DOI: 10.14515/monitoring.2021.1.1750

Н. В. Ярцева

АНАЛИЗ БОЛЬШИХ ОБЪЕМОВ ДАННЫХ: ВОЗМОЖНОСТИ GDELT PROJECT ПРИ ИСПОЛЬЗОВАНИИ ЯЗЫКА ПРОГРАММИРОВАНИЯ PYTHON. ОПЫТ ГУМАНИТАРИЯ, РЕШИВШЕГО ПОСТИЧЬ BIG DATA

Правильная ссылка на статью:
Ярцева Н. В. Анализ больших объемов данных: возможности Gdelt Project при использовании языка программирования Python. Опыт гуманитария, решившего постичь Big Data // Мониторинг общественного мнения: экономические и социальные перемены. 2021. № 1. С. 351—367. https://doi.org/10.14515/monitoring.2021.1.1750.
For citation:
Yartseva N. V. (2021) Analysing Large Amounts of Data: GDELT Project's Opportunities Using the Python Programming Language. A Humanities Scholar's Experience With Big Data. *Monitoring of Public Opinion: Economic and Social Changes.* No. 1. P. 351–367. https://doi.org/10.14515/monitoring.2021.1.1750. (In Russ.)

DOI: 10.14515/monitoring.2021.1.1750

АНАЛИЗ БОЛЬШИХ ОБЪЕМОВ ДАННЫХ: ВОЗМОЖНОСТИ GDELT PROJECT ПРИ ИСПОЛЬЗОВАНИИ ЯЗЫКА ПРОГРАММИРОВАНИЯ PYTHON. ОПЫТ ГУМАНИТАРИЯ, РЕШИВШЕГО ПОСТИЧЬ BIG DATA

ANALYSING LARGE AMOUNTS OF DATA: GDELT PROJECT'S OPPORTUNITIES USING THE PYTHON PROGRAMMING LANGUAGE. A HUMANITIES SCHOLAR'S EXPERIENCE WITH BIG DATA

ЯРЦЕВА Наталья Владимировна — кандидат политических наук, доцент, Самарский университет, Самара, Россия; эксперт-консультант, Всероссийский центр изучения общественного мнения, Москва, Россия
E-MAIL: yartseva.nat@gmail.com
https://orcid.org/0000-0002-7236-1812

Аннотация. Научный мир развивается по междисциплинарному пути, одно из самых популярных направлений — соединение возможностей программирования, позволяющего обрабатывать большие объемы данных, и гуманитарного знания. В статье я описываю свой опыт освоения Big Data, анализирую возможности ресурса GDELT и показываю, как с помощью команд на языке программирования Python обрабатывать большие данные. Благодаря этому данная статья может иметь и вполне практическое применение — в ней перечислены и проанализированы шаги, которые позволят ученым, не знакомым с обработкой больших объемов данных, не только разобраться в сути метода, но и самостоятельно сделать первые шаги в обработке данных на Python. Кроме того, работа проиллюстрирована кейсом французских «желтых жилетов», что позволяет лучше разобраться в структуре кода и принципах работы GDELT.

Ключевые слова: GDELT, Big Data, Python, BigQuery

Natalia V. YARTSEVA[1,2] — Cand. Sci. (Polit.), Associate Professor; Expert
E-MAIL: yartseva.nat@gmail.com
https://orcid.org/0000-0002-7236-1812

[1] Samara University, Samara, Russia
[2] Russian Public Opinion Research Center, Moscow, Russia

Abstract. The scientific community is currently following the interdisciplinary path, and one of the most popular directions is a combination of Programing enabling processing large volumes of data and the Humanities. The paper describes the author's experience with Big Data, provides her analysis of the GDELT Project opportunities and shows how large amounts of data can be processed using the Python programming language. The analysis of steps in data processing using Python can help scholars who have not dealt with large amounts of data. The French "yellow vests" is the case study the author uses to illustrate how the GDELT Project works.

Keywords: GDELT, Big Data, Python, BigQuery

DOI: 10.14515/monitoring.2021.1.1750

В инструментарии современного гуманитарного исследователя не так много технических средств, предназначенных для работы с большими объемами данных. SPSS и «Статистика» прекрасно справляются с обработкой данных, это отличные статистические программы, предназначенные для анализа данных, с широким набором функций и параметров.

Однако и у них есть недостатки — высокие требования к системе компьютера: требуются большие объемы оперативной памяти, память на жестком диске и быстрый процессор; высокая цена.

В этой статье я расскажу:

— что такое GDELT;
— что собой представляет блог GDELT Project;
— что он делает и как повторить его аналитику;
— что может Google Cloud Platform;
— как при помощи языка SQL написать на нем запросы и что-то получить.

Затем попробуем повторить то же самое на Jupiter и Python.

Данные платформы будут представлены на примере движения «желтых жилетов» во Франции. В качестве исследовательской гипотезы я возьму предположение, что французская пресса пишет о «желтых жилетах» в более негативном ключе, чем, например, итальянская. Мотивом этого может служить тот факт, что «желтые жилеты» проводят большее количество демонстраций во Франции, чем в других европейских странах. Поэтому для Франции они условные «бузотеры», выходящие на несанкционированные протесты, а для других стран «желтые жилеты» — граждане, отстаивающие свои права, и тон публикаций на тему данного движения, соответственно, гораздо дружелюбнее.

Обзор GDELT

Платформа данных GDELT представляется принципиально новым направлением, позволяющим обрабатывать большие объемы информации: а) бесплатно и б) не привлекая большой памяти компьютера. Именно эти два критерия, на мой взгляд, и делают GDELT востребованным порталом, позволяющим за короткий срок обрабатывать и систематизировать большие данные.

GDELT[1] — мировая база данных о социальных, политических, экономических и культурологических процессах, происходящих в государствах. GDELT отслеживает мировые вещательные, печатные и веб-новости практически из каждого уголка земного шара на более чем ста языках. Он идентифицирует людей, места, организации, темы, источники, эмоции, цифры, цитаты, изображения и события, которые происходят в мире и влияют на наше общество. GDELT — это бесплатная мировая платформа, предоставляющая доступ к терабайтам самой разной информации.

GDELT в режиме 24/7 отслеживает мировые средства массовой информации практически из каждого уголка земного шара. Данные GDELT привязаны к координатам (широте и долготе), что и позволяет составлять карты: от индекса счастья до карты конфликтов. При этом у GDELT есть несколько сервисов. Сам GDELT — это источник данных, а GDELT Analytical Services — инструмент работы с этими данны-

[1] The GDELT Project. URL: https://www.gdeltproject.org (дата обращения: 19.02.2021).

DOI: 10.14515/monitoring.2021.1.1750

ми. Он и позволяет предсказывать возможность эскалации конфликта (опираясь на совокупность данных и опыт предыдущих конфликтов, которые имели свои особенности) и даже распространения вируса гриппа.

GDELT анализирует в том числе Twitter, популярные телевизионные шоу и даже региональные СМИ, не пишущие на английском языке. Все глобальные новости GDELT отслеживает практически в режиме реального времени, и это составляет 98,4 % от ежедневного объема не англоязычных СМИ. Эти материалы переводятся на английский и обрабатываются. В ближайшее время портал намерен расширить свою базу данных до 1800 г.[2]

Данные GDELT можно скачать в CSV, но поскольку их много (2,5 ТВ/год), то на практике используют либо Google BigQuery, либо GDELT Analytical Services.

Google хранит у себя «копию» GDELT (и не только его). Для анализа GDELT и выборки данных по нужным нам критериям Google предлагает использовать свой сервис — BigQuery public datasets[3]. Он хорош тем, что данные не нужно загружать на компьютер, а все общение с облаком данных происходит исключительно посредством языка SQL. Данные по указанным критериям отбираются в облаке, а на компьютер исследователя выгружаются только те строки, что подходят под критерии.

Задачи исследования

Обратимся к анализу данных GDELT. Про движение «желтых жилетов» писали все мировые СМИ — с конца 2018 г. и по сей день на эту тему вышло большое количество статей. Давайте попробуем проанализировать через GDELT, как освещались неконтролируемые акции протеста в 2019 г. в Париже и какое место среди этих акций занимали «желтые жилеты»[4].

Google Cloud Platform — платформа, представляющая собой набор облачных служб и позволяющая проводить разного рода вычисления. Она содержит около 300 различных сервисов, но не предлагает готовых решений. У нее есть русский интерфейс и пояснения для каждой задачи.

У Google Cloud Platform в открытом доступе много полезных наборов данных, облегчающих работу аналитику. В частности, в нем находятся материалы GDELT. Извлечем кусочек информации по «желтым жилетам». Для этого заходим на сервис Google Cloud Public Datasets[5], где хранятся популярные общедоступные наборы данных в облаке. Здесь можно получить доступ к более чем ста общедоступным наборам данных из разных отраслей и тем. Отсюда можно запрашивать данные непосредственно из GDELT и пользоваться понятным для гуманитария интерфейсом. Нажимаем на кнопку “Explore public datasets” и переходим к анализу общедоступных наборов данных. Далее попадаем на страницу с общедоступными наборами данных, где и находим GDELT.

---

[2] Intro // The GDELT Project. URL: https://www.gdeltproject.org/#intro (дата обращения: 19.02.2021).

[3] В BigQuery public datasets необходимо зарегистрироваться и выбрать по этой ссылке данные с GDELT — URL: https://console.cloud.google.com/marketplace/product/the-gdelt-project/gdelt-2-events.BigQuery (дата обращения: 19.02.2021).

[4] По умолчанию BigQuery API (то, при помощи чего мы будем анализировать данные) может быть выключено. Включить его можно здесь: https://console.cloud.google.com/apis/library/BigQuery.googleapis.com.

[5] Google Cloud Public Datasets. URL: https://cloud.google.com/public-datasets?hl=ru (дата обращения: 19.02.2021).

DOI: 10.14515/monitoring.2021.1.1750

По запросу “gdelt” нам выдается несколько баз данных, среди которых выбираем GDELT 2.0 Event Database. По щелчку “Посмотреть набор данных” переходим в редактор запросов. Далее выбираем папку “events” и все дальнейшие действия уже осуществляем в ней.

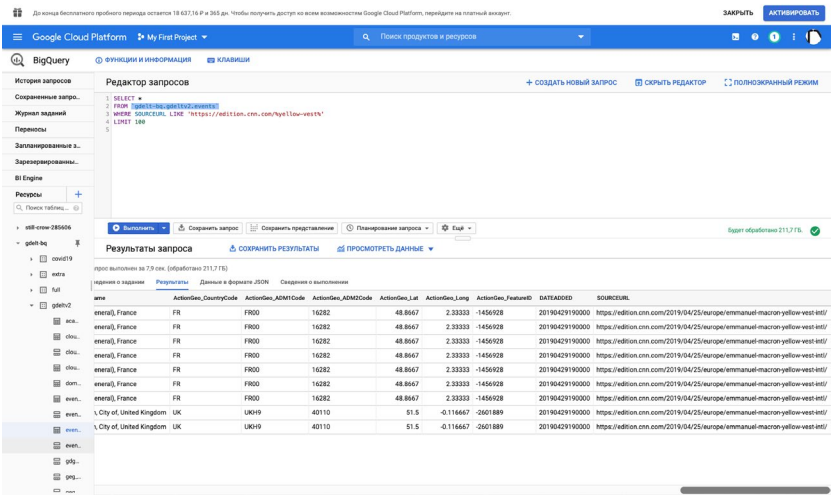
Все запросы в Google Cloud Platform мы делаем на языке SQL. Этот язык применяется для создания, модификации и управления данными. SQL — это стандарт индустрии при работе с реляционными данными (данными, организованными в таблицы со связями между таблицами)[6].

В редакторе запросов на языке SQL мы выполняем команду, где после WHERE идут условия фильтрации, а SOURCEURL — это название колонки в нужной нам таблице данных.

```
SELECT *
FROM `gdelt-bq.gdeltv2.events`
WHERE SOURCEURL like “%yellow-vest%”
LIMIT 100
```

Так мы ищем все упоминания «желтых жилетов», но лимит строк ограничиваем 100 штуками. На выходе получается обширная таблица, где собраны: даты, акторы (кто и о ком говорит), очень важный eventcode, который классифицирует событие (о нем ниже), avgTone (в каком тоне) и, наконец, ссылки на те ресурсы, с которых GDELT получает информацию.

EventCode	EventBaseCode	EventRootCode	QuadClass	GoldsteinScale	NumMentions	NumSources	NumArticles	AvgTone	Actor1Geo_Type	Actor1Geo_FullName	Actor1Geo_CountryCode	Actor1Geo_ADM1Code
040	040	04	1	1.0	6	1	6	-5.1819184123484	4	Paris, France (general), France	FR	FR00
175	175	17	4	-9.0	2	1	2	-5.1819184123484	4	Paris, France (general), France	FR	FR00
040	040	04	1	1.0	1	1	1	-5.1819184123484	4	Paris, France (general), France	FR	FR00
050	050	05	1	3.5	5	1	5	-5.1819184123484	4	Paris, France (general), France	FR	FR00
0874	087	08	2	10.0	10	1	10	-5.1819184123484	4	Paris, France (general), France	FR	FR00
120	120	12	3	-4.0	5	1	5	-5.1819184123484	4	Paris, France (general), France	FR	FR00
175	175	17	4	-9.0	6	1	6	-5.1819184123484	4	Paris, France (general), France	FR	FR00
100	100	10	3	-5.0	6	1	6	-5.1819184123484	4	London, London, City of, United Kingdom	UK	UKH9
040	040	04	1	1.0	1	1	1	-5.1819184123484	4	London, London, City of, United Kingdom	UK	UKH9



[6] SQL запросы быстро. Часть 1 // Хабр. 2019. 17 декабря. URL: https://habr.com/ru/post/480838/ (дата обращения: 19.02.2021).

 DOI: 10.14515/monitoring.2021.1.1750

Полученного количества ответов слишком много, теперь попробуем максимально сузить поиск. Давайте выберем только то, что про «желтые жилеты» писала CNN, например, отфильтровав по URL онлайн-публикации. Для этого выполним команду:

```
SELECT *
FROM `gdelt-bq.gdeltv2.events`
WHERE SOURCEURL LIKE ‘https://edition.cnn.com/%yellow-vest%’
LIMIT 100
```

На выходе снова получим таблицу со множеством граф, но уже ту, которая показывает только данные CNN.

	ActionGeo_CountryCode	ActionGeo_ADM1Code	ActionGeo_ADM2Code	ActionGeo_Lat	ActionGeo_Long	ActionGeo_FeatureID	DATEADDED	SOURCEURL
	FR	FR00	16282	48.8667	2.33333	-1456928	20190429190000	https://edition.cnn.com/2019/04/25/europe/emmanuel-macron-yellow-vest-ir
	FR	FR00	16282	48.8667	2.33333	-1456928	20190429190000	https://edition.cnn.com/2019/04/25/europe/emmanuel-macron-yellow-vest-ir
	FR	FR00	16282	48.8667	2.33333	-1456928	20190429190000	https://edition.cnn.com/2019/04/25/europe/emmanuel-macron-yellow-vest-ir
	FR	FR00	16282	48.8667	2.33333	-1456928	20190429190000	https://edition.cnn.com/2019/04/25/europe/emmanuel-macron-yellow-vest-ir
	FR	FR00	16282	48.8667	2.33333	-1456928	20190429190000	https://edition.cnn.com/2019/04/25/europe/emmanuel-macron-yellow-vest-ir
	FR	FR00	16282	48.8667	2.33333	-1456928	20190429190000	https://edition.cnn.com/2019/04/25/europe/emmanuel-macron-yellow-vest-ir
	FR	FR00	16282	48.8667	2.33333	-1456928	20190429190000	https://edition.cnn.com/2019/04/25/europe/emmanuel-macron-yellow-vest-ir
n	UK	UKH9	40110	51.5	-0.116667	-2601889	20190429190000	https://edition.cnn.com/2019/04/25/europe/emmanuel-macron-yellow-vest-ir
n	UK	UKH9	40110	51.5	-0.116667	-2601889	20190429190000	https://edition.cnn.com/2019/04/25/europe/emmanuel-macron-yellow-vest-ir

Обратите внимание на колонку EventBaseCode:

IsRootEvent	EventCode	EventBaseCode
1	175	175
0	175	175
1	175	175
0	175	175
1	175	175
0	175	175
0	175	175

Код 175, так часто присваиваемый репортажам CNN о «желтых жилетах», говорит о действиях неповиновения, подавляемых властью.

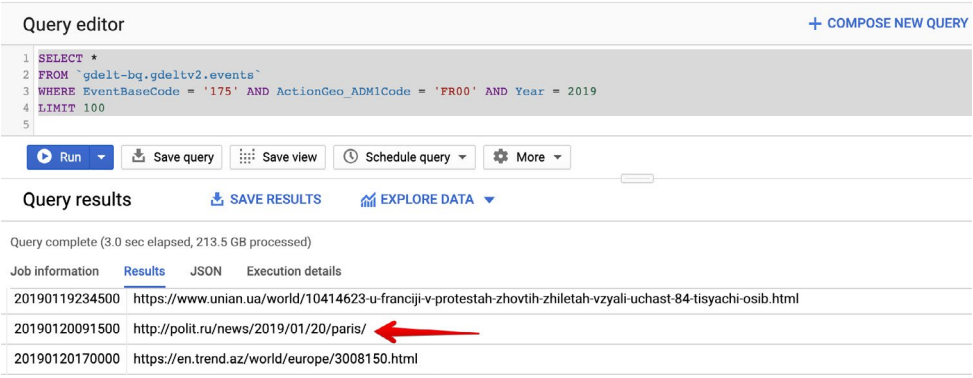
Каждую новость GDELT относит к конкретному типу. Это может быть деловая встреча, образовательный проект, уличная акция и еще множество иных вариантов событий. GDELT использует классификацию CAMEO[7].

Фильтруя по коду 175, мы можем получить ссылки на статьи мировых онлайн-изданий (а не только CNN), освещающие гражданское неповиновение и применение репрессий. В следующем примере происходит фильтрация по трем полям:

```
EventBaseCode—что случилось—“175 Use repression”;
ActionGeo_ADM1Code—где случилось—Париж;
Year—когда случилось—2019 г.
```

[7] Коды CAMEO с выдержками из медиа см.: http://data.gdeltproject.org/documentation/CAMEO.Manual.1.1b3.pdf (дата обращения: 19.02.2021).

DOI: 10.14515/monitoring.2021.1.1750



В полученной выборке (см. скриншот) фигурирует статья с сайта polit.ru, в которой говорится о разгоне демонстрантов спецназом с применением слезоточивого газа.

Какие коды нам еще могут быть интересны?

Например, регулярно встречающийся код 170, свидетельствующий о принуждении, репрессиях и насилии в отношении гражданских лиц. Например — предотвращение демонстрации.

Или код 011 — уклонение от комментариев (в любом виде и любой форме). Например, отказ представителей НАТО комментировать те или иные военные события.

Большим преимуществом Google BigQuery в сфере обработки больших объемов данных является то, что он способен (как мы увидели на примерах выше) извлекать из терабайтов данных GDELT только необходимые записи и экспортировать полученные результаты в таблицу. Эту таблицу можно сохранить на компьютер в формате CSV для дальнейшего анализа в MS Excel.

Если таблички для работы нам хватает, то этот сервис — то, что нам нужно.

GDELT Analytical Services

В процессе освоения GDELT мне очень помог блог GDELT Project[8], в частности, представленные там графики.

В поисковой строке блога забиваем, скажем, Russia. В ответ выдается большое количество статей блога, внутри которых содержатся графики, где так или иначе фигурирует Россия. Выбираем статью — Russia Fades From Television News («Россия исчезает из телевизионных новостей»). В ней с октября 2016 по июль 2020 г. собрана публикационная активность трех медиапорталов (CNN, Fox News, MSNBC) относительно представленности в американских новостях сообщений о России. Считается упоминание России в процентном соотношении к другим новостям. Если максимальные доли достигали отметки в 12%—14%, то к 2020 г. процент новостей, связанных с Россией, сократился до 2%. Это не так много, учитывая, что одно шоу Рэйчел Мэддоу составляет 5,5% всех упоминаний России.

[8] The GDELT Project. URL: https://blog.gdeltproject.org

DOI: 10.14515/monitoring.2021.1.1750

О чем говорят подобные графики? В первую очередь — как работает GDELT. По какому из принципов он может собирать и фильтровать данные. И здесь мы видим сразу несколько критериев: данные были собраны по конкретной стране, оценен общий объем материала, выпускаемого СМИ США, и внутри него выделена доля информации о России. При этом на графиках мы видим сравнение нескольких медиапорталов во временном контексте, что наглядно показывает, как изменялся процент упоминания России в трех СМИ.

Мы можем повторить это либо через инструментарий GDELT Analytical Services, либо самостоятельно, используя Jupiter Notebook и Python (об этом чуть позже).

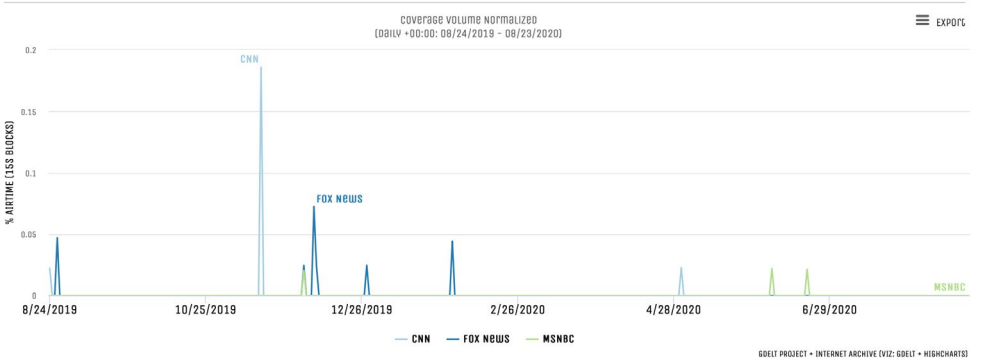
Здесь действует такой алгоритм.

Открываем страницу инструмента GDELT Summary — https://api.gdeltproject.org/api/v2/summary/summary.

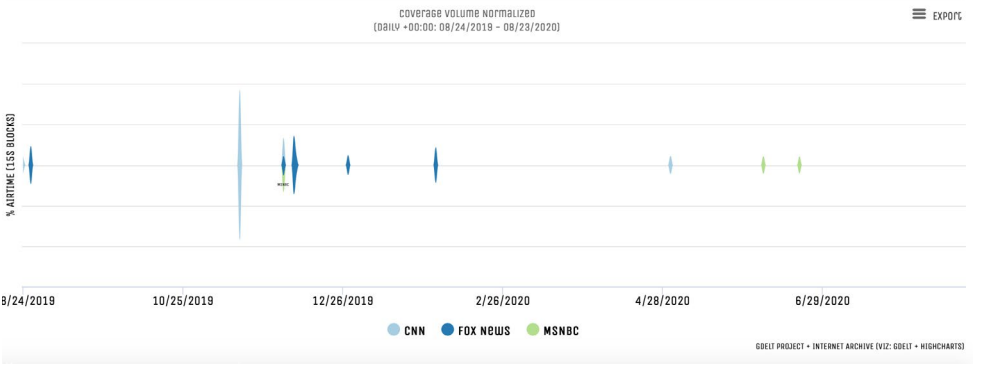
Задаем критерии выборки:

```
Step 1: Dataset = “Television News”
Step 2: Output type = “Summary Overview Dashboard”
Step 3: Search / Keywords = justice – “chief justice” – “justice department”
```

У меня получилось несколько интересных дашбордов, построенных по тому же принципу, что и блог GDELT. Например, график за последние два года с процентами эфирного времени, уделяемого «желтым жилетам».

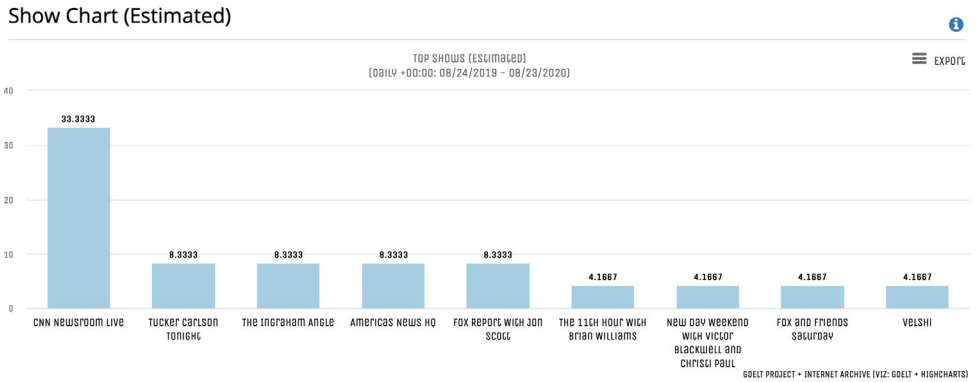


Volume Timeline (Streamgraph)



DOI: 10.14515/monitoring.2021.1.1750

А вот какие шоу были популярны в связи с протестами:



И в завершение — слова, которые чаще всего встречаются со словосочетанием «желтые жилеты»:

Мы использовали только возможности GDELT Analytical Services для того, чтобы получить нужные данные. Попробуем написать код на Python и получить нечто похожее.

DOI: 10.14515/monitoring.2021.1.1750

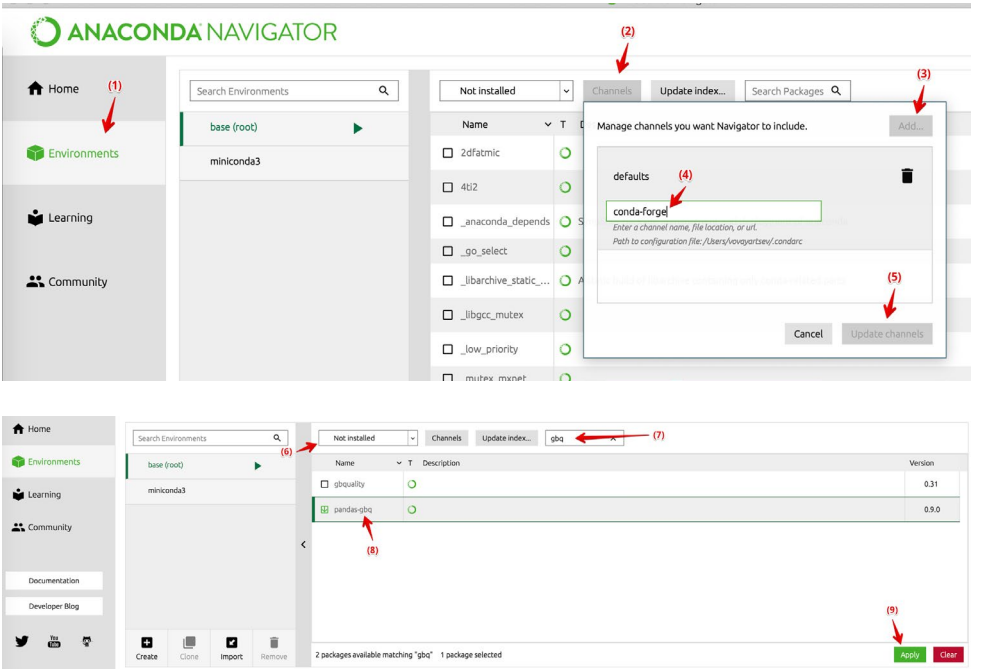
Подготовка окружения для разработки

Jupyter[9] — это среда для программирования на Python. Jupyter открывается в окне браузера и позволяет создавать и просматривать Jupyter Notebooks — интерактивные документы, содержащие Python-код, данные, графики и поясняющие тексты к ним. В интернете можно найти большое количество руководств по работе с Jupyter[10].

Один из самых быстрых способов настройки рабочей среды — это установка Anaconda[11]. На сайте в разделе Getting Started приведено 15-минутное обучающее видео (на английском языке).

По шагам это выглядит так:

— скачиваем Anaconda Individual Edition (~450 MB),
— запускаем Anaconda Navigator,
— во вкладке Environments подключаем канал conda-forge и устанавливаем пакет pandas-gbq (он нам потребуется для доступа к Google BigQuery).



Последнее действие также можно выполнить через командную строку: /opt/anaconda3/bin/conda install -c conda-forge pandas-gbq.

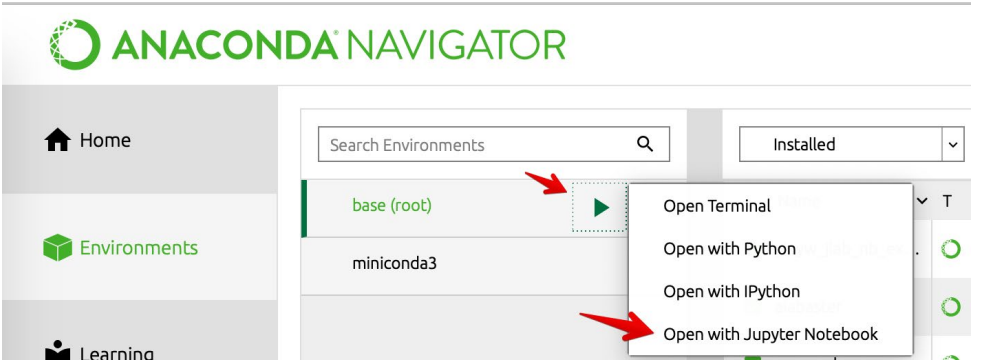
1. Во вкладке Environments щелкаем на треугольной зеленой иконке и выбираем Open with Jupyter Notebook:

[9] URL: https://jupyter.org.

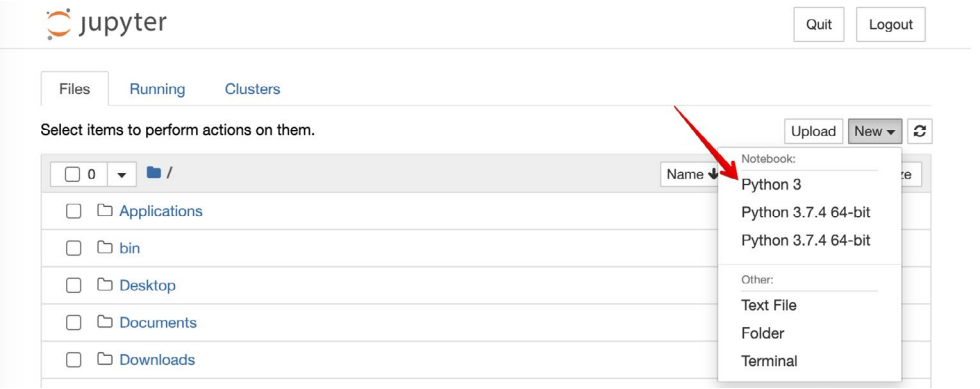
[10] Особенности Jupyter Notebook, о которых вы (может быть) не слышали // Хабр. 2016. 6 декабря. URL: https://habr.com/ru/company/wunderfund/blog/316826/ (дата обращения: 19.02.2021).

[11] URL: https://www.anaconda.com/.

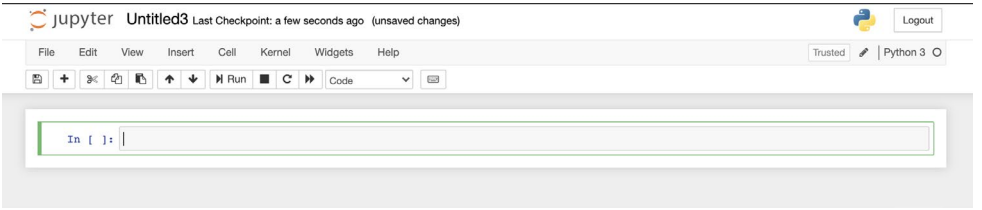
DOI: 10.14515/monitoring.2021.1.1750



2. В открывшемся окне создаем новый Notebook (Python 3):



3. Откроется окно Jupyter, куда можно писать код:



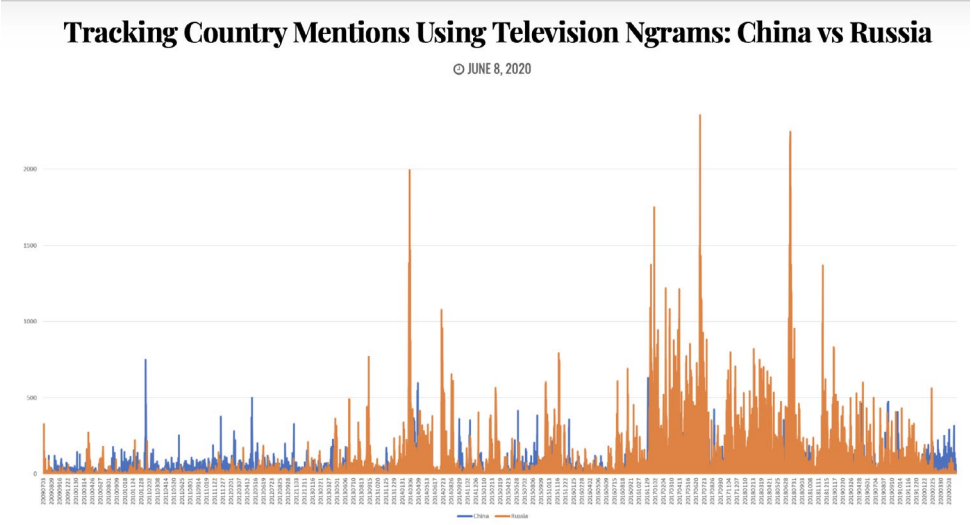
Базовый пример на Python и Jupyter

Достоинством подхода, предполагающего написание кода на Питоне, по сравнению с GDELT Analytical Services является то, что он оперирует “сырыми данными” (и может выполнять их предобработку перед построением графиков), и то, что эти графики удобно группируются в блокноты Jupiter Notebook.

Для начала возьмем одну из самых несложных задач для аналитики — упоминания в новостях России и Китая[12]:

[12] Tracking Country Mentions Using Television Ngrams: China vs Russia // The GDELT Project. 2020. 8 June. URL: https://blog.gdeltproject.org/tracking-country-mentions-using-television-ngrams-china-vs-russia/ (дата обращения: 19.02.2021).

DOI: 10.14515/monitoring.2021.1.1750

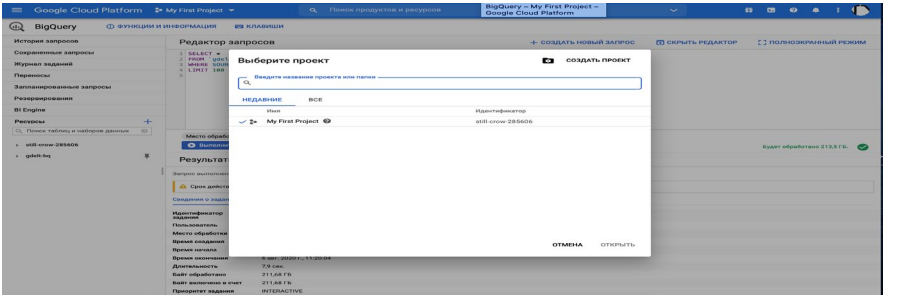


В статье блога приведен SQL-запрос для выборки исходных данных этого графика из Google BigQuery:

```
SELECT DATE, China, Russia from (
SELECT DATE, sum(COUNT) China, 0 Russia FROM `gdelt-bq.gdeltv2.iatv_1gramsv2` where STATION='CNN' and (NGRAM='china' OR NGRAM='chinese' OR NGRAM='beijing') group by DATE
UNION ALL
SELECT DATE, 0 China, sum(COUNT) Russia FROM `gdelt-bq.gdeltv2.iatv_1gramsv2` where STATION='CNN' and (NGRAM='russia' OR NGRAM='russian' OR NGRAM='moscow' OR NGRAM='kremlin' OR NGRAM='putin') group by DATE) order by DATE asc
```

Этот SQL мы возьмем за основу, чтобы воспроизвести график, приведенный в статье GDELT, средствами Python и Jupiter.

Чтобы авторизовать компьютер для работы с Google BigQuery, нам потребуется идентификатор проекта из консоли Google Cloud Platform:



Сначала построим график только по одной стране. Для этого вставим код в окно Jupyter (заменив <ваш-идентификатор-проекта> на значение со скриншота выше) и выполним его, нажав Shift-Enter:

DOI: 10.14515/monitoring.2021.1.1750

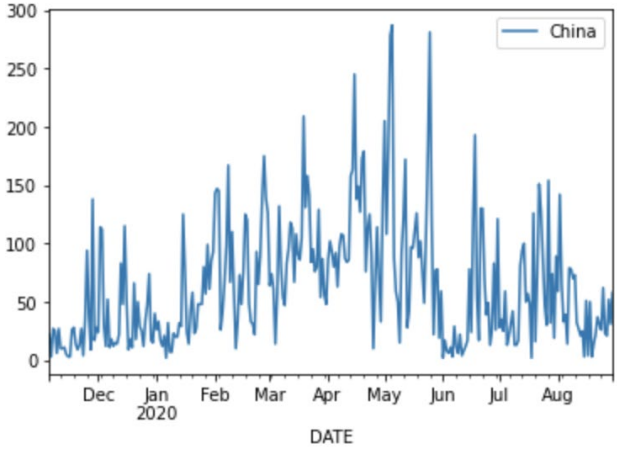
```
[*]: import pandas_gbq
     import pandas as pd


     # ВАЖНО: это должен быть ID проекта из Вашего аккаунта Google Compute Engine
     # (иначе пример не запустится)
     project_id = "calcium-bridge-691"

     sql = """
     SELECT DATE, sum(COUNT) China
     FROM `gdelt-bq.gdeltv2.iatv_1gramsv2`
     WHERE STATION='CNN' and (NGRAM='china' OR NGRAM='chinese' OR NGRAM='beijing')
     GROUP BY DATE
     """
     df = pandas_gbq.read_gbq(sql, project_id=project_id)
     df['DATE'] = pd.to_datetime(df['DATE'], format='%Y%m%d')
     df = df.groupby('DATE').sum()

     df.tail(300).plot()
```

```
Please visit this URL to authorize this application: https://accounts.google.com/o/oauth2/auth?response_type=code
5577420-unm2gnkiprugilg743tkbig250f4sfsj.apps.googleusercontent.com&amp;redirect_uri=urn%3Aietf%3Awg%3Aoauth%3A2.
tps%3A%2F%2Fwww.googleapis.com%2Fauth%2Fbigquery&amp;state=urKuyhYYdxee1DD7bMgslHnz1zoTxC&amp;prompt=consent&amp;
Enter the authorization code:
```

При первом запуске на экране появится запрос авторизации. Нужно пройти по указанной ссылке, в появившемся окне разрешить Pandas GBQ доступ к Google BigQuery, скопировать авторизационный код в окошко ввода в Jupyter и нажать Enter. При последующих запусках запрос авторизации больше появляться не будет.

В результате выполнения приведенного выше кода мы получим график, в котором каждая точка обозначает количество упоминаний Китая в новостях за соответствующий день.

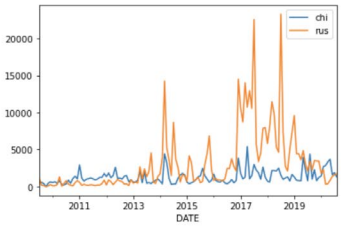
```
Out[1]: <matplotlib.axes._subplots.AxesSubplot at 0x11c549a00>
```



Немного усложним SQL-запрос для выборки данных по двум странам и используем функцию resample(), чтобы сгруппировать данные по месяцам:

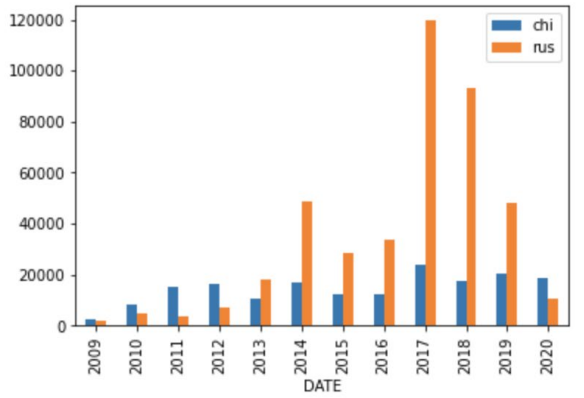
DOI: 10.14515/monitoring.2021.1.1750

```
In [7]: import pandas_gbq
        import pandas as pd


        # ВАЖНО: это должен быть ID проекта из Вашего аккаунта Google Compute Engine
        # (иначе пример не запустится)
        project_id = "alien-clover-203818"

        sql = """
        SELECT DATE, sum(China) as chi, sum(Russia) as rus from (
        SELECT DATE, sum(COUNT) China, 0 Russia FROM `gdelt-bq.gdeltv2.iatv_1gramsv2` where STATION='CNN'
            and (NGRAM='china' OR NGRAM='chinese' OR NGRAM='beijing') group by DATE
        UNION ALL
        SELECT DATE, 0 China, sum(COUNT) Russia FROM `gdelt-bq.gdeltv2.iatv_1gramsv2` where STATION='CNN'
            and (NGRAM='russia' OR NGRAM='russian' OR NGRAM='moscow' OR NGRAM='kremlin' OR NGRAM='putin') group by DATE
        )
        GROUP BY DATE
        ORDER BY DATE asc
        """
        df = pandas_gbq.read_gbq(sql, project_id=project_id)
        df['DATE'] = pd.to_datetime(df['DATE'], format='%Y%m%d')

        Downloading: 100%|| 3968/3968 [00:01<00:00, 3005.75rows/s]

In [9]: df.groupby('DATE').sum().resample('M').sum().plot()

Out[9]: <matplotlib.axes._subplots.AxesSubplot at 0x11d4e5760>
```



Этот график совпадает с тем, что мы видели в блоге GDELT, но он более читаемый. Попробуем сгруппировать упоминания России и Китая по годам:

```
In [10]: by_year = df.groupby('DATE').sum().resample('Y').sum()
         years = by_year.index.map(lambda dt: dt.year)
         by_year.set_index(years).plot.bar()

Out[10]: <matplotlib.axes._subplots.AxesSubplot at 0x11d2d1490>
```

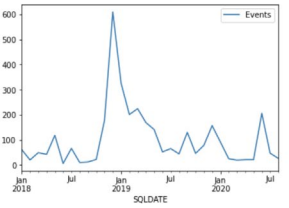


На последнем графике четко виден тренд исчезновения России из мировых новостей.

DOI: 10.14515/monitoring.2021.1.1750

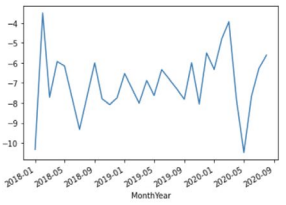
И в заключение — график количества мировых публикаций, освещающий события в Париже за 2018—2019 гг. с кодом 175 “Use repression” с группировкой по месяцам:

```
In [8]:  import pandas_gbq
         import pandas as pd


         # ВАЖНО: это должен быть ID проекта из Вашего аккаунта Google Compute Engine
         # иначе пример не запустится
         project_id = "alien-clover-203818"

         sql = """
         SELECT SQLDATE, count(*) as Events,
         FROM `gdelt-bq.gdeltv2.events`
         WHERE EventBaseCode = '175' AND ActionGeo_ADM1Code = 'FR00' AND Year > 2017
         GROUP BY SQLDATE
         """
         df = pandas_gbq.read_gbq(sql, project_id=project_id)
         df['SQLDATE'] = pd.to_datetime(df['SQLDATE'], format='%Y%m%d')
         df = df.groupby('SQLDATE').sum()

Downloading: 100%|| 535/535 [00:00<00:00, 1633.52rows/s]

In [9]:  df.resample('M').sum().plot()

Out[9]: <matplotlib.axes._subplots.AxesSubplot at 0x117da1c70>
```

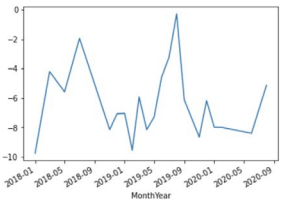


Также можно проследить изменение тона этих публикаций в отношении демонстрантов и полиции:

```
In [68]:
         sql = """
         SELECT MonthYear, Actor1Name, avg(AvgTone) as Tone,
         FROM `gdelt-bq.gdeltv2.events`
         WHERE EventBaseCode = '175' AND ActionGeo_ADM1Code = 'FR00' AND Year > 2017
         GROUP BY MonthYear, Actor1Name
         """
         df = pandas_gbq.read_gbq(sql, project_id=project_id)

Downloading: 100%|| 504/504 [00:01<00:00, 498.96rows/s]

In [73]: df_indexed = df.set_index(pd.to_datetime(df['MonthYear'] , format='%Y%m'))
         df_indexed[df_indexed.Actor1Name.eq('POLICE')].loc[:, 'Tone'].plot()

Out[73]: <matplotlib.axes._subplots.AxesSubplot at 0x1187dfa60>
```




```
In [72]: df_indexed[df_indexed.Actor1Name.eq('DEMONSTRATOR')].loc[:, 'Tone'].plot()

Out[72]: <matplotlib.axes._subplots.AxesSubplot at 0x1186ed160>
```



DOI: 10.14515/monitoring.2021.1.1750

Проверка статистической значимости

После всех выкладок и обилия кода остановимся на проверке статистической значимости моей гипотезы. Напомню, она заключается в том, что французская пресса пишет о «желтых жилетах» в более негативном контексте, чем итальянская.

При проверке статистических гипотез результат имеет статистическую значимость, когда он очень маловероятен при нулевой гипотезе. Нулевая гипотеза — это предположение, что исследуемой закономерности вообще не существует. В данной статье нулевая гипотеза — это предположение, что и французская, и итальянская пресса пишут о желтых жилетах в одинаковом контексте.

Решение о статистической значимости исследования принимают путем сравнения двух значений:

— * p-значение, p — это вероятность получения результата наблюдения при условии, что нулевая гипотеза истинна;
— α — уровень значимости исследования.

По стандартам исследования результат статистически значим, когда $p \leqslant \alpha$.

Уровень значимости α для исследования выбирается до сбора данных и обычно устанавливается на уровне 5% или ниже — в зависимости от области исследования. В любом эксперименте или наблюдении, включающем выборку из популяции/совокупности, всегда существует вероятность того, что наблюдаемый эффект произошел бы только из-за ошибки выборки. Но если p-значение наблюдаемого эффекта меньше уровня значимости (или равно ему), исследователь может заключить, что эффект отражает характеристики всей популяции, тем самым отвергая нулевую гипотезу.

В данной статье исходными наблюдениями являются значения AVGTONE публикаций на сайтах из доменов .fr и .it за 2019 г. и имеющих в URL фразу «yellow-vest».

```
sql = """
SELECT AvgTone
FROM `gdelt-bq.gdeltv2.events`
WHERE SOURCEURL LIKE 'http%.it/%yellow-vest%'
AND year = 2019
"""
it = pandas_gbq.read_gbq(sql, project_id=project_id)
```

Так как наблюдаемые значения являются нормально распределенными и независимыми, мы можем использовать t-student's test.

После проверки данных на нормальность p-value можно вычислить, используя функцию normaltest из пакета scipy.stats.

```
normaltest(it['AvgTone'])
NormaltestResult(statistic=18,688692012556487,
pvalue=8,74585136912613e-05)

normaltest(fr['AvgTone'])
NormaltestResult(statistic=78,62375287543983,
pvalue=8,454131325695306e-18)
```

DOI: 10.14515/monitoring.2021.1.1750

```
ttest_ind(fr['AvgTone'], it['AvgTone'], equal_var = False)
Ttest_indResult(statistic= -12,93921478149851,
pvalue=5,768147975657512e-23)
```

Так как p = 5e-23 < α = 0,05, гипотезу признаем статистически значимой.

Чтобы проиллюстрировать, что данный метод проверки статистической значимости работает, можно провести A-A тест (подать на вход два фрагмента одной и той же выборки, например сайтов .fr)

```
ttest_ind(fr['AvgTone'].sample(frac=0,5), fr['AvgTone'].
sample(frac=0,5), equal_var = False)

Ttest_indResult(statistic= -0,8931836531373752,
pvalue=0,3720966690912032)
```

В данном случае p = 0,37 > α = 0,05, так как обе выборки состоят из публикаций французских СМИ[13].

Таким образом, выдвинутая гипотеза подтверждается. Французская пресса действительно пишет о «желтых жилетах» в более негативном ключе, чем итальянская.

[13] Исходные коды всех примеров из данной статьи доступны по адресу https://github.com/YartsevaNat/GPD.