

Методология обнаружения ботнетов в комментариях

Теоретическая основа

Обнаружение ботнетов в комментариях основано на комбинации нескольких методов анализа, включающих:

1. **Взвешенную оценку признаков (Feature-based scoring)** - основной метод, использующий набор признаков поведения с соответствующими весами
2. **Обнаружение аномалий (Anomaly detection)** - для выявления отклонений от нормального поведения пользователей
3. **Кластеризацию (Clustering)** - для группировки ботов по схожим поведенческим паттернам
4. **Сетевой анализ (Network analysis)** - для выявления связей между ботами через комментируемые видео

Математическая модель определения ботов

1. Взвешенная система оценки (Weighted Scoring System)

Основа метода - вычисление "оценки бота" (bot score) как взвешенной суммы нормализованных признаков:

$$\text{BotScore}(a) = \sum_{i=1}^n w_i \cdot f_i(a)$$

Где:

- a - автор (аккаунт)
- $f_i(a)$ - нормализованное значение i -го признака для автора a
- w_i - вес i -го признака
- n - количество признаков

Ключевые признаки и их веса в модели:

```
weights = {
    'comment_frequency': 0.15,      # Частота комментирования
    'cv_comment_length': -0.10,     # Коэффициент вариации длины комментария
    'comment_similarity': 0.20,     # Сходство комментариев
    'keyword_ratio': 0.15,          # Доля ключевых слов
    'hours_std': -0.10,             # Стандартное отклонение времени постинга
    'time_entropy': -0.10,          # Энтропия времени постинга
    'unique_comments_ratio': -0.15, # Доля уникальных комментариев
    'comment_to_video_ratio': 0.05, # Отношение комментариев к видео
    'videos_per_day': 0.10,         # Видео в день
}
```

Примечание: Отрицательные веса указывают на обратную зависимость - чем ниже значение признака, тем выше вероятность, что аккаунт является ботом.

2. Нормализация признаков

Все признаки нормализуются к диапазону [0, 1]:

$$f_i(a) = \frac{f_i(a) - \min(f_i)}{\max(f_i) - \min(f_i)}$$

Для признаков с отрицательными весами, нормализованное значение инвертируется:

$$f'_i(a) = 1 - f_i(a)$$

3. Расчет сходства комментариев

Сходство комментариев вычисляется на основе косинусного сходства TF-IDF векторов:

$$\text{Similarity}(a) = \frac{1}{|C_a|(|C_a|-1)} \sum_{i=1}^{|C_a|} \sum_{j=i+1}^{|C_a|} \cos(\text{TFIDF}(c_i), \text{TFIDF}(c_j))$$

Где:

- C_a - множество комментариев автора a
- $\text{TFIDF}(c_i)$ - TF-IDF вектор комментария c_i
- $\cos(v_1, v_2)$ - косинусное сходство между векторами v_1 и v_2

TF-IDF (Term Frequency-Inverse Document Frequency) рассчитывается как:

$$\text{TFIDF}(t, c, C) = \text{TF}(t, c) \cdot \text{IDF}(t, C)$$

Где:

- $\text{TF}(t, c)$ - частота термина t в комментарии c
- $\text{IDF}(t, C)$ - обратная частота документа: $\log \frac{|C|}{|\{c \in C: t \in c\}|}$

4. Автоматическая настройка порога (Threshold Tuning)

Для классификации аккаунта как бота используется пороговое значение. В автоматическом режиме порог определяется методом "локтя" (elbow method):

1. Сортируются все значения bot_score : $[s_1, s_2, \dots, s_n]$
2. Вычисляются разности между последовательными значениями: $\Delta_i = s_{i+1} - s_i$
3. Определяется порог для крупных разрывов: $\Delta_{\text{threshold}} = \text{percentile}(\Delta, 95)$
4. Находятся индексы крупных разрывов: $I = \{i \mid \Delta_i > \Delta_{\text{threshold}}\}$
5. Выбирается первый крупный разрыв после медианы: $i^* = \min\{i \in I \mid i \geq \frac{n}{2}\}$
6. Устанавливается порог: $\text{threshold} = s_{i^*}$

Кластеризация ботов

После обнаружения ботов производится их кластеризация для выявления ботнетов. Основные методы:

1. HDBSCAN (Hierarchical Density-Based Spatial Clustering of Applications with Noise)

HDBSCAN выбран как основной метод кластеризации из-за его способности:

- Обнаруживать кластеры произвольной формы
- Определять шум (outliers)
- Не требовать предварительного знания о количестве кластеров

Математически HDBSCAN строит иерархию плотностных кластеров и извлекает наиболее стабильные кластеры:

1. Строится граф взаимной достижимости по плотности (mutual reachability graph)
2. Строится минимальное остовное дерево (MST) этого графа
3. Строится иерархический кластер-дендрограмма
4. Извлекаются кластеры, максимизирующие стабильность

Оптимальные параметры HDBSCAN определяются путем поиска по сетке (grid search) на основе комбинированной метрики качества:

$$\text{Score} = 0.4 \cdot n_{\text{clusters}} + 0.3 \cdot \text{non_noise_ratio} + 0.3 \cdot \text{silhouette}$$

Где:

- n_{clusters} - количество обнаруженных кластеров
- non_noise_ratio - доля точек, отнесенных к кластерам
- silhouette - показатель силуэта

2. Gaussian Mixture Models (GMM)

В качестве альтернативного метода используется модель гауссовых смесей. Оптимальное количество компонентов определяется по критерию BIC (Bayesian Information Criterion):

$$\text{BIC} = -2 \ln(L) + k \ln(n)$$

Где:

- L - максимальное значение функции правдоподобия модели
- k - количество параметров модели
- n - количество наблюдений

Сетевой анализ для обнаружения ботнетов

Ключевой компонент обнаружения ботнетов - построение и анализ двудольного графа (bipartite network) отношений бот-видео:

$$G = (A \cup V, E)$$

Где:

- A - множество аккаунтов ботов
- V - множество видео
- $E \subseteq A \times V$ - множество рёбер, соответствующих комментариям

1. Проекция графа

Для анализа отношений между ботами строится проекция двудольного графа на множество ботов:

$$G_A = (A, E_A)$$

Где ребро $(a_i, a_j) \in E_A$ существует, если боты a_i и a_j комментировали хотя бы одно общее видео. Вес ребра определяется количеством общих видео:

$$w(a_i, a_j) = |N(a_i) \cap N(a_j)|$$

Где $N(a)$ - множество видео, прокомментированных ботом a .

2. Обнаружение сообществ

Для обнаружения ботнетов в графе ботов применяется алгоритм выявления сообществ Лувена (Louvain method), который максимизирует модулярность графа:

$$Q = \frac{1}{2m} \sum_{i,j} \left[A_{ij} - \frac{k_i k_j}{2m} \right] \delta(c_i, c_j)$$

Где:

- A_{ij} - вес ребра между узлами i и j
- k_i - сумма весов рёбер, подключенных к узлу i
- m - сумма весов всех рёбер в графе
- c_i - сообщество, к которому принадлежит узел i
- $\delta(c_i, c_j)$ - функция Кронекера (1 если $c_i = c_j$, иначе 0)

3. Анализ синхронизированной активности

Для выявления координированного поведения ботов анализируется временное распределение комментариев. Синхронизированной считается активность, когда:

1. Несколько ботов (≥ 3) комментируют в течение одного часа
2. Стандартное отклонение времени размещения комментариев мало ($\sigma < 15$ минут)

$$\text{SynchronizationScore}(h) = \frac{|B_h|}{std_min(h)}$$

Где:

- B_h - множество ботов, активных в час h
- $std_min(h)$ - стандартное отклонение минут размещения комментариев в час h

Выводы

Представленная методология объединяет несколько подходов к обнаружению ботнетов:

1. **Анализ индивидуального поведения** - с использованием взвешенной системы оценки признаков и обнаружения аномалий
2. **Анализ групповой структуры** - с использованием кластеризации схожих поведенческих паттернов
3. **Анализ взаимосвязей** - с использованием сетевого анализа и выявления сообществ
4. **Анализ временных паттернов** - с выявлением синхронизированной активности

Такой многоуровневый подход позволяет не только обнаруживать отдельных ботов, но и выявлять ботнеты - координированные группы ботов, действующие вместе для достижения определенных целей.