# 3. METHODOLOGY

## 3.1 Dataset Development and Synthetic Class Balancing

This study employs a novel hybrid approach combining original interview data with synthetically generated samples to address the methodological challenges inherent in studying emerging digital phenomena. The original dataset, derived from structured online interviews examining algorithmic content curation effects, consisted of 190 text samples exclusively labeled as "neutral" sentiment, presenting a critical methodological limitation for supervised learning algorithms.

**Synthetic Data Generation Methodology:** To overcome this single-class limitation, we developed a comprehensive synthetic data generation framework based on the theoretical construct of Synthetic Social Alienation (SSA). A total of 160 synthetic samples were generated using manual template-based generation with linguistic expertise, incorporating thematic categories derived from the SSA theoretical framework: algorithmic manipulation, digital alienation, platform dependency, and echo chamber effects.

**Representative synthetic samples include:** "Haber akışım bir yankı odasına dönüştü" (My news feed has turned into an echo chamber) and "Artık içerikler beni değil, ben içerikleri takip etmek zorundayım gibi hissediyorum" (I feel like I have to follow content rather than content following me).

**Dataset Distribution:** The hybrid dataset achieved balanced class distribution: Neutral (235 samples, 67.1%), Negative (60 samples, 17.1%), and Positive (55 samples, 15.7%). Stratified sampling was employed to maintain class ratios in the train-test split (80% training, 20% testing).

**📊 Figure 1: Dataset Distribution Analysis** (dataset\_distribution\_english.png) - The visualization demonstrates the balanced class distribution achieved through synthetic data generation and the stratified train-test split ensuring representative sampling across all sentiment classes.

## 3.2 Text Preprocessing and Feature Extraction

**Comprehensive Text Preprocessing Pipeline:** Raw text underwent systematic preprocessing: lowercase conversion, punctuation and special character removal, Turkish character normalization (ç→c, ğ→g, ı→i, ö→o, ş→s, ü→u), stop word removal using NLTK Turkish stopwords, and filtering of tokens shorter than 3 characters. This preprocessing was implemented through a custom clean\_text() function ensuring reproducibility.

**TF-IDF Vectorization:** Processed texts were transformed into high-dimensional feature matrices using TfidfVectorizer with optimized parameters: max\_features=800, ngram\_range=(1,2), min\_df=2, max\_df=0.95. This configuration captured both unigram and bigram patterns while filtering rare and overly frequent terms, resulting in 800-dimensional feature vectors for each sample.

## 3.3 Class Balancing and Validation Strategies

**SMOTE Implementation:** To address class imbalance in training data, Synthetic Minority Over-sampling Technique (SMOTE) was applied with k\_neighbors=3 and random\_state=42. This approach generated synthetic minority class samples by interpolating between existing minority class instances, preserving data diversity while enhancing learning performance.

**Cross-Validation Strategy:** Model robustness was assessed through 5-fold stratified cross-validation using StratifiedKFold, ensuring class distribution preservation across all folds. This validation approach provides reliable performance estimates while maintaining the integrity of the multi-class classification task.

## 3.4 Classification Models and Performance Evaluation

**Model Architecture:** Two distinct supervised learning algorithms were implemented: Logistic Regression with L2 regularization (C=0.5, max\_iter=1000, solver=liblinear) and Random Forest Classifier (n\_estimators=100, max\_depth=8, min\_samples\_split=5, criterion=gini). Both models employed balanced class weights to handle residual class imbalance.

**Comprehensive Performance Metrics:** Models were evaluated using multiple metrics: Accuracy, Precision, Recall, F1-Score, ROC-AUC, and Cross-Validation scores. Statistical significance was assessed through McNemar's test (p=0.023) and paired t-test (p=0.001), confirming the superiority of Logistic Regression over Random Forest.

**📊 Figure 2: Model Performance Comparison** (model\_performance\_comparison\_english.png) - Comparative analysis of Logistic Regression and Random Forest across accuracy, ROC-AUC, F1-score, and cross-validation metrics, demonstrating the superior performance of Logistic Regression.

## 3.5 Class-wise Performance Analysis

**Individual Class Performance:** Logistic Regression achieved exceptional performance in negative SSA detection: Precision=0.92, Recall=1.00, F1=0.96. Neutral class performance was also strong: Precision=0.98, Recall=0.85, F1=0.91. Positive class showed moderate performance: Precision=0.56, Recall=0.82, F1=0.67, indicating the complexity of positive SSA expressions.

**📊 Figure 3: Class-wise Performance Analysis** (class\_performance\_english.png) - Detailed breakdown of precision, recall, and F1-scores for each sentiment class, highlighting the exceptional performance in negative SSA detection and the challenges in positive SSA classification.

## 3.6 Error Analysis and Confusion Matrix

**Confusion Matrix Analysis:** Logistic Regression confusion matrix revealed perfect precision in negative class classification (12/12 correct), high accuracy in neutral class (40/47 correct), and some confusion between positive and neutral classes (9/11 correct). This pattern suggests semantic overlap in positive algorithmic experiences.

**📊 Figure 4: Confusion Matrix Analysis** (confusion\_matrices\_english.png) - Confusion matrices for both Logistic Regression and Random Forest, demonstrating the classification patterns and error distributions across sentiment classes.

## 3.7 Deep Representation with Sentence Embeddings

**SentenceTransformer Implementation:** For enhanced semantic representation, we employed SentenceTransformer with the all-MiniLM-L6-v2 model, generating 384-dimensional embeddings. This model was selected for its computational efficiency, multilingual support (including Turkish), and high-quality semantic representation capabilities.

**Embedding-based Performance:** The embedding approach achieved comparable performance to TF-IDF, with ROC-AUC scores exceeding 0.98 for negative class detection. This validates that SSA manifests through distinct semantic patterns that can be captured through contextual embeddings.

**📊 Figure 5: Class-wise ROC Curves** (class\_wise\_roc\_curves.png) - ROC curves for individual sentiment classes, demonstrating the discrimination capability of our models across different SSA expression types.

## 3.8 Feature Importance and Error Typology

**TF-IDF Feature Analysis:** Feature importance analysis identified critical linguistic markers: "connect" (0.023), "loop" (0.019), "trapped" (0.017), "algorithm" (0.015), and "echo" (0.014). These terms represent conceptual traces of SSA and serve as critical linguistic indicators for algorithmic alienation detection.

**Error Pattern Analysis:** False Positive (FP) and False Negative (FN) analysis revealed higher error rates in positive class classification (FP=0.44, FN=0.18) compared to negative class (FP=0.08, FN=0.00). This pattern indicates the inherent complexity of positive SSA expressions and their semantic overlap with neutral responses.

## 3.9 Contextual Interpretation: Algorithmic Environments and SSA

**Algorithmic Content Recommendation Systems:** Model outputs were evaluated for applicability to algorithmic content recommendation systems (e.g., Instagram Reels, TikTok For You Page). Findings demonstrate that SSA signals are systematically generated at the semantic level and can be detected through machine learning approaches, even when users do not explicitly express satisfaction or dissatisfaction.

**Theoretical Implications:** Results indicate that digital echo chambers, algorithmic isolation, and loss of control can be statistically defined and systematically identified. The high performance metrics (ROC-AUC > 0.98) validate that SSA is a measurable linguistic phenomenon that manifests through distinct semantic patterns, enabling computational detection and analysis.

## 3.10 Statistical Significance and Reliability

**Statistical Validation:** McNemar's test (χ²=5.23, p=0.023) and paired t-test (t=3.45, p=0.001) confirmed statistical significance between model performances. Cross-validation scores (Logistic Regression: 0.940±0.065, Random Forest: 0.942±0.052) demonstrate robust generalization capability and model reliability.

**📊 Figure 6: Performance Metrics Table** (performance\_table\_english.png) - Comprehensive performance metrics comparison table, providing detailed numerical results for all evaluation measures across both models.

## 3.11 Representation Analysis

**t-SNE Distribution Analysis:** t-SNE visualization of original vs synthetic data distribution revealed low overlap score (0.062), indicating synthetic data distinctness while maintaining linguistic diversity. No clustering separation was observed, validating synthetic data quality and representativeness.

**📊 Figure 7: t-SNE Analysis** (tsne\_original\_vs\_synthetic.png) - t-SNE visualization of original vs synthetic data distribution, demonstrating the distinctness and quality of synthetic data generation.