

## 1. Метрики для текстов с помощью библиотеки ruTS

### а. Основные статистики

Метрики были рассчитаны на предыдущем этапе для 9 авторов, где в каждом из произведений одна глава считалась отдельным объектом. В результате расчета метрик для каждого из авторов вышло следующее число глав:



```
df['author'].value_counts()
```

```
lev-tolstoi          747
ivan-turgenev        333
fedor-dostoevskii    279
nikolai-gogol        206
anton-chekhov        186
maksim-gorkii        100
dmitriy-mamin-sibiryak  96
ivan-goncharov        57
sergey-aksakov       32
Name: author, dtype: int64
```

Но если мы затем посмотрим на остальные статистики отдельных глав, то становится понятно, что в перечень глав также попали аннотации, небольшие вступления и другие подобные части текста, в которых очень мало предложений и которые сложно считать отдельным объектом:

	n_sents	n_words	n_unique_words	n_long_words	n_complex_words	n_simple_words	n_monosyllable_words	n_polysyllable_words	n_chars	n_letters
count	2036.000000	2036.000000	2036.000000	2036.000000	2036.000000	2036.000000	2036.000000	2036.000000	2.036000e+03	2036.000000
unique	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
top	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
freq	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
mean	129.905697	2319.201375	943.064342	959.969057	326.750982	1851.349705	700.290766	1477.809921	1.463957e+04	11514.990177
std	327.975848	6608.464037	1558.548883	3000.992001	1036.824003	5272.056270	1824.574326	4496.687108	4.370152e+04	34642.677192
min	1.000000	1.000000	1.000000	1.000000	0.000000	1.000000	0.000000	1.000000	7.000000e+00	7.000000
25%	32.000000	614.250000	375.000000	255.750000	83.000000	479.000000	170.000000	391.750000	3.905250e+03	3089.500000
50%	63.000000	1131.500000	617.500000	476.000000	161.000000	890.000000	325.500000	720.500000	7.146000e+03	5595.500000
75%	126.000000	2001.000000	994.000000	845.000000	291.000000	1591.500000	619.500000	1278.250000	1.256675e+04	9976.250000
max	7286.000000	170197.000000	33087.000000	80175.000000	27654.000000	135233.000000	44097.000000	118790.000000	1.147321e+06	911997.000000

Посмотрим на названия глав, в которых нет хотя бы 5 предложений:



```
print(df[df['n_sents'] < 5].index)
```

```
Index(['vishnevyyi-sad. Text/chapter1.xhtml', 'chaika. Text/chapter1.xhtml',  
      'tri-sestry. Text/chapter1.xhtml', 'ty-i-vy. Text/annotation.xhtml',  
      's-zhenoi-possorilsya. Text/annotation.xhtml',  
      'protekciya. Text/annotation.xhtml', 'sovet. Text/annotation.xhtml',  
      'zhilec. Text/annotation.xhtml', 'grisha. Text/annotation.xhtml',  
      'tif. Text/annotation.xhtml', 'radost. Text/annotation.xhtml',  
      'ushla. Text/annotation.xhtml',  
      'ekzamen-na-chin. Text/annotation.xhtml',  
      'baran-i-baryshnya. Text/annotation.xhtml',  
      'dyadya-vanya. Text/chapter1.xhtml',  
      'aptekarsha. Text/annotation.xhtml', 'bez-mesta. Text/annotation.xhtml',  
      'bezzakonie. Text/annotation.xhtml',  
      'bezotcovshina. Text/chapter1.xhtml',  
      'beseda-pyanogo-s-trezvym-chertom. Text/annotation.xhtml',  
      'blagodarnyi. Text/annotation.xhtml', 'v-apteke. Text/annotation.xhtml',  
      'v-gostinoi. Text/annotation.xhtml', 'v-sarae. Text/annotation.xhtml',  
      'vyigryshnyi-bilet. Text/annotation.xhtml',  
      'stikhotvoreniya-v-proze. Text/chapter1.xhtml',  
      'stikhotvoreniya-v-proze. Text/chapter52.xhtml',  
      'stikhotvoreniya-v-proze. Text/chapter67.xhtml',  
      'stikhotvoreniya-v-proze. Text/chapter68.xhtml',  
      'stikhotvoreniya-v-proze. Text/chapter73.xhtml',  
      'stikhotvoreniya-v-proze. Text/chapter74.xhtml',  
      'stikhotvoreniya-v-proze. Text/chapter76.xhtml',  
      'stikhotvoreniya-v-proze. Text/chapter77.xhtml',  
      'stikhotvoreniya-v-proze. Text/chapter81.xhtml',  
      'bezhin-lug. Text/annotation.xhtml', 'voskresenie. Text/chapter2.xhtml',  
      'na-dne. Text/chapter1.xhtml',  
      'pesnya-o-burevestnike. Text/annotation.xhtml',  
      'foma-gordeev. Text/annotation.xhtml',  
      'nesvoevremennye-mysli-zametki-o-revolyucii-i-kulture. Text/chapter12.xhtml',  
      'nesvoevremennye-mysli-zametki-o-revolyucii-i-kulture. Text/chapter32.xhtml',  
      'nesvoevremennye-mysli-zametki-o-revolyucii-i-kulture. Text/chapter33.xhtml',  
      'nesvoevremennye-mysli-zametki-o-revolyucii-i-kulture. Text/chapter37.xhtml',  
      'nesvoevremennye-mysli-zametki-o-revolyucii-i-kulture. Text/chapter39.xhtml',  
      'nesvoevremennye-mysli-zametki-o-revolyucii-i-kulture. Text/chapter43.xhtml',  
      'nesvoevremennye-mysli-zametki-o-revolyucii-i-kulture. Text/chapter49.xhtml',  
      'nesvoevremennye-mysli-zametki-o-revolyucii-i-kulture. Text/annotation.xhtml',  
      'vechera-na-khutore-bliz-dikanki. Text/chapter12.xhtml',  
      'vechera-na-khutore-bliz-dikanki. Text/chapter18.xhtml',  
      'revizor. Text/chapter1.xhtml', 'mirgorod. Text/chapter1.xhtml',  
      'zhenitba. Text/chapter1.xhtml', 'shinel. Text/annotation.xhtml'],  
      dtype='object', name='Unnamed: 0')
```

В основном это аннотации и первые главы текстов. Например, для “Детские годы Багрова-внука” в этот список попали такая часть текста:

## К читателям

Внучке моей  
Ольге Григорьевне  
Аксаковой

Я написал отрывки из "Семейной хроники"<sup>1</sup> по рассказам семейства гг. Багровых, как известно моим благосклонным читателям. В эпилоге к пятому и последнему отрывку я простился с описанными мною личностями, не думая, чтобы мне когда-нибудь привелось говорить о них. Но человек часто думает ошибочно: внук Степана Михайлыча Багрова рассказал мне с большими подробностями историю своих детских годов; я записал его рассказы с возможною точностью, а как они служат продолжением "Семейной хроники", так счастливо обратившей на себя внимание читающей публики, и как рассказы эти представляют довольно полную историю дитяти, жизнь человека в детстве, детский мир, создающийся постепенно под влиянием ежедневных, новых впечатлений, — то я решился напечатать записанные мною рассказы. Желая, по возможности, передать живость изустного повествования, я везде говорю прямо от лица рассказчика. Прежние лица "Хроники" выходят опять на сцену, а старшие, то есть дедушка и бабушка, в продолжение рассказа оставляют ее навсегда... Снова поручаю моих Багровых благосклонному вниманию читателей.

С. Аксаков

## Сноски к главе

1—"Семейная хроника" С.Т. Аксакова вышла из печати в 1856 году, за два года до того, как вышли в свет "Детские годы Багрова-внука".

Но также в этот список попали и обычные главы, которые либо просто сами по себе очень маленькие (например, "Стихотворение в прозе" И. Тургенева), либо в них используются сложные предложения (например, в произведениях Л. Толстого). Поэтому мы не можем просто взять и удалить все такие объекты, но как минимум можем убрать сейчас все объекты, в названии которых содержится слово "annotation". Затем уберем еще объекты, где нет хотя бы 3 предложений, чтобы исключить различные объекты вроде обращения к читателю.

Тогда число объектов по авторам и основные статистики выглядят так:

```
df['author'].value_counts()
```

```
lev-tolstoi          732
ivan-turgenev        315
fedor-dostoevskii    269
nikolai-gogol         191
anton-chekhov         103
dmitriy-mamin-sibiryak  93
maksim-gorkii         88
ivan-goncharov        55
sergey-aksakov        29
Name: author, dtype: int64
```

```
df.describe()
```

	n_sents	n_words	n_unique_words	n_long_words	n_complex_words	n_simple_words	n_monosyllable_words	n_polysyllable_words	n_chars	n_letters
count	1875.000000	1875.000000	1875.000000	1875.000000	1875.000000	1875.000000	1875.000000	1875.000000	1.875000e+03	1875.000000
mean	140.588267	2507.511467	1015.544000	1036.722667	352.260267	2003.624533	758.529067	1597.355733	1.581910e+04	12443.274133
std	339.654549	6853.813849	1603.477645	3115.294931	1076.617946	5467.077042	1890.011633	4666.518751	4.534625e+04	35948.672546
min	4.000000	27.000000	19.000000	13.000000	3.000000	19.000000	5.000000	16.000000	1.770000e+02	139.000000
25%	38.000000	748.500000	443.000000	315.000000	103.000000	581.000000	210.000000	480.000000	4.772000e+03	3746.000000
50%	69.000000	1215.000000	658.000000	517.000000	175.000000	947.000000	349.000000	776.000000	7.753000e+03	6073.000000
75%	135.000000	2229.500000	1075.500000	915.000000	313.000000	1773.500000	673.500000	1409.000000	1.403000e+04	10951.000000
max	7286.000000	170197.000000	33087.000000	80175.000000	27654.000000	135233.000000	44097.000000	118790.000000	1.147321e+06	911997.000000

8 rows × 11 columns

Общее число объектов сократилось с 2036 до 1875, а также судя по числу слов и предложений удалось убрать совсем маленькие части текста.

#### b. Пропуски

Основные статистики вроде числа букв, предложений, символов и т.д. есть у каждого объекта, но для показателей женского/мужского/нейтрального родов, множественного числа, настоящего/прошедшего/будущего времен, активного/пассивного залогов, первого/второго/третьего лиц есть пропуски, которые означают, что это не используется в тексте. То есть такие пропуски равнозначны нулям, поэтому ими их и заполним.

Больше всего на данной выборке не были найдены 2-ое лицо и активный залог.

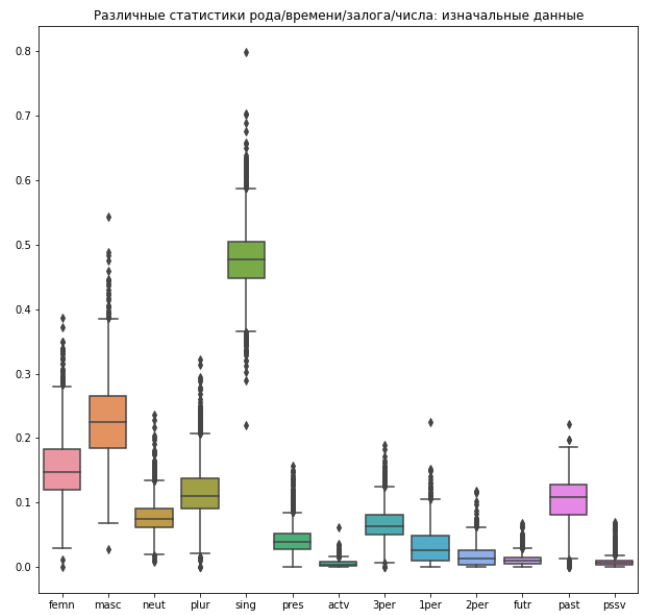
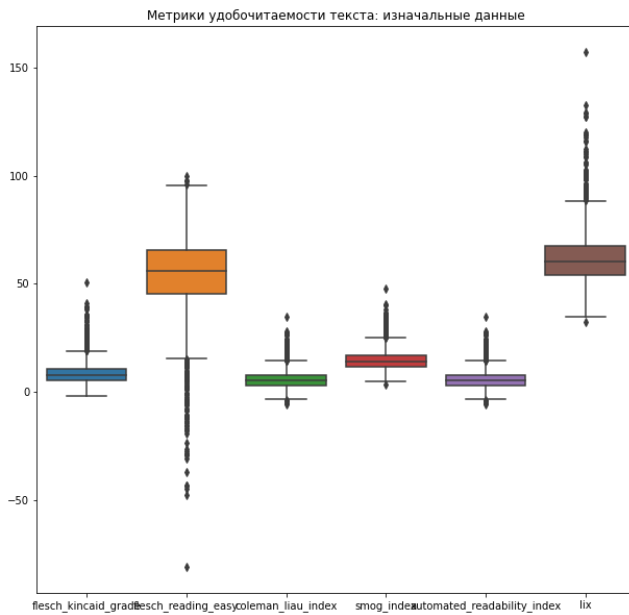
femn	0.000491
masc	0.000000
neut	0.000000
plur	0.001473
sing	0.000000
pres	0.001473
actv	0.129175
3per	0.001965
ttr	0.000000
rttr	0.000000
cttr	0.000000
httr	0.000000
sttr	0.000000
mttr	0.000000
dttr	0.000000
mattr	0.000000
msttr	0.000000
mtld	0.000000
mamtld	0.000000
hdd	0.000000
simpson_index	0.000000
hapax_index	0.000000
author	0.000000
1per	0.090373
2per	0.160118
futr	0.095285
past	0.003929
pssv	0.052554

#### c. Выбросы и аномальные значения

Изучим боксплоты для оставшихся объектов.

По графикам видно, что в выборке довольно много выбросов, поэтому посмотрим, что именно это за объекты.

- Рассмотрим удобочитаемость на примере метрики `flesh_reading_easy`: чем больше показатель, тем легче текст:



```

lower_bound = df['flesch_reading_easy'].quantile(q=0.025)
upper_bound = df['flesch_reading_easy'].quantile(q=0.975)

to_easy = df[df['flesch_reading_easy'] > upper_bound]
to_hard = df[df['flesch_reading_easy'] < lower_bound]

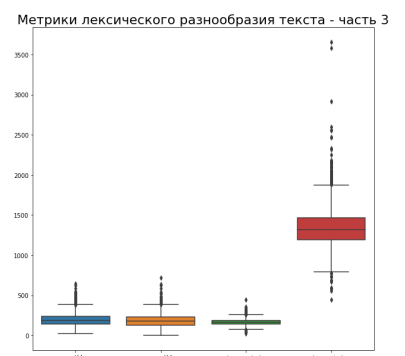
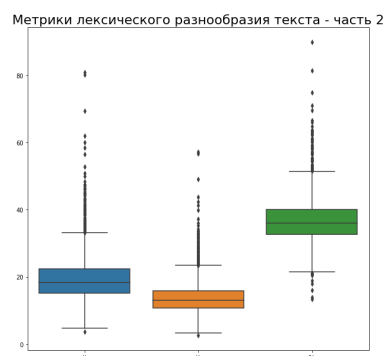
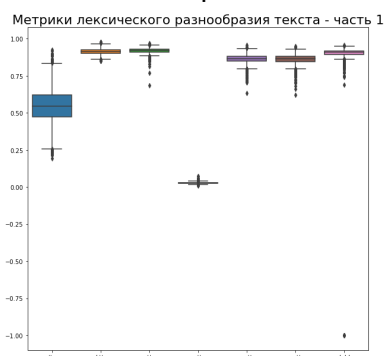
print(set(to_easy.index[:16]))
print(set(to_hard.index[:16]))

{'stikhotvoreniya-v-proze. Text/chapter4.xhtml', 'zhivoi-trup. Text/chapter9.xhtml', 'voina-i-mir. Text/chapter26.xhtml', 'otrochestvo. Text/chapter10.xhtml', 'kreicero
{'voina-i-mir. Text/chapter339.xhtml', 'nesvoevremennye-mysli-zametki-o-revolyucii-i-kulture. Text/chapter34.xhtml', 'nesvoevremennye-mysli-zametki-o-revolyucii-i-kultu

```

“Война и мир” попала как в нижний, так и в верхний квантиль по показателю сложности текста, а несколько глав из “Несвоевременные мысли заметки о революции и культуре” оказались в квантиле сложных текстов. Остальные метрики удобочитаемости текста в основном повторяют результаты друг друга, поэтому выбросы там те же самые.

- Выбросы для показателей времени/залога/рода и т.д. было найдено слишком много или слишком мало.
- Метрики разнообразия текста, как и метрики удобочитаемости, тоже очень похожи между собой, поэтому можно взять выборочно один показатель и так же на него посмотреть.



```

lower_bound = df['mttr'].quantile(q=0.025)
upper_bound = df['mttr'].quantile(q=0.975)

to_diverse = df[df['mttr'] > upper_bound]
to_similar = df[df['mttr'] < lower_bound]

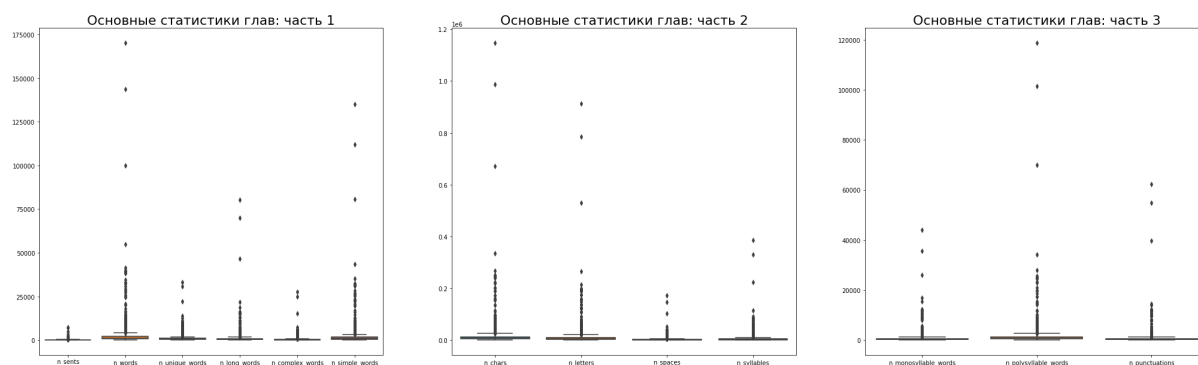
print(set(to_diverse.index[:16]))
print(set(to_similar.index[:16]))

{'zhivoi-trup. Text/chapter12.xhtml', 'voyna-i-mir. Text/chapter169.xhtml', 'kreicerova-sonata. Text/chapter2.xhtml', 'revizor. Text/chapter55.xhtml', 'r
{'nesvoevremennye-mysli-zametki-o-revolycii-i-kulture. Text/chapter24.xhtml', 'stikhotvoreniya-v-proze. Text/chapter50.xhtml', 'stikhotvoreniya-v-proze.

```

Слишком однообразные главы оказались у, например, “Стихотворения в прозе” или “Несвоевременные мысли заметки о революции и культуре”, а слишком разнообразие - “Война и мир” или “Ревизор”.

- Если посмотреть на основные статистики текста, то можно увидеть, что есть слишком большие главы:



Слишком длинные главы получились, например, в “Преступлении и наказании”, а слишком короткие - в “Стихотворениях в прозе”:

```

lower_bound = df['n_words'].quantile(q=0.025)
upper_bound = df['n_words'].quantile(q=0.975)

to_big = df[df['n_words'] > upper_bound]
to_small = df[df['n_words'] < lower_bound]

print(set(to_big.index[:16]))
print(set(to_small.index[:16]))

{'bezotcovshina. Text/chapter2.xhtml', 'besy. Text/chapter8.xhtml', 'prestuplenie-i-nakazanie. Text/chapter3.xhtml', 'prestuplenie-i-nakazanie. Text/chapter6.xhtml',
{'stikhotvoreniya-v-proze. Text/chapter61.xhtml', 'stikhotvoreniya-v-proze. Text/chapter41.xhtml', 'stikhotvoreniya-v-proze. Text/chapter73.xhtml', 'zhivoi-trup.

```

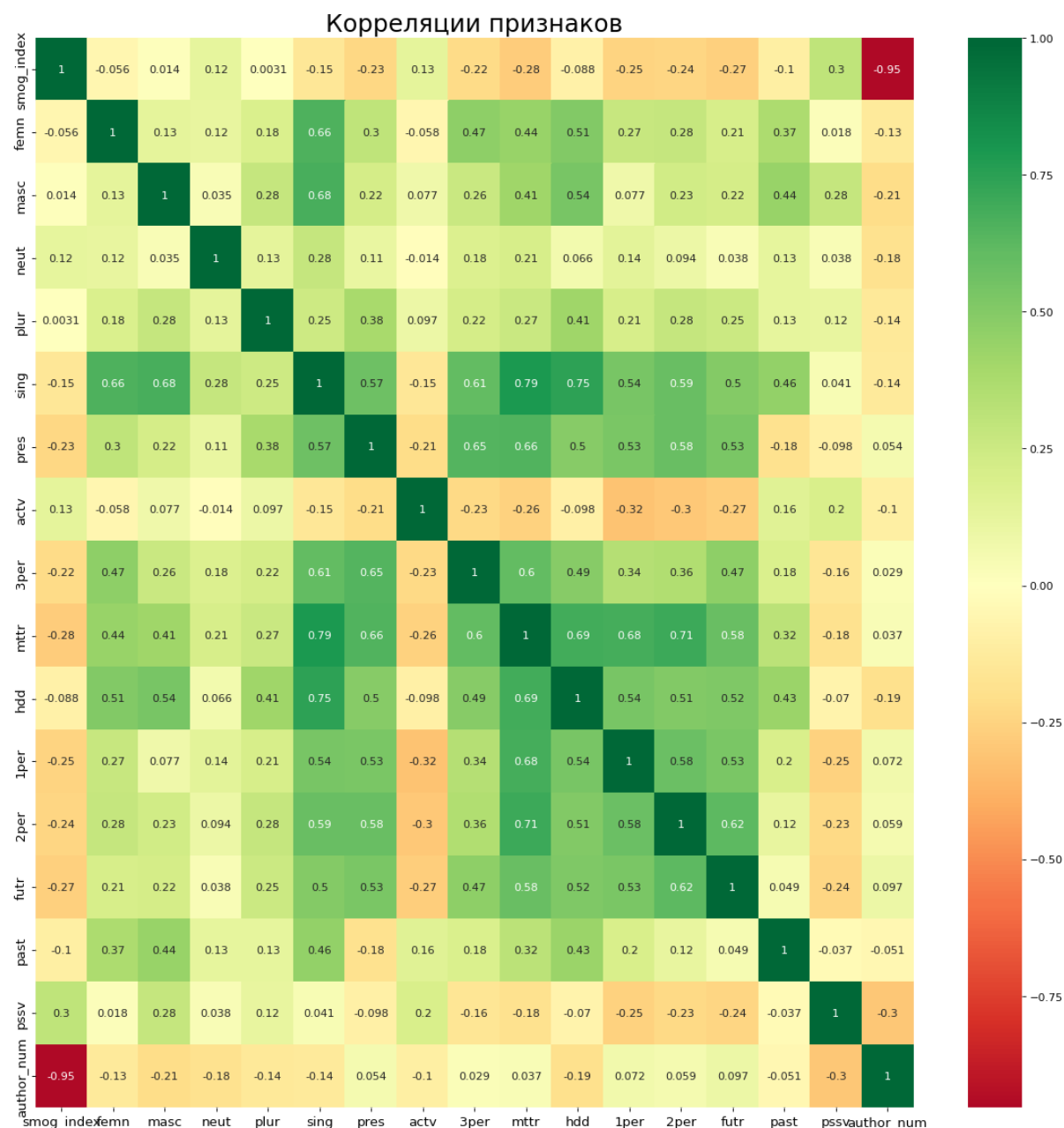
#### d. Корреляции признаков

По определению метрик удобочитаемости и по их корреляциям, нет смысла использовать сразу все, так как показатели внутри каждой из групп очень скоррелированы. Поэтому в итоге оставим только те, которые наиболее устойчивы к длине текста по определению:

- Можно оставить только **smog\_index** для удобочитаемости. Чем больше значение этой метрики, тем сложнее текст.
- Для лексического разнообразия:
  - **hdd** Наиболее достоверная реализация алгоритма VocD (2010, McCarthy & Jarvis). В основе алгоритм лежит метод случайного отбора из текста сегментов длиной от 32 до 50 слов и вычисления для них TTR с последующим усреднением.

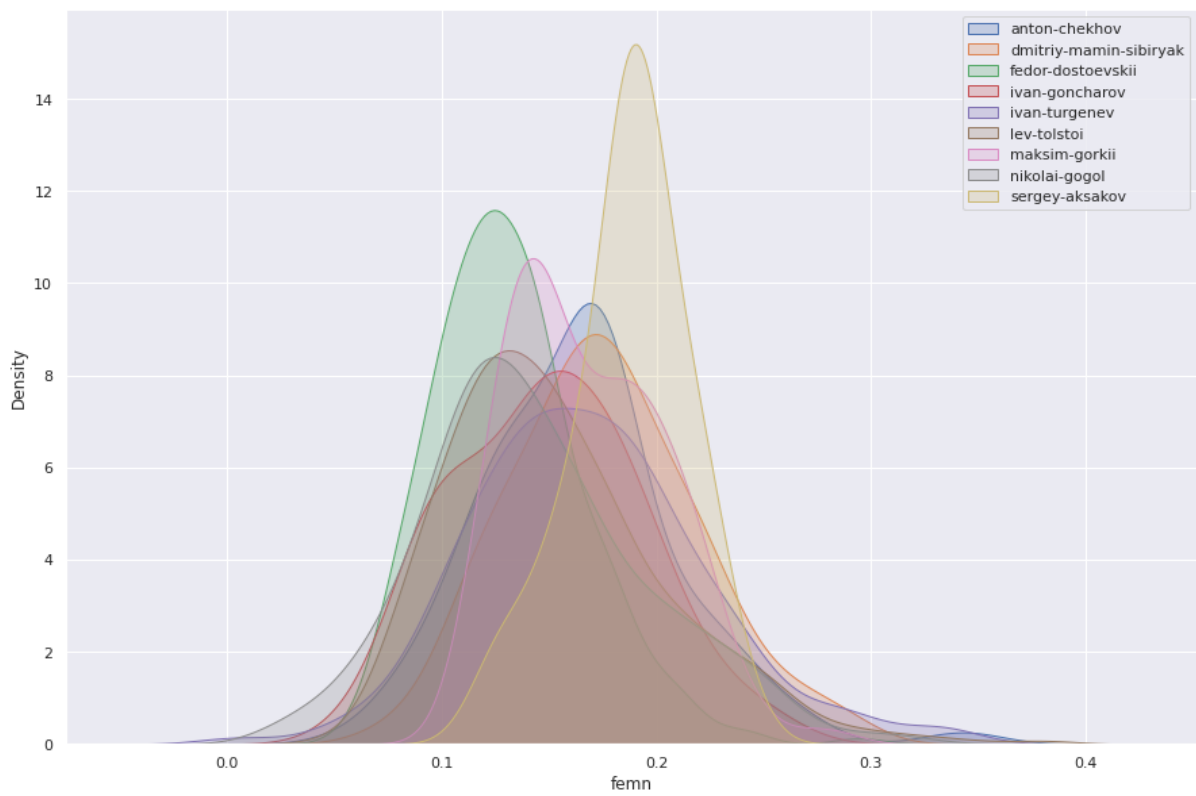
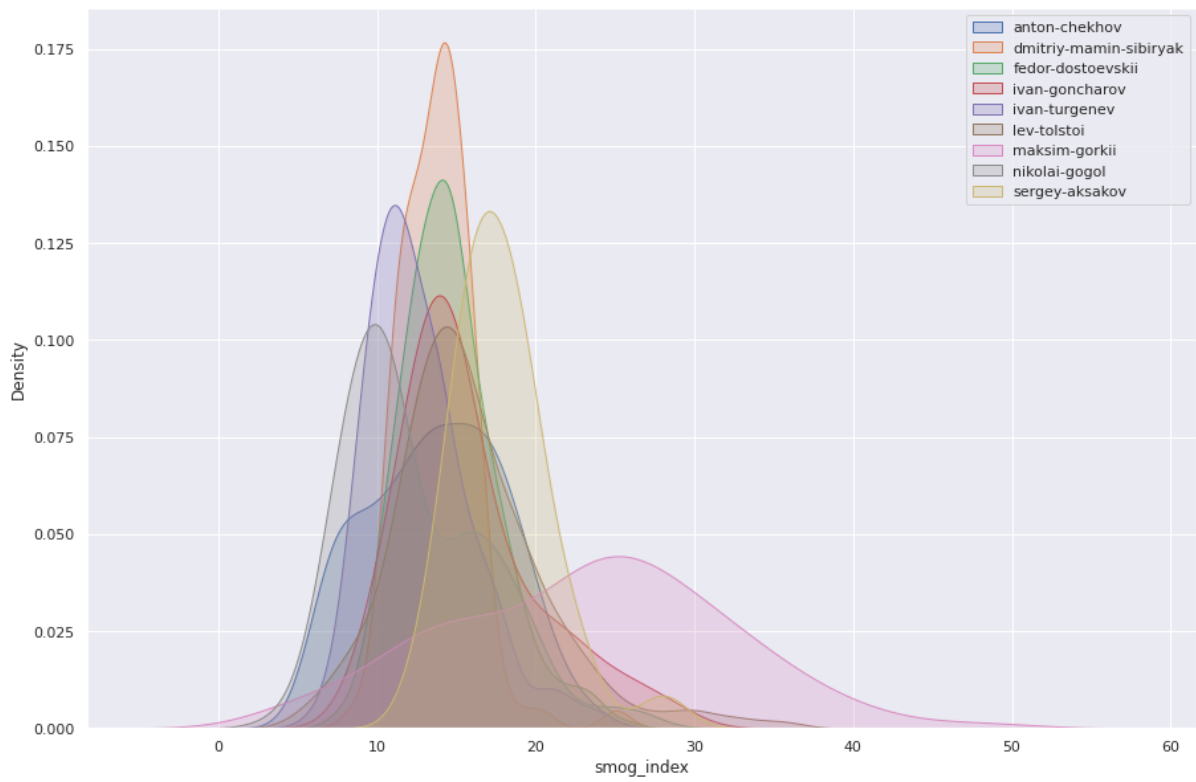
- **mttr** Модификация метрики TTR с использованием логарифмической функции (1966, Mass). Наиболее стабильная метрика в отношении длины текста.

Все метрики лица/рода/времени и других из этой группы мы делим на число слов в главе. Переменную автора временно закодируем с помощью OrdinalEncoder, но потом надо будет подобрать более подходящий способ кодировки для данной задачи. В результате после нормировки столбцов корреляции выглядят так:

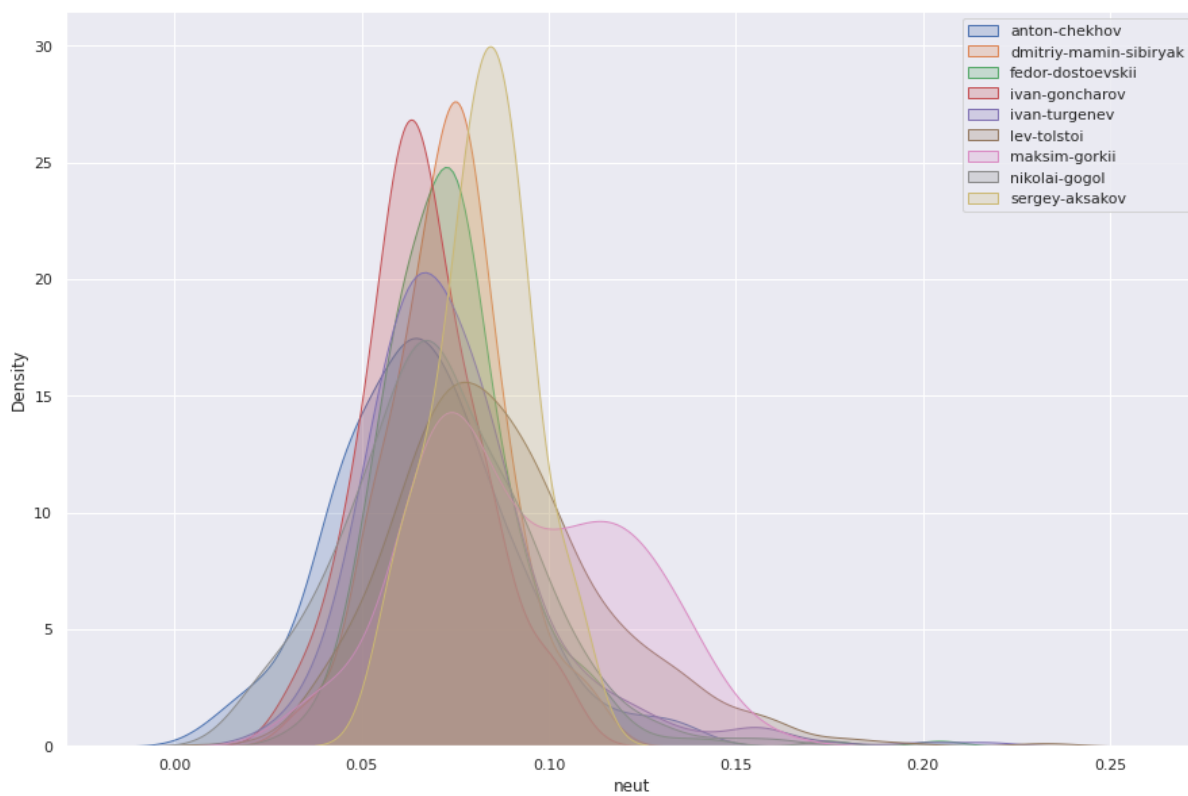
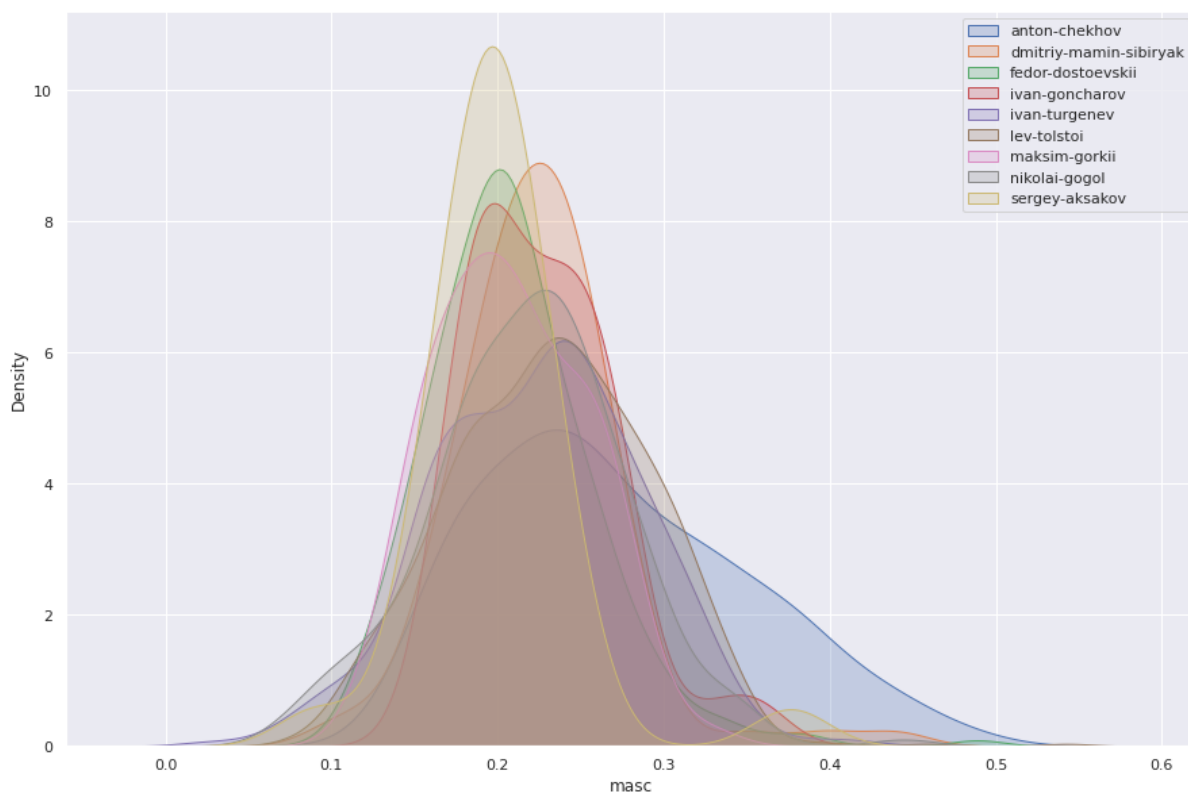


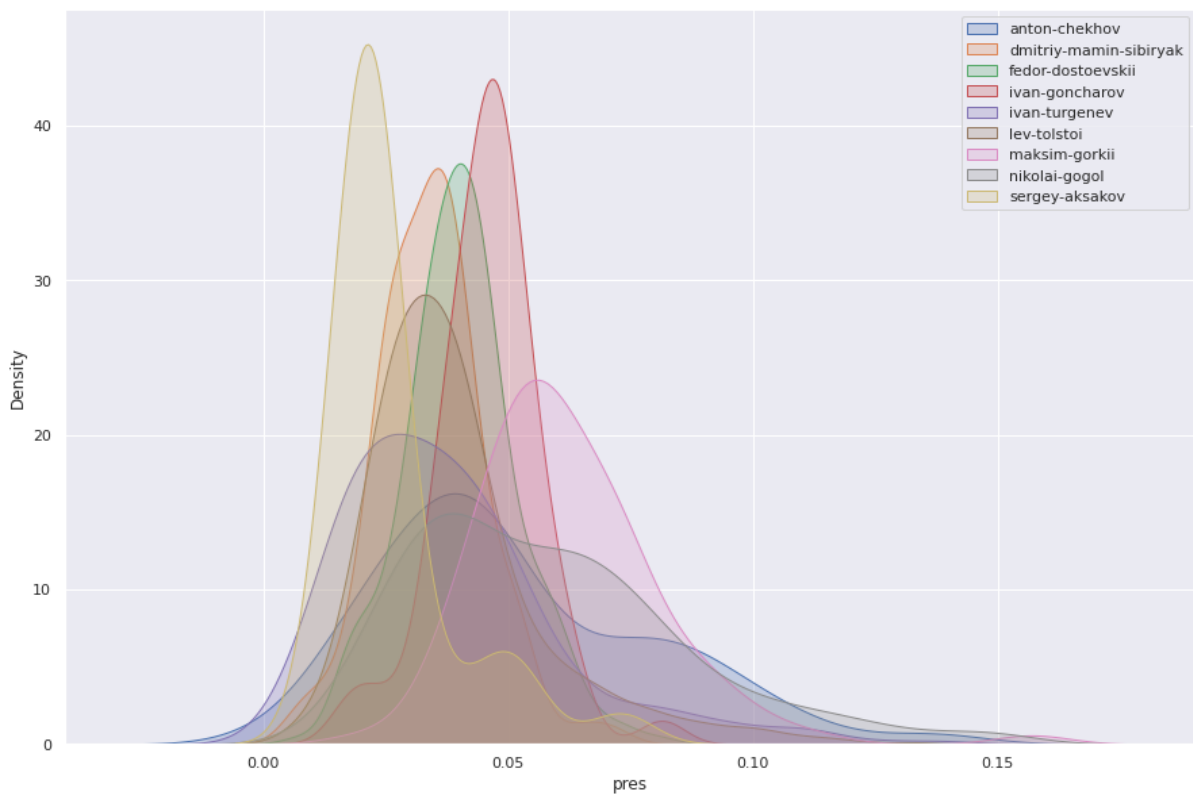
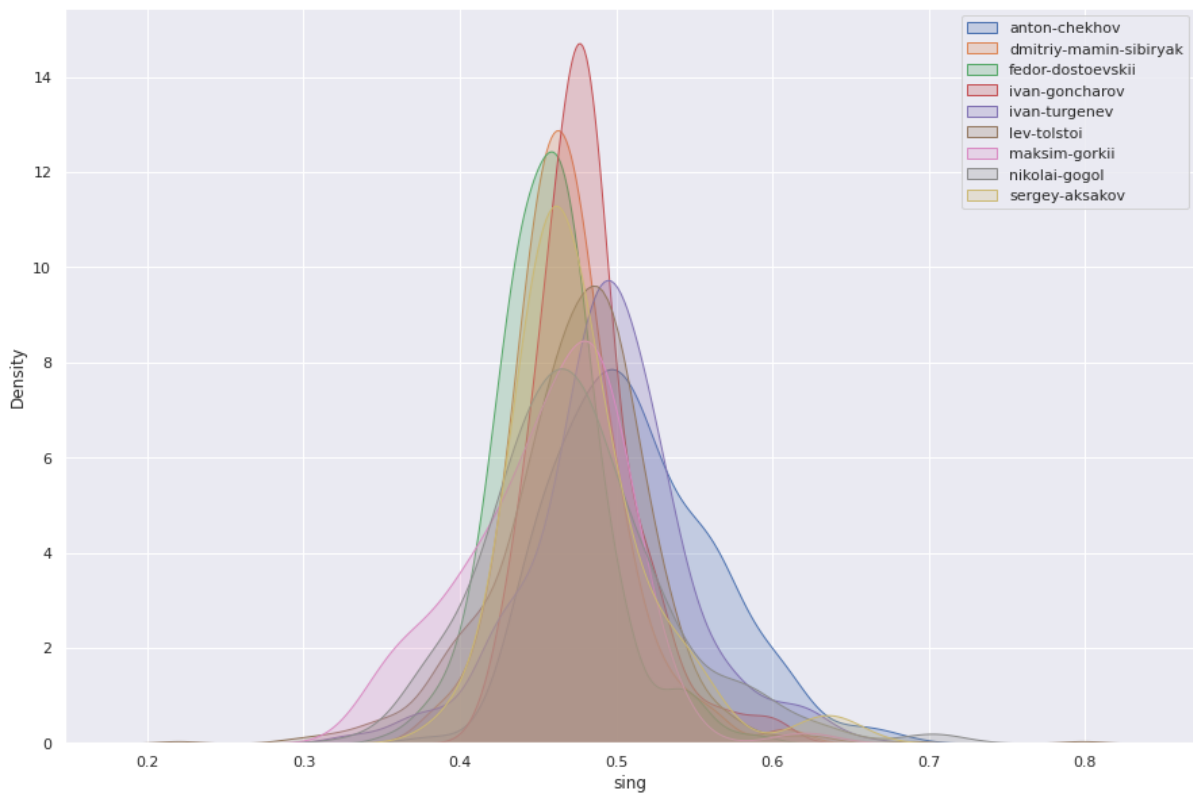
Между метриками есть интересные корреляции. Например, на разнообразие текста положительно влияет использование единственного числа, третьего лица и настоящего времени, а активный залог делает текст менее разнообразным.

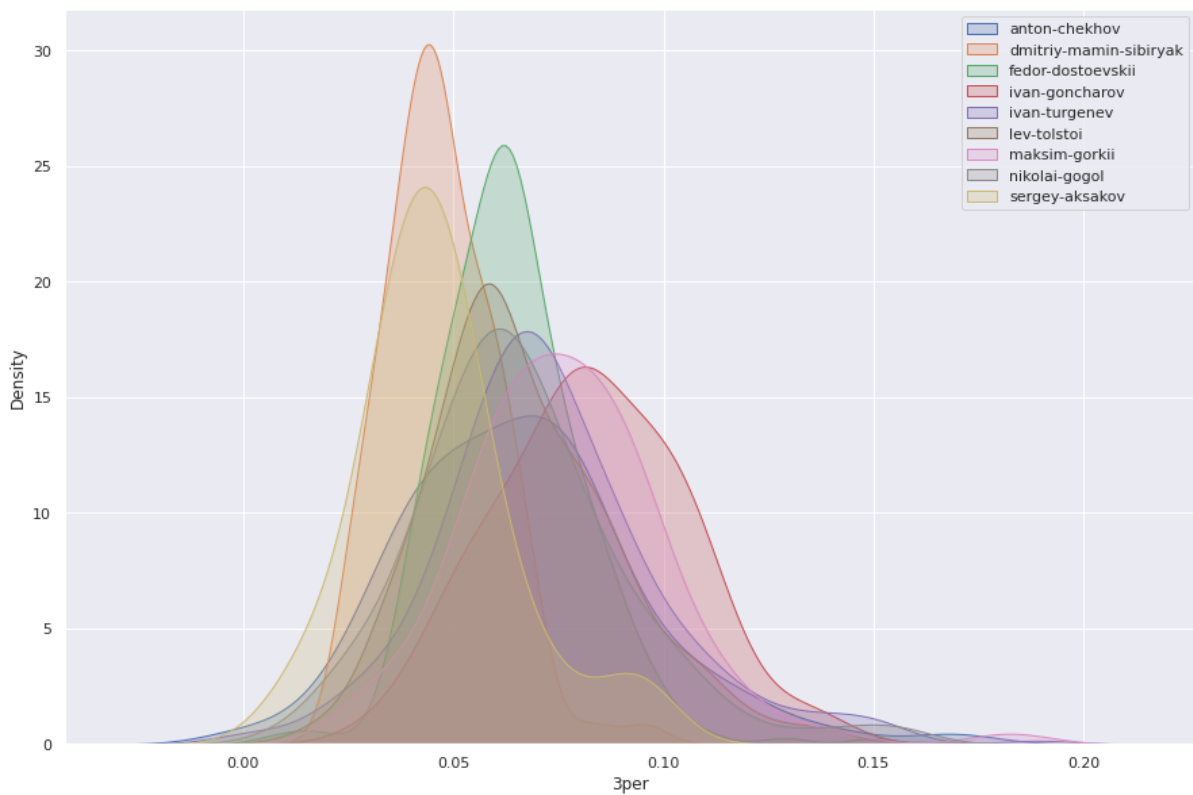
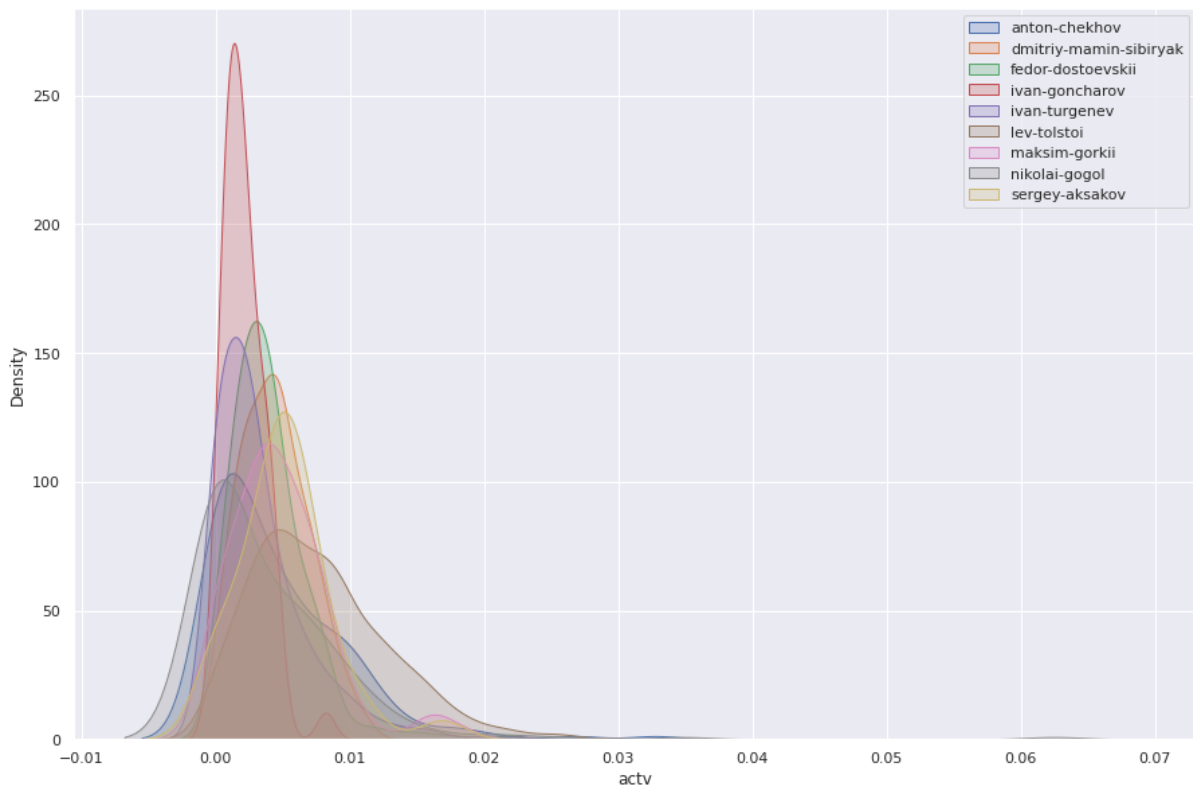
#### е. Поведение признаков

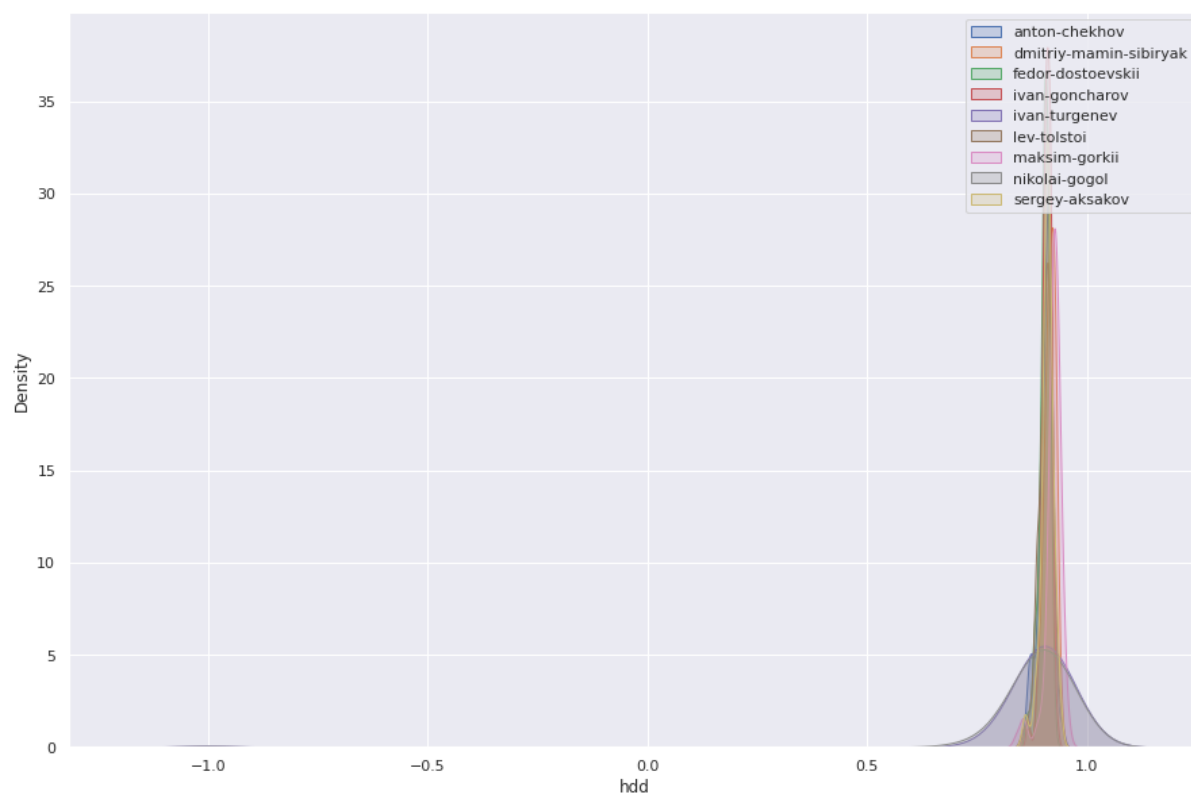
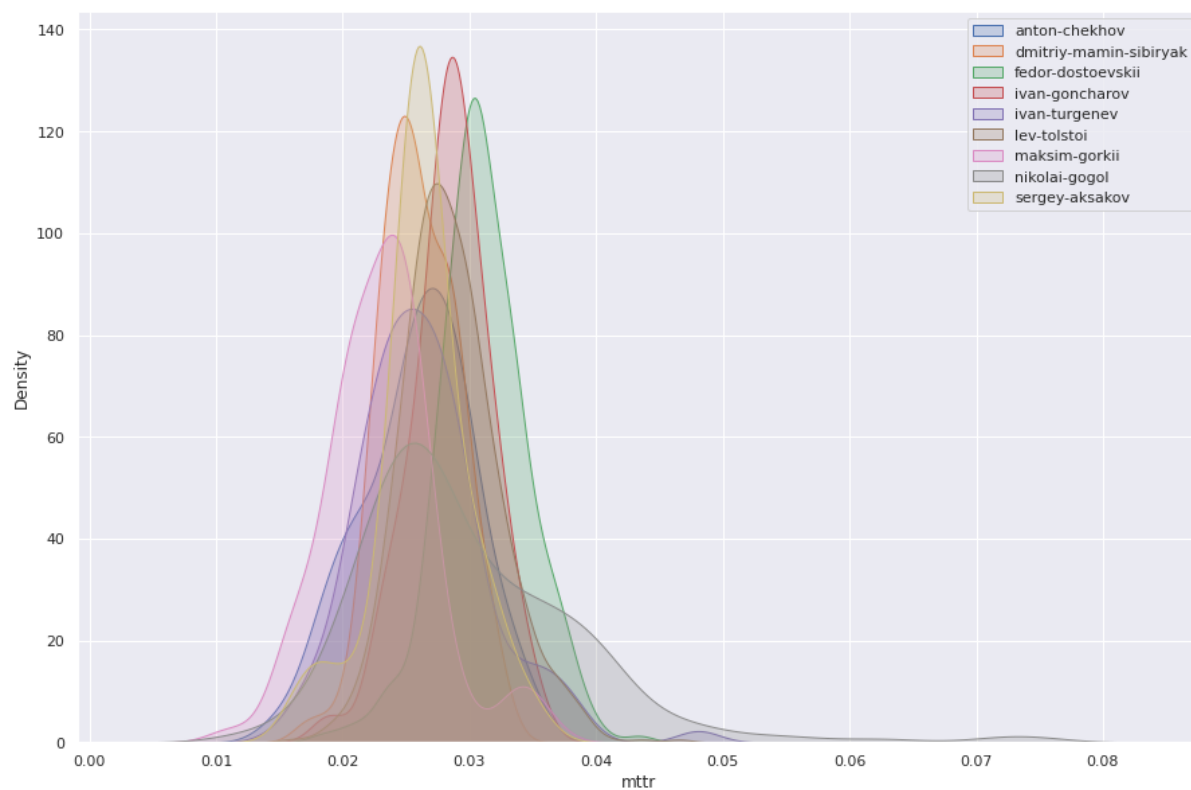


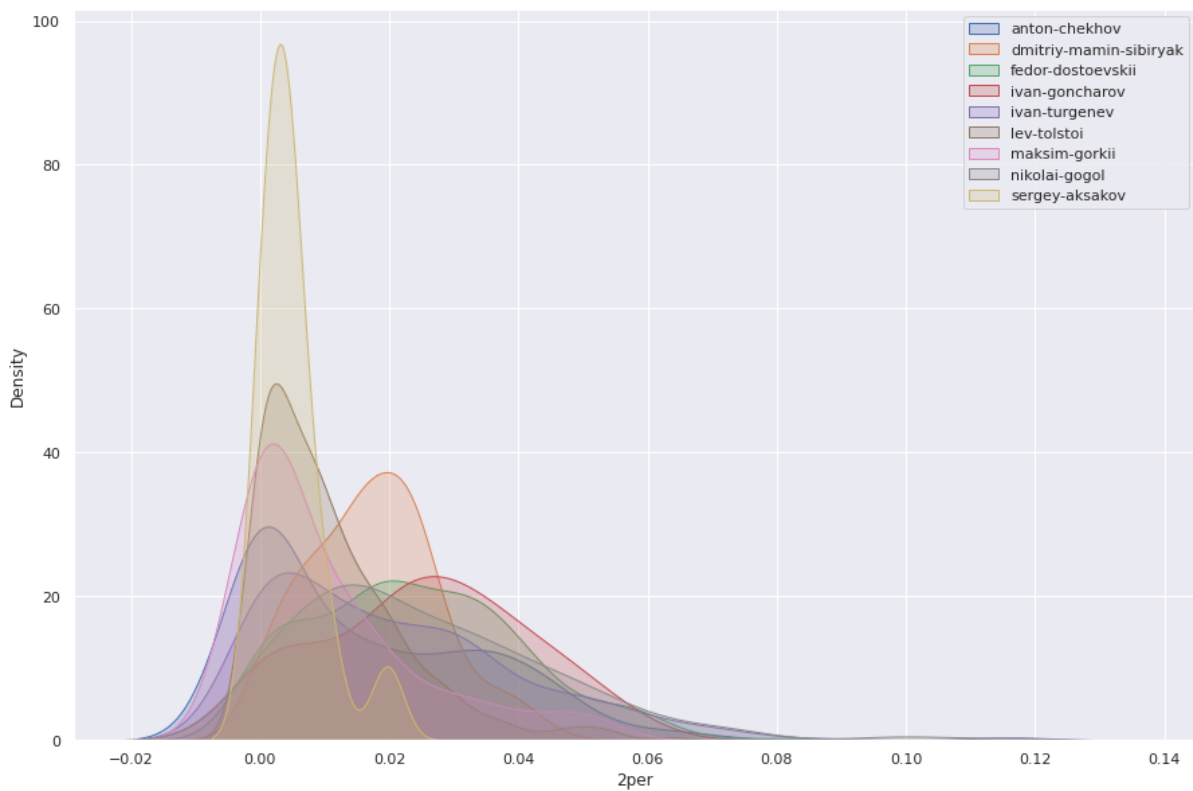
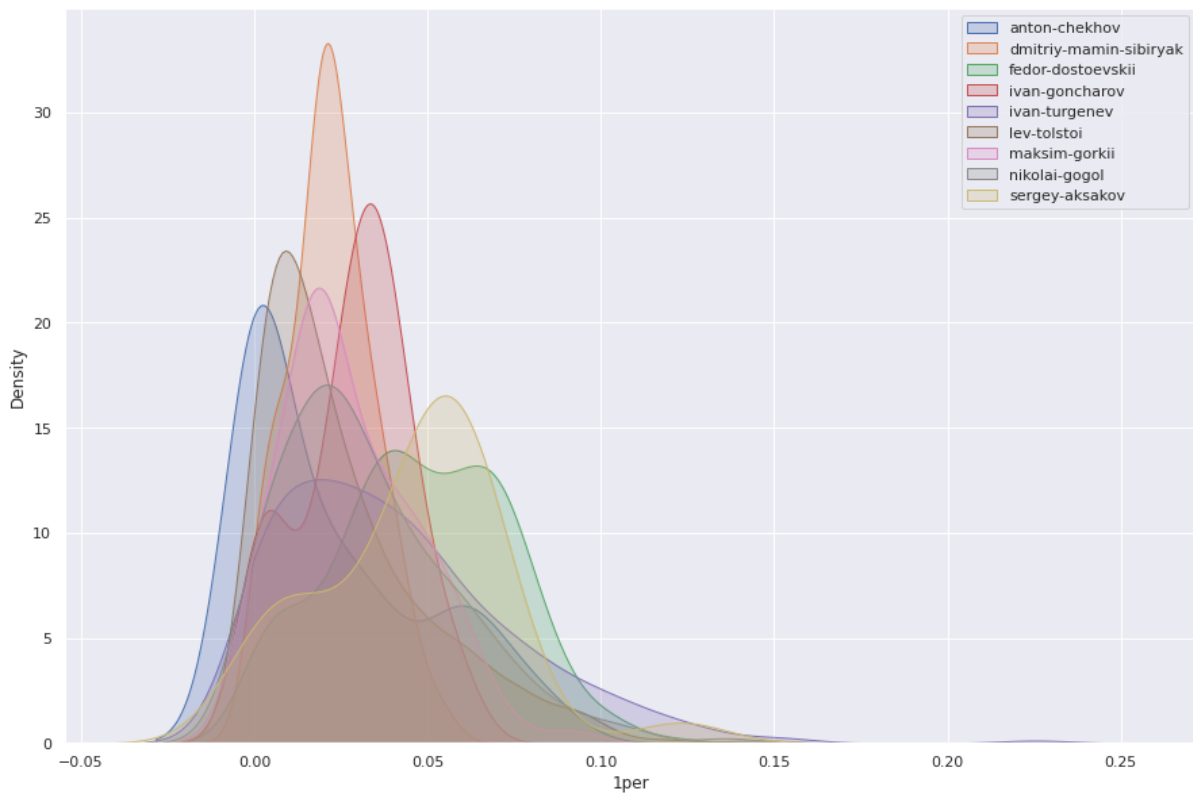


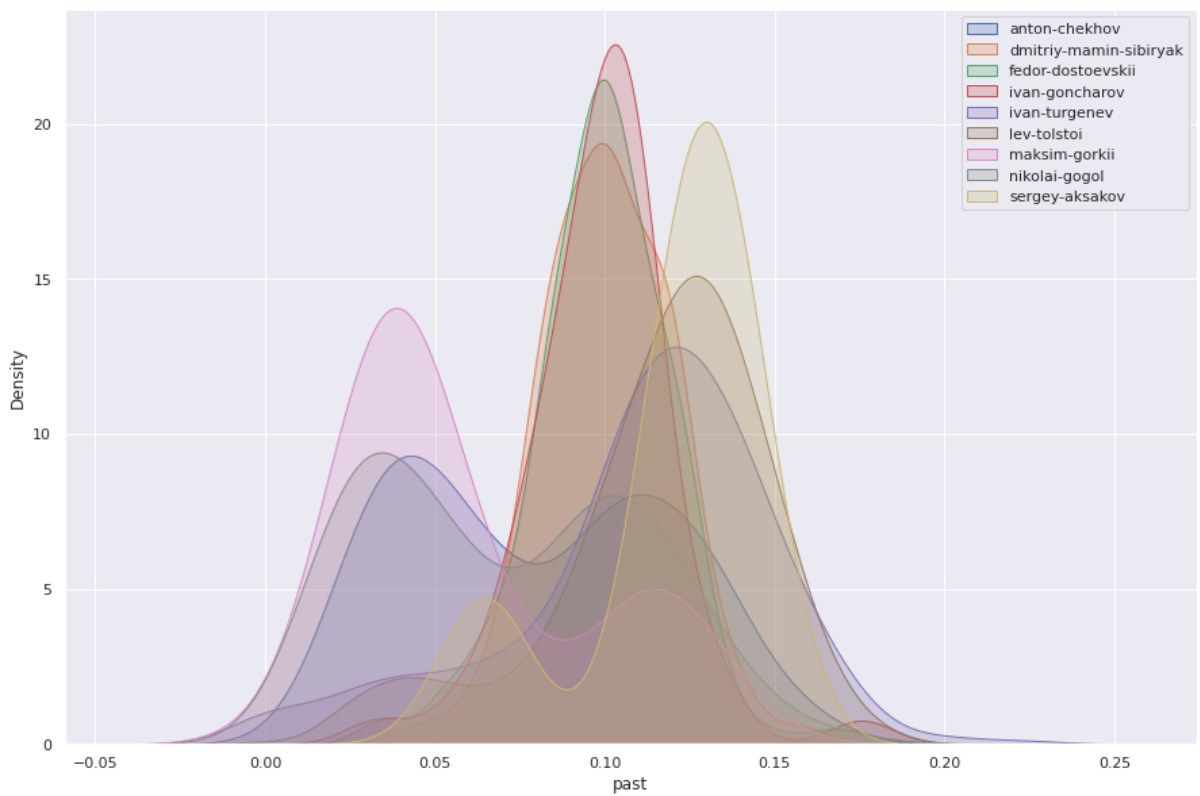
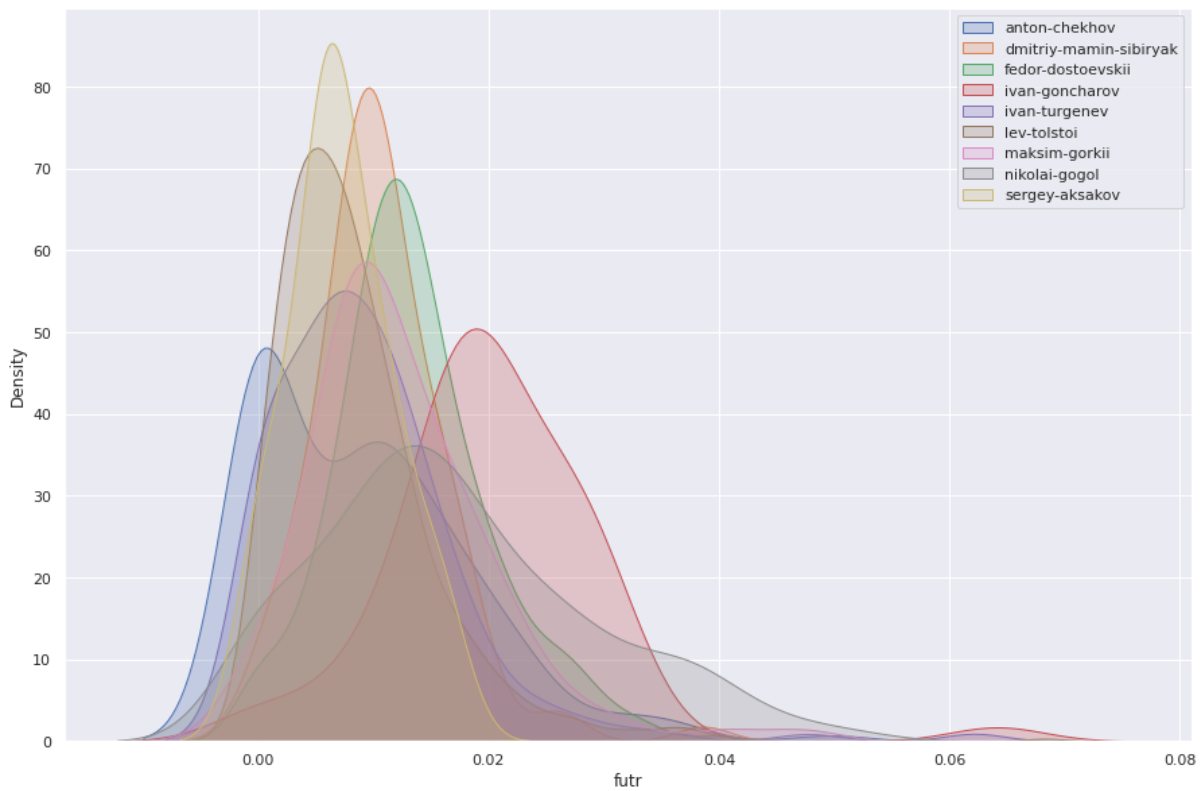


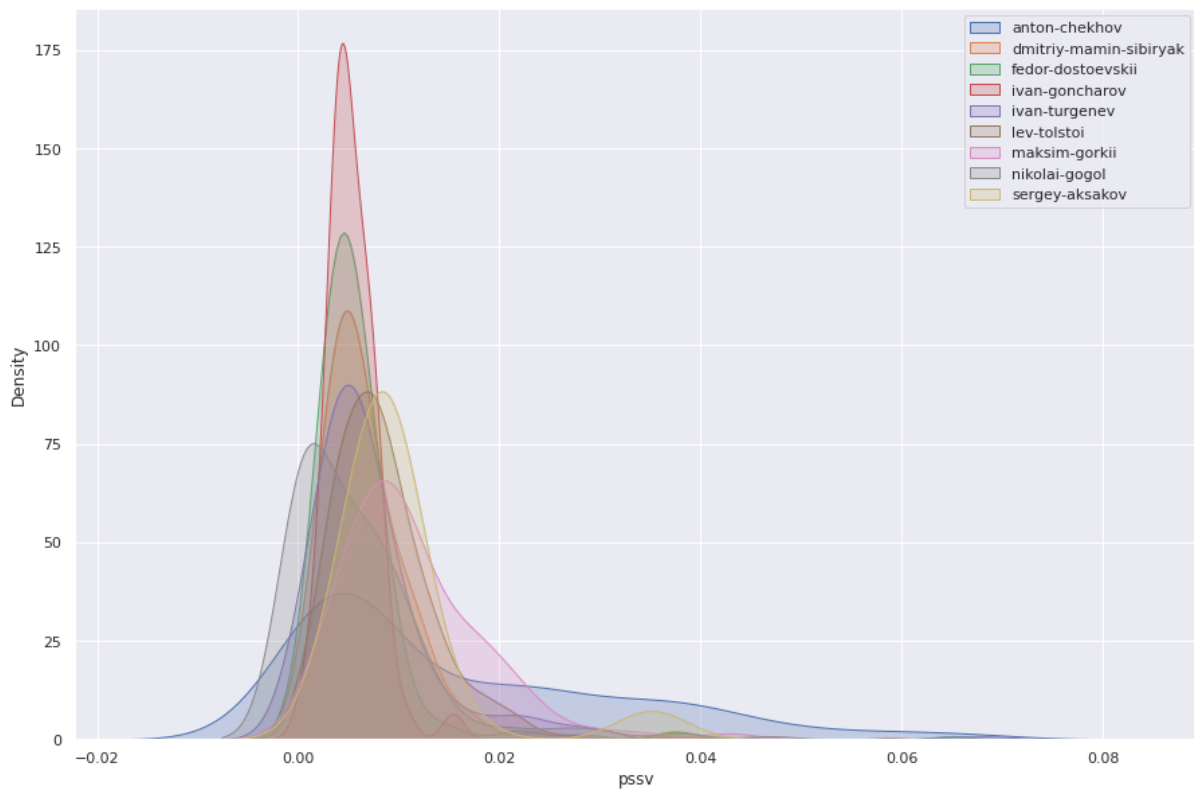








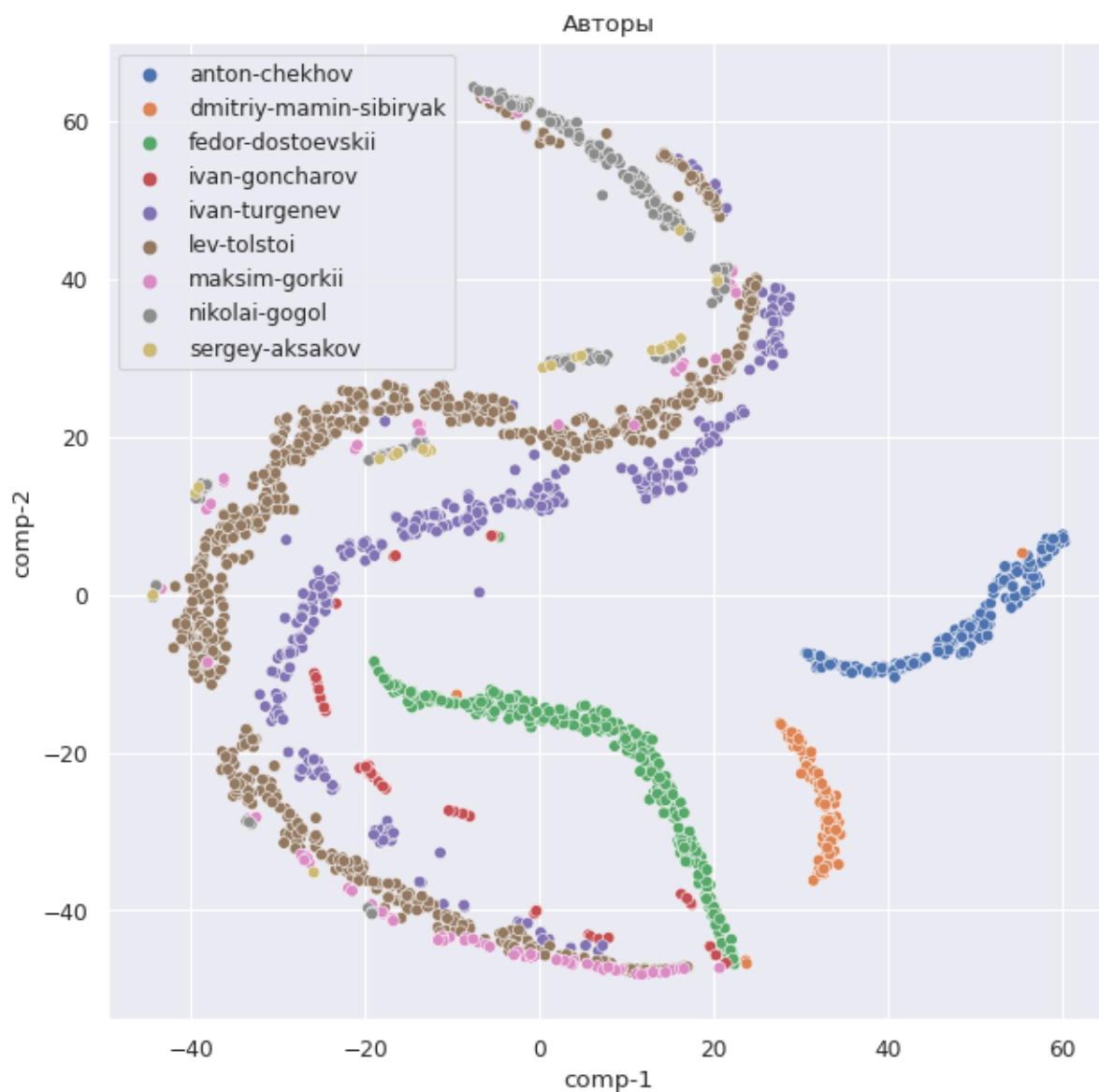




По распределениям можно сделать некоторые наблюдения, например:

- У Горького сложность глав наименее сконцентрирована
- У Аксакова чаще всех встречается женский род
- Частота использования единственного числа похожа у авторов

Визуализация всех признаков через tsne дает следующий результат:



На таком графике заметно выделяются Чехов, Мамин-Сибиряк, Гоголь, а вот, например, Толстой более разнообразный.