# Обнаружение мультиавторности в тексте с помощью ансамбля моделей

А.Фаттахов, Р.Кузнецова, В.Стрижов

*Московский Физико-Тенический Институт*

**Abstract**

В работе рассматривается задача обнаружения мультиавторности в тексте. Задача обнаружения мультиавторности возникает, когда необходимо проверить текст на наличие заимствований, но при этом нет доступа к внешним корпусам. Предполагается, что авторы имеют свой уникальный стиль написания, и разнообразие стилей сигнализирует о вероятном заимствовании. Предлагается построить ансамбль базовых моделей для обнаружения мультиавторности. В качестве базовых моделей используются бустинг с применением текстовых статистик, рекуррентные сети LSTM, работающие с векторными представлениями слов, сверточные сети с векторными представлениями символов в качестве признаков. Вычислительный эксперимент проводится на выборке с соревнования PAN-2018.

## 1. Введение

Задача поиска заимствований в тексте [1] включает задачи обнаружения внешних и внутренних заимствований. Анализ внешних заимствований (external plagiarism detection) [2] заключается в попарном сравнении подозрительного текста с определенной коллекцией внешних текстов. Задача анализа внутренних заимствований (internal plagiarism detection) [3] состоит в обнаружении подозрительного текста без использования дополнительной коллекции, по которой ведется поиск. При этом алгоритмы анализа внутренних заимствований учитывают стиль письма и выявляют признаки в тексте, свойственные данному автору.

Задача анализа внутренних заимствований включает следующие подзадачи. Задача поиска участков в тексте, на которых происходит смена автора, рассматривается в PAN2011 [4]. Задача расширяется на авторскую диаризацию (Author Diarization) [5], PAN2016 [6], решение победителя [7]. Более узкая задача определения того, написан текст одним автором или несколькими, ставится в PAN2018 [8].

Задачи анализа внутренних заимствований решаются с помощью текстовых статистик. В [9] предложен статистический подход к решению задачи, основанный на

*Email addresses:* fattahov.ao@phystech.edu (А.Фаттахов), rita.kuznetsova@phystech.edu (Р.Кузнецова), strijov@phystech.edu (В.Стрижов)

*Preprint submitted to кпукпукп* *2 мая 2018 г.*

признаках tf-idf: словесные n-граммы (в работе рассматриватеся только $n = 1$ и $n = 3$), пунктуации, части речи с использованием PST Tagbank Penn Treebank [10]. В [11] предлагается подход с использованием аналогичных признаков, но в постановке задачи обучения без учителя. В [12] описан подход с отображением предложений в многомерное пространство, с последующим определением стилевой функции. Помимо статистических встречаются подходоы с применением нейронных сетей [13].

В данной работе предлагается использование ансамбля моделей для решения задачи определения того, написан текст одним автором или несколькими. Исследуется вклад каждой базовой модели в ансамбль. В качестве базовых моделей используются бустинг с применением текстовых статистик, рекуррентные сети LSTM, работающие с векторными представлениями слов, сверточные сети с векторными представлениями символов в качестве признаков. Работа ансамбля тестируется на данных с конкурса PAN2018 [8]. Для оценки качества используется отношение количества верно откласcифицированныъ текстов ко всем текстам, на которых делались предсказания модели.

## 2. Постановка задачи

Поставим задачу классификации текста формально. Дана выборка $D = \{(d_k, t_k)\}$, где $d_k$ — текстовый документ, а $t_k \in \{0, 1\}$ - является ли текст мультиавторным. Требуется построить модель классификации $f : d_k \mapsto y_k$, минимизирующую функцию ошибки $S$ (2) :

$$f = \frac{1}{|D|} \sum_{k=1}^{|D|} \arg\min_f S(y_k, t_k) \tag{1}$$

$$S = -\frac{1}{|D|} \sum_{k=1}^{|D|} t_k \log y_k + (1 - t_k) \log (1 - y_k) \tag{2}$$

В качестве ансамбля моделей предлагается использование стекинга [14]. Выборка документов $D$ разбивается на $k$ фолдов $F_i$: $D = \bigcup_{i=1}^{k} F_i$, при этом разбиение выбирается таким образом, чтобы распределения классов $t_i$ в фолдах совпадали. Параметры каждой модели 3.1-3.5 настраиваются на каждом из всевозможных объединений $k - 1$ фолдов и делается предсказание на оставшийся фолд. Все предсказания объединяются в новую выборку $\{d'_k, t_k\}$, где i-й столбец $d'_k$ - это предсказания $f_i(d_k)$ соответствующей модели. Полученная выборка рзделяется на $k$ аналогичных фолдов и обучается модель второго уровня $f' : d'_k \mapsto \{0, 1\}$.

## 3. Описание базовых моделей

### *3.1. Модели с использованием текстовых статистик*

Предлагается использовать градиентный бустинг деревьев, обученный на признаках, характеризующих частоты встречания слов(3.1.1) и символов(3.1.2) в тексте.

*3.1.1. Tf-idf слов*

Для кажого терма $w_i$ считается частота его встречания в тексте.

$$u_{kj} = \frac{1}{|d_k|} \sum_{\mathrm{i}\,=\,1}^{|d_k|} I(d_{ki} = w_j) \tag{3}$$

В качестве термов $w_i$ берутся слова, их группы, парные, двойные и тройные интеракции. К частотом слов применяется tf-idf [15] - статистическая мера, используемая для оценки важности слова в контексте документа, являющегося частью коллекции документов. Полученные частоты конкатенируются, образуя векторное представление текста $d_k$. Над полученными признаки обучается градиентый бустинг деревьев.

*3.1.2. Tf-idf символов*

Базовая модель аналогична 3.1.1, но вместо слов $w_i$ используются симолы $c_i$ и их n-граммы.

*3.2. Модель с прриментением функции стиля*

Предлагаемая в [12] модель работает с частотными признаками описания текста. В качестве таких признаков выбраны частоты встречаемости слов. Исходный текст подвергается предобработке: удаляются служебные символы, все буквы переводятся в нижний регистр. Также из текста удаляются стоп-слова. Текст разбивается на предложения. Затем формируется разбиение текста на сегменты $s_{ki}$: если длина очередного предложения меньше минимальной длины сегмента, к этому предложению добавляется следующее за ним — процесс повторяется, пока длина сегмента $s_{ki}$ не превысит заданную минимальную длину. Минимальная длина сегмента является настраиваемым параметром алгоритма. Для каждого сегмента $s_{ki}$ текста строится вектор признаков $\mathbf{s}_{ki}$. Затем строится статистика $\sigma(\mathbf{s}_{ki})$, которая сглаживается скользящим средним:

$$\sigma(\mathbf{s}_{ki}) = ||\mathbf{s}_{ki} - \mathbf{s}_{kave}|| \tag{4}$$

$$\mathbf{s}_{kave} = \frac{1}{n} \sum_{\mathrm{i}\,=\,1}^{n} \mathbf{s}_{ki} \tag{5}$$

$$\sigma'(\mathbf{s}_i) = \frac{1}{2n+1} \sum_{\mathrm{k}\,=\,\mathrm{i}\,-\,\mathrm{n}}^{i+n} \sigma(s_i) \tag{6}$$

Значения $\sigma'(s_i)$ исследуются на выбросы. С некоторого порога сегментов-выбросов текст классифицируется как мультиавторский.

$$f(\mathbf{d}_k) = I(\max_{i,j} |\sigma'(\mathbf{s}_{ki}) - \sigma'(\mathbf{s}_{kj})| > threshold) \tag{7}$$

*3.3. LSTM модель с векторными представлениями слов в качестве признаков*

Используется рекуррентная нейронная сеть на основе векторных представлений $\mathbf{w}_{ki}$ слов $w_{ki}$. В качестве отображения $e : w_{ki} \mapsto \mathbf{w}_{ki}$ возьмем представления слов, полученные с помощью модели word2vec [16]. $\mathbf{h}_i = LSTM(\mathbf{h}_{i-1}, \mathbf{w}_i)$ - внутреннее состояние сети на слове $\mathbf{w}_i$. После двух LSTM слоев конечное состояние передается на трёхслойный перцептрон с сигмоид активацией на последнем слое из одного нейрона. Архитектура сети приведена на (Рис. 1).

*3.4. Улучшеннаый алгоритм на основе LSTM*

Поскольку длина предложений в текстах из корпуса может составлять до 2000 слов, то наивное использование LSTM как в 3.3 дает не очень хорошие результаты. Поскольку внутреннее состояние ячейки LSTM не в состоянии фиксировать настолько длинные взаимодействия, то текст разбивается на сегменты $s_{ki}$, аналогичные 3.2. К каждому сегменту $s_{ki}$ применяется своя $\text{LSTM}_i$, после чего внутренние состояния конкатенируются в один вектор. Полученный вектор подается на вход трехслойному перцептрону, аналогичному 3.3. Архитектура сети приведена на (Рис. 2)

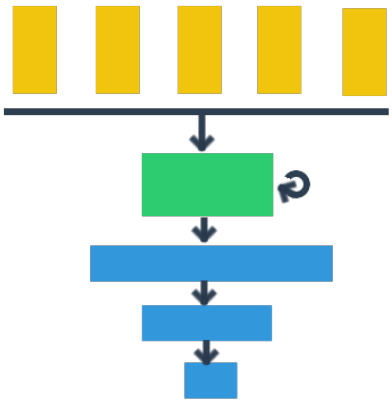

Рис. 1: Нейронная сеть на основе LSTM.

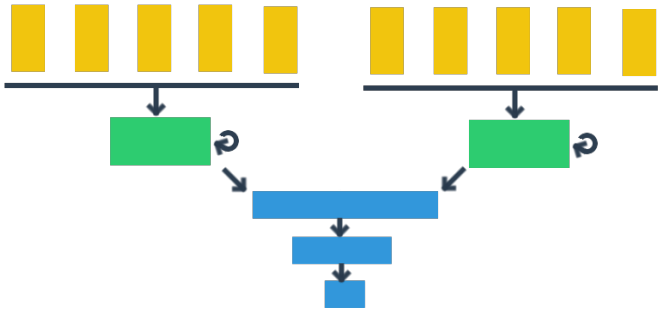

Рис. 2: Улучшенная архитектура на основ LSTM.

*3.5. Сверточная сеть на основе символьных представлений текстов*

Каждому уникальному символу $c'_{ki}$ текста $d_k$ ставится в соответствие one-hot-encoding вектор $\mathbf{c'}_{ki}$. Последовательность $\mathbf{c'}_{ki}$ подается на вход одномерной сверточной сети [17]. На выходе сети получется векторное представление $\mathbf{d_k}$ текста текста $d_k$. Полученное представление проходит через перцептрон, аналогичный 3.3, 3.4. Схема сети показана на (Рис. 3)

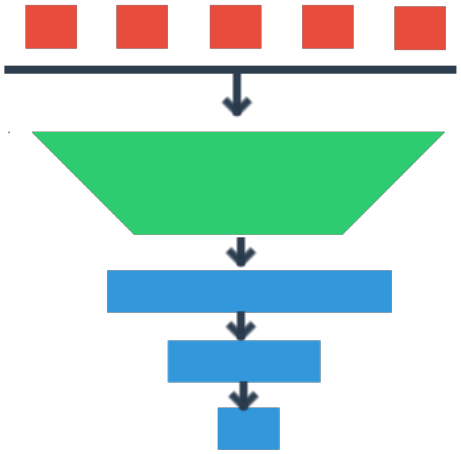

Рис. 3: Схема сети на основе char-CNN

## 4. Описание выборки

В работе используется набор текстовых документов с соревнования PAN-2018 [8], собранных с различных сайтов сети StackExchange [18]. Тематики и авторы у текстов различные. При этом у каждого текста может быть как один автор, так и несколько. Обучающая выборка состоит из 2980 текстов, при этом к каждому из них прилагается файл с разметкой. Несмотря на то, что стоит задача классификации, в разметке присутсвуют сами границы смены автора текста, что может помочь при разработке алгоритмов. Как видно из Рис. 4, количество заимствований линейно убывает, а большинство текстов имеет длину от двух до семи тысяч(Рис. 5).

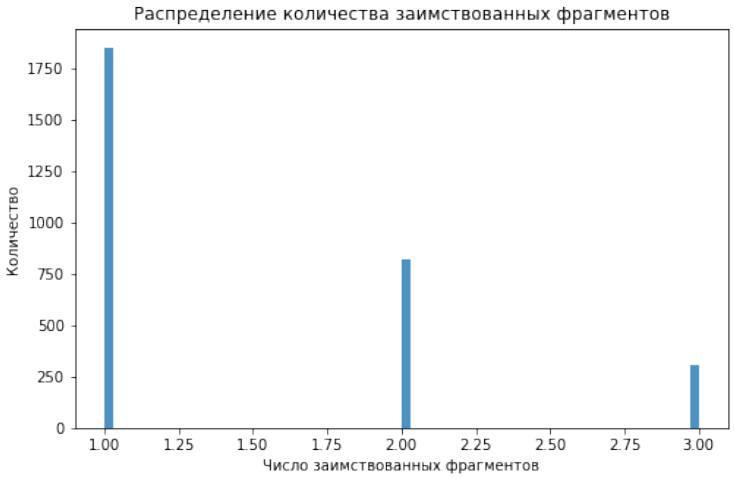


Рис. 4: Гистограмма распределения числа заимствований

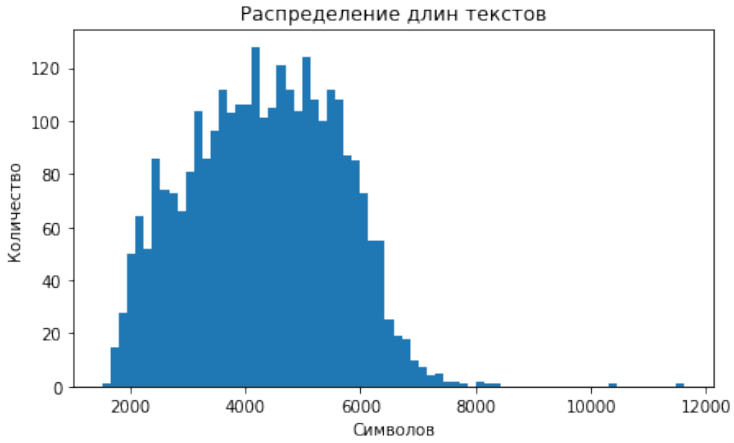


Рис. 5: Гистограмма распределения длины текстов

## 5. Ансамбль моделей

В качестве ансамбля моделей предлагается использование стекинга [14]. Выборка документов $D$ разбивается на $k$ фолдов $F_i$: $D = \bigcup_{i=1}^{k} F_i$, при этом разбиение выбирается таким образом, чтобы распределения классов $t_i$ в фолдах совпадали. Параметры каждой модели 3.1-3.5 настраиваются на каждом из всевозможных объединений $k-1$ фолдов и делается предсказание на оставшийся фолд. Все предсказания объединяются в новую выборку $\{d'_k, t_k\}$, где i-й столбец $d'_k$ - это предсказания $f_i(d_k)$ соответствующей модели. Полученная выборка рзделяется на $k$ аналогичных фолдов и обучается модель второго уровня $f' : d'_k \mapsto \{0, 1\}$.

В данной работе в качестве модели второго уровня используется логистическая регрессия. Качество базовых моделей и его стандартное отклонение, посчитанное по $k$ фолдам приведено в таблице 1. На Рис. 6 показана матрица корелляций базовых моделей. Видно, что статистически подходы сильно отличаются от нейросетевых, при том что они дают примерно одинаковое качество. Это хорошо, потому что увеличивает разнообразность ансамбля. В таблице 2 приведено качество итогового решения. На Рис. 7 показаны модули коэффициентов логистической регрессии, что можно интерпретировать как важность базовых алгоритмов в ансамбле.

Таблица 1: Качество моделей

| | модель1 | модель2 | модель3 | модель4 | Ансамбль |
|---|---|---|---|---|---|
| Accuracy | 0.726 | 0.669 | 0.638 | 0.661 | 0.730 |
| std | 0.013 | 0.016 | 0.001 | 0.021 | 0.010 |

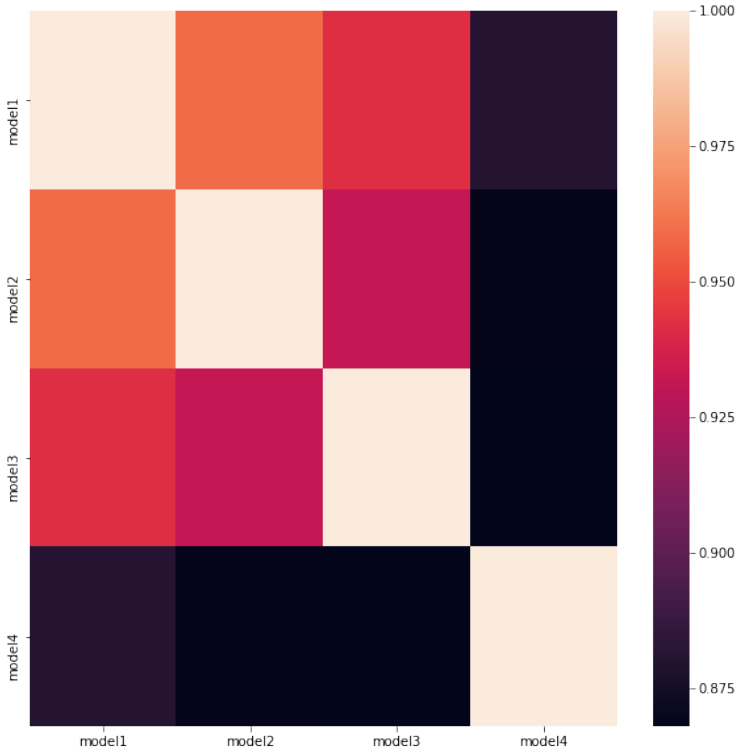


Рис. 6: Матрица корелляций базовых моделей

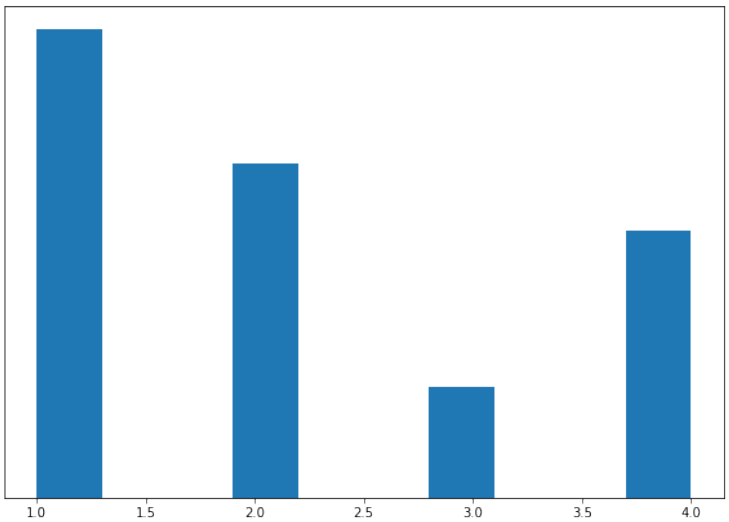


Рис. 7: Важности базовых моделей в ансамбле

## Список литературы

[1] S. W. P. G. D. o. C. S. C. Daria Sorokina, Johannes Gehrke, I. N. U. Information Science, Department of Physics Cornell University, Plagiarism detection, arXiv.

URL https://arxiv.org/pdf/cs/0702012.pdf
[2] V. S. R. Sobha Lalitha Devi, Pattabhi R K Rao, A. Akilandeswari, External plagiarism detection, CLEF 2010.
URL http://ceur-ws.org/Vol-1176/CLEF2010wn-PAN-DeviEt2010.pdf
[3] H. A. Chowdhury, D. K. Bhattacharyya, Plagiarism: Taxonomy, tools and detection techniques, arXiv.
URL https://arxiv.org/pdf/1801.06323.pdf
[4] Pan 2011.
URL http://pan.webis.de/clef11/pan11-web/
[5] H. R. I. Abdul Sittar, R. M. A. Nawab, Author diarization using cluster-distance approach, CLEF 2016.
URL https://pdfs.semanticscholar.org/1158/6f70a6bf976bf3bfc56c51c4e13e3fdd0168.pdf
[6] Pan 2016.
URL http://pan.webis.de/clef16/pan16-web/
[7] M. Kuznetsov, A. Motrenko, R. Kuznetsova, V. Strijov, Methods for Intrinsic Plagiarism Detection and Author Diarization—Notebook for PAN at CLEF 2016, in: K. Balog, L. Cappellato, N. Ferro, C. Macdonald (Eds.), CLEF 2016 Evaluation Labs and Workshop – Working Notes Papers, 5-8 September, Évora, Portugal, CEUR-WS.org, 2016.
URL http://ceur-ws.org/Vol-1609/
[8] Pan 2018.
URL http://pan.webis.de/clef18/pan18-web/
[9] D. Karaś, M. Śpiewak, P. Sobecki, OPI-JSA at CLEF 2017: Author Clustering and Style Breach Detection— Notebook for PAN at CLEF 2017.
URL http://ceur-ws.org/Vol-1866/
[10] Pst tagbank penn treebank.
URL https://www.ling.upenn.edu/courses/Fall$_2$003/$ling$001/$penn_treebank_pos.html$
[11] J. Khan, Style Breach Detection: An Unsupervised Detection Model—Notebook for PAN at CLEF 2017, in: L. Cappellato, N. Ferro, L. Goeuriot, T. Mandl (Eds.), CLEF 2017 Evaluation Labs and Workshop – Working Notes Papers, 11-14 September, Dublin, Ireland, CEUR-WS.org, 2017.
URL http://ceur-ws.org/Vol-1866/
[12] K. Safin, R. Kuznetsova, Style Breach Detection with Neural Sentence Embeddings—Notebook for PAN at CLEF 2017, in: L. Cappellato, N. Ferro, L. Goeuriot, T. Mandl (Eds.), CLEF 2017 Evaluation Labs and Workshop – Working Notes Papers, 11-14 September, Dublin, Ireland, CEUR-WS.org, 2017.
URL http://ceur-ws.org/Vol-1866/
[13] R. K. Kamil Safin, Style Breach Detection with Neural Sentence Embeddings, CLEF 2017, 2017.
URL https://pdfs.semanticscholar.org/c70e/7f8fbc561520accda7eea2f9bbf254edb255.pdf
[14] Гущин, Методы ансамблирования обучающихся алгоритмов, 2015.
URL http://www.machinelearning.ru/wiki/images/5/56/Guschin2015Stacking.pdf
[15] J. Ramos, Using tf-idf to determine word relevance in document queries, 2002.
URL http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.121.1424rep=rep1type=pdf
[16] G. C. J. D. Tomas Mikolov, Kai Chen, Efficient Estimation of Word Representations in Vector Space, 2013.
URL https://arxiv.org/abs/1301.3781
[17] Y. L. Xiang Zhang, Junbo Zhao, Character-level convolutional networks for text classification, 2015.
URL https://ru.sharelatex.com/project/5adc83a9405d7136e106ebfa
[18] Stackexchange network.
URL https://github.com/Harrix/Math-Harrix-Library