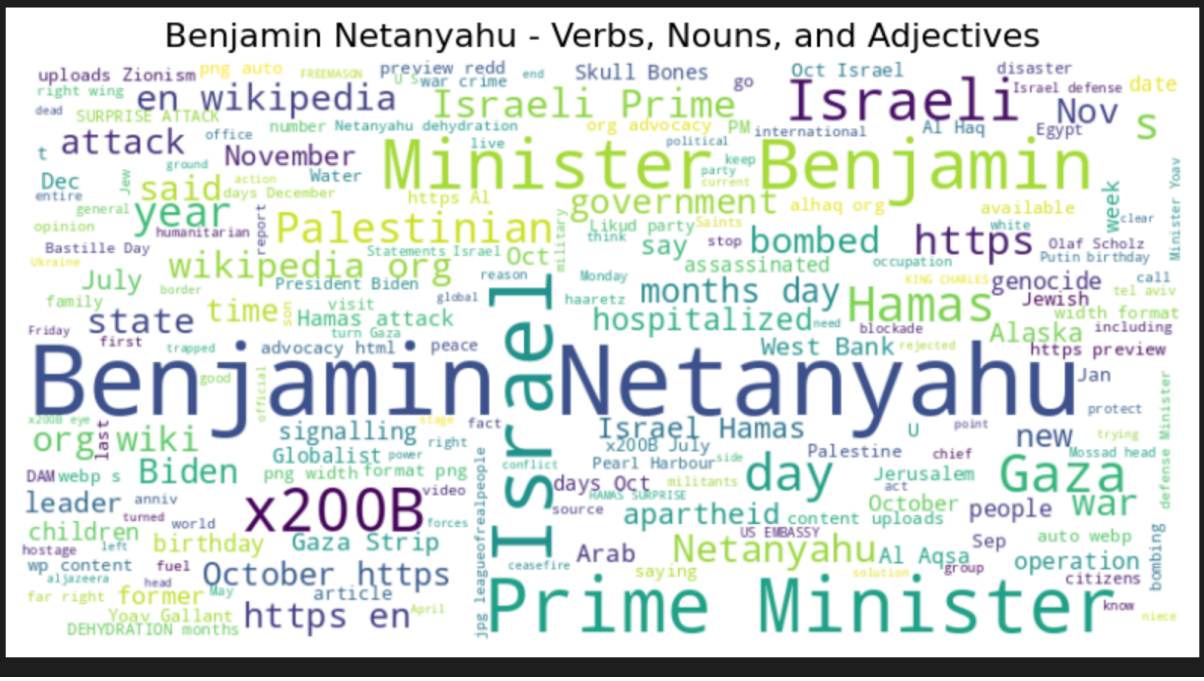


Word Cloud for Hamas



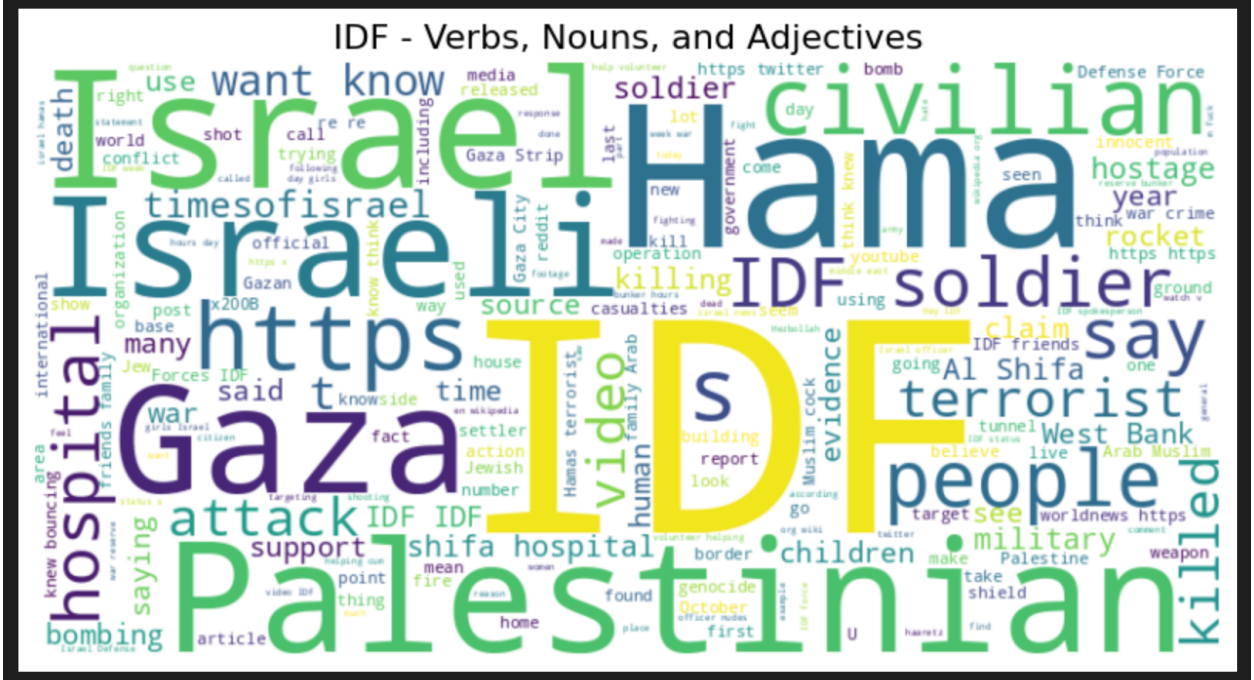
Word Cloud for Antony Blinken

Word Cloud for Benjamin Netanyahu

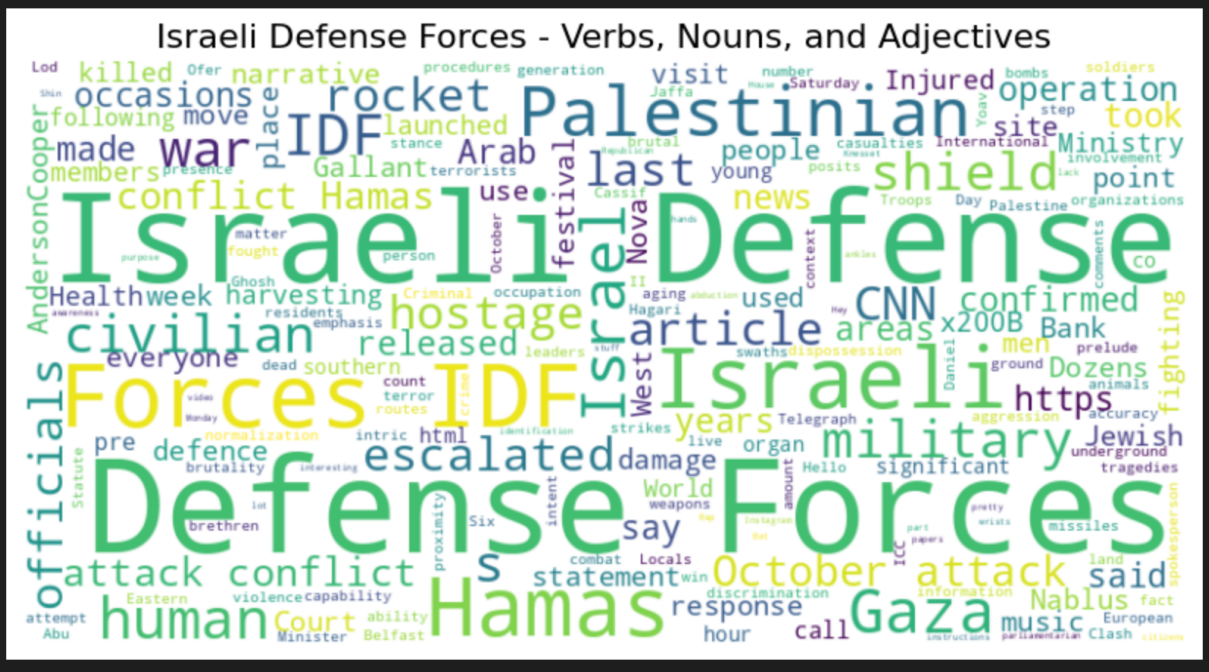


Word Cloud for Netanyahu

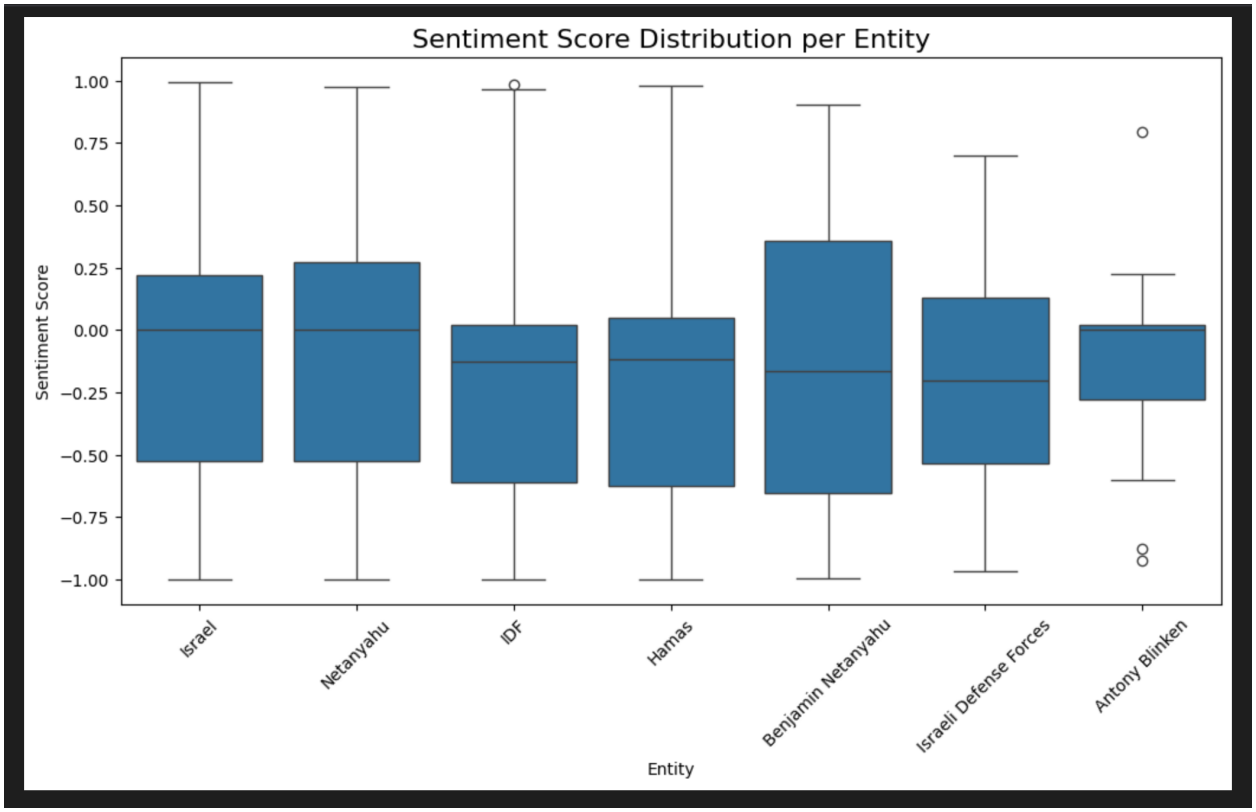
Word Cloud for IDF

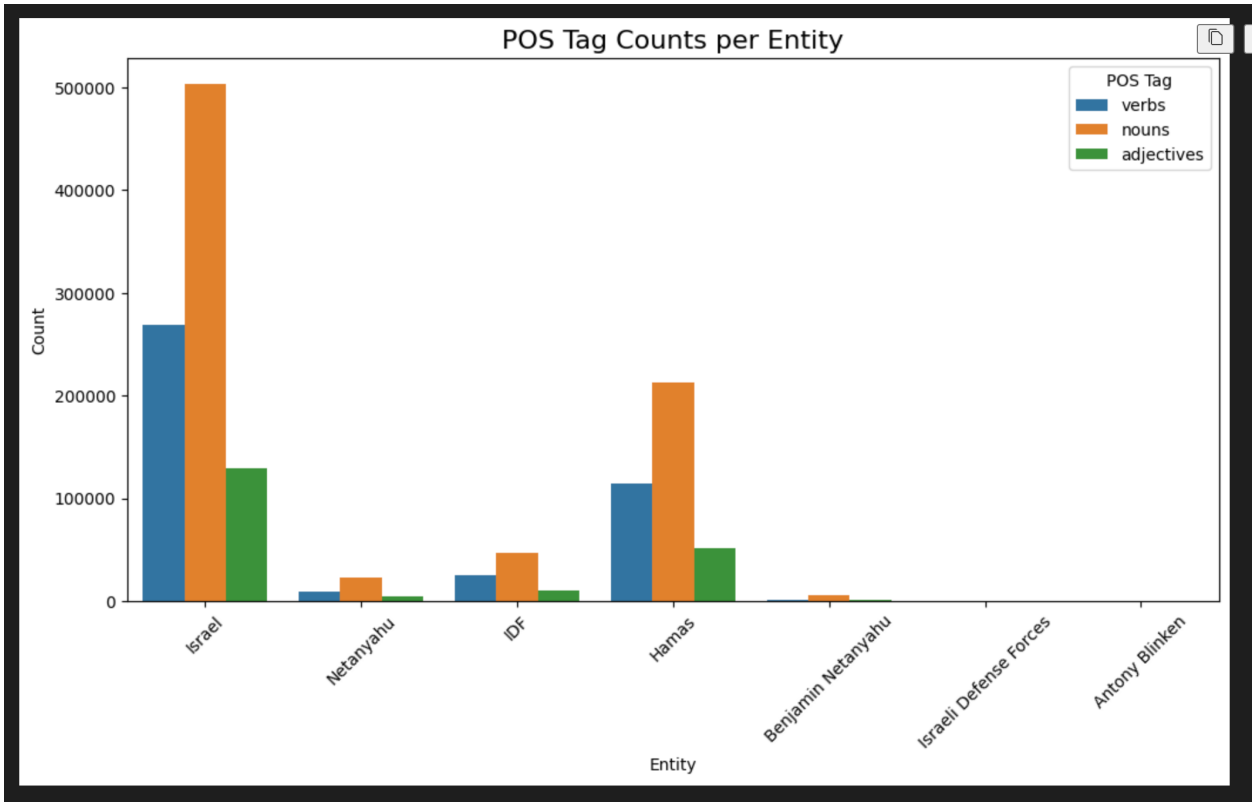


Word Cloud for Israeli Defense Forces



Boxplots provided sentiment score distributions across entities.





The NER and sentiment analysis task provided valuable insights into how key entities involved in the Israel-Hamas conflict are perceived within Reddit discussions:

- Sentiments are predominantly negative across all entities, with varying degrees of intensity.
- The combination of sentiment scores and POS-tagged word clouds helped capture the nuanced portrayal of entities in public discourse.

Task 3: Predicting Score for Each Reddit Post

The primary objective of Task 3 is to build predictive models that estimate the "score" of Reddit posts based on their title and text. The "score" represents the net upvotes (upvotes minus downvotes) a post received. The task involves using text-based features and training machine learning models to predict this score.

1. Data Preparation:

Loading and Preprocessing the Data

- The dataset (combined_file.csv) was loaded.
- To enhance the predictive power of the models, both the title and text of each Reddit post were combined into a single combined_text field.
- The combined_text column was then cleaned to remove noise:
    - Special characters and numbers were removed.
    - Text was converted to lowercase.
    - Any extra whitespace was stripped.

2. Feature Selection:

2.1 Topic Modeling Using LDA

- Latent Dirichlet Allocation (LDA) was employed to extract thematic structures from the text.
- The LDA model was trained on the cleaned text, with the number of topics ranging from 1 to 10.
- Coherence scores were used to evaluate topic quality, and the optimal number of topics was determined as 3, with a coherence score of 0.6713.

Topics Extracted:

1. Topic 1: jewish, people, muslims, islam, reddit
2. Topic 2: like, feel, want, people, allah
3. Topic 3: israel, hamas, gaza, palestine

Topic Features: The LDA model generated topic distributions for each document, providing three topic probabilities for each Reddit post. These distributions were stored as features.

2.2 Sentence Embeddings Using Word2Vec

- **Word2Vec** embeddings were utilized to capture semantic information from the text.
- A Word2Vec model was trained on the cleaned text using a vector size of **100** and a context window of **5**.
- For each document, the average Word2Vec embedding was computed, yielding a 100-dimensional feature vector.

3. Model Training and Evaluation

3.1 Feature Combination

The final feature set consisted of:

1. **LDA Topic Distributions** (3 features).
2. **Word2Vec Embeddings** (100 features).

These features were combined into a single dataset for model training.

3.2 Train-Test Split

- The dataset was split into **training (80%)** and **testing (20%)** sets.
- Features were standardized using StandardScaler to normalize the data.
-

4. Model Training and Performance

4.1 Support Vector Machine (SVM) with RBF Kernel

- **Model:** Support Vector Regression (SVR) with a radial basis function (RBF) kernel was used to capture complex, non-linear relationships in the data.
- **Hyperparameters:**
  - C=1.0
  - gamma='scale'
- **Performance:**
  - **Mean Squared Error (MSE):** 7900.2152
  - **Accuracy:** -0.0220 (indicating poor predictive power)
  -

4.2 XGBoost Regressor

- **Model:** XGBoost Regressor, known for its performance and flexibility with structured data, was applied.
- **Hyperparameters:**
  - n_estimators=100
  - learning_rate=0.1
- **Performance:**
  - **Mean Squared Error (MSE):** 7978.9375
  - **Accuracy:** -0.0322 (similar to SVM)

5. Evaluation Metrics

- **Mean Squared Error (MSE):** Both models had relatively high MSE, indicating that predicting Reddit post scores is a challenging task due to high variability in the data.
- Both **SVM** and **XGBoost** exhibited comparable performance, with SVM slightly outperforming XGBoost in terms of MSE.
- Neither model achieved strong predictive accuracy, indicating the need for more diverse or domain-specific features.