



## RUNER - WorksPad™ NLP

---

### Информация о релизе 1.3

МобилитиЛаб (MobilityLab, LLC.)

© 2011-2021 МобилитиЛаб (MobilityLab, LLC.). Все права защищены.

МобилитиЛаб, MobilityLab, WorksPad, логотип WorksPad, RUNER, логотип RUNER являются торговыми марками или зарегистрированными торговыми марками МобилитиЛаб (MobilityLab, LLC.) в США, России и других странах.

Названия прочих компаний и продуктов, упомянутые здесь, могут являться товарными знаками соответствующих компаний.

Продукты сторонних фирм упоминаются исключительно в информационных целях и конфигурирования зависимостей WorksPad. Компания МобилитиЛаб не несет ответственности за эксплуатационные качества и использование этих продуктов. Все договоренности, соглашения или гарантийные обязательства, при наличии таковых, заключаются непосредственно между поставщиком и потенциальными пользователями. При составлении данного руководства были предприняты все усилия для обеспечения достоверности и точности информации. Данное руководство является предметом изменений в соответствии с динамикой развития продукта и может не содержать наиболее последних версий копий экранов, имен параметров и других характеристик продукта. МобилитиЛаб не несет ответственности за опечатки или описки.

Официальная страница продукта: <https://www.workspad.ru/runer>.

Официальный веб-сайт: <https://www.workspad.ru>.

**Оглавление**

Информация о релизах .....	3
Релиз 1.3.....	3
Релиз 1.2.....	6
Релиз 1.1.....	9

## Информация о релизах

### Релиз 1.3

Дата релиза – 20 января 2021.

#### Распознавание именованных сущностей адресов, стран и городов

Главным нововведением данного релиза RUNER является поддержка распознавания адресов, а также стран и городов. Они дополняют набор стандартных сущностей RUNER. Определение конкретных полей адреса или населенного пункта – названия проспекта, улицы, номера дома, подъезда, или корпуса; предлогов и вспомогательных слов осуществляется с учетом их специфики написания и употребления в русском языке.

Например, возможен разбор:

- Адреса, составленные из череды сокращений, соответствующих конкретным объектам: “ул Ленинская Слобода 19 корп 6”, “1-й Нижн. Михайловский пр. д. 9 стр. 6”, или записанные через запятую: “Садовая-Триумфальная ул., 5, стр. 8”. Адреса, в которых номер дома может записываться кратко – “2-я ул. Марьиной Рощи 13к2с5” – распознаются, а его компоненты распределяются по соответствующим параметрам сущности (см. далее номер дома/корпуса/строения).
- Синонимичные названия: “Россия” / “Российская Федерация”, “г.Москва” / “город Москва”
- Названия населенных пунктов с дефисом: “из города Ростова-на-Дону”

Дополненная структура объекта Sentence именованными сущностями:

Параметр	Тип	Описание
<b>addresses</b>	Address[]	Список именованных сущностей типа «Адреса».
<b>cities</b>	City[]	Список именованных сущностей типа «Города».
<b>countries</b>	Country[]	Список именованных сущностей типа «Страны».

#### Объект Address

Параметр	Тип	Описание
<b>start</b>	number	Номер начального токена, соответствующего именованной сущности.
<b>end</b>	number	Номер конечного токена, соответствующего именованной сущности.
<b>type_s</b>	string	Тип адреса (улица, проспект, переулок и т.д.).

<b>street</b>	string	Название улицы.
<b>house</b>	string	Номер дома (при наличии)
<b>ownership</b>	string	Номер владения (при наличии).
<b>corpus</b>	string	Номер корпуса (при наличии).
<b>build</b>	string	Номер строения (при наличии).
<b>office</b>	string	Номер офиса (при наличии).
<b>apartment</b>	string	Номер квартиры (при наличии).

Ниже представлен результат обработки текста “ул Ленинская Слобода 19 корп 6”, “1-й Нижн. Михайловский пр. д. 9 стр. 6”, “2-я ул. Марьиной Рощи”. Как можно наблюдать из результата, извлечение параметров происходит только при их наличии в адресе, но с обязательно присутствующим типом – будь то улица, проспект, или проезд, и т.д.:

```
"addresses":
  [
    {
      "start": 2,
      "end": 7,
      "type_s": "улица",
      "street": "Ленинская Слобода",
      "house": 19,
      "corpus": 6
    },
    {
      "start": 8,
      "end": 15,
      "type_s": "проезд",
      "street": "1-й Нижн. Михайловский",
      "house": 9,
      "build": 6
    },
    {
      "start": 16,
      "end": 19,
      "type_s": "улица",
      "street": "2-я ул. Марьиной Рощи"
    }
  ]
```

## Объект City

Параметр	Тип	Описание
<b>start</b>	number	Номер начального токена, соответствующего именованной сущности.

<b>end</b>	number	Номер конечного токена, соответствующего именованной сущности.
<b>city</b>	string	Название города
<b>pre</b>	string	Предлог, относящийся к городу

## Объект Country

Параметр	Тип	Описание
<b>start</b>	number	Номер начального токена, соответствующего именованной сущности.
<b>end</b>	number	Номер конечного токена, соответствующего именованной сущности.
<b>country</b>	string	Название страны
<b>pre</b>	string	Предлог, относящийся к стране в склонении

## Нормализация названий улиц

В процессе выделения сущностей адреса, и входящих в них названий улиц, производится попытка нормализации улиц в соответствии с общепринятыми формами их записи (например, реестре). Запрос: “[...] на 1ом Михайловском пр д 5 [...]”. Адрес будет возвращен в следующем виде:

```
"addresses": [
  {
    "start": 1,
    "end": 5,
    "type_s": "проезд",
    "street": "1-й Михайловский",
    "house": 5
  }
],
```

## Расширение и корректировка работы с сокращениями

Список обрабатываемых форм сокращений, теперь включает в себя форматы записи не только с точками, но и без пробелов, а также варианты с отсутствующими точками.

## RUNER API: Изменения в JSON-результатах анализа

Были скорректированы имена параметров некоторых именованных сущностей, возвращаемых в результирующем JSON, а именно:

- Все названия физических величин в стандартной сущности Quantity теперь возвращаются в виде сокращений, записанных в латинице;
- Количественная величина “штука” в сущности Quantity, независимо от числа и склонения в обрабатываемом тексте, в возвращаемом JSON будет всегда обозначаться как “item”;
- Конфликтующие сущности, расположенные в диапазонах сущностей Addresses, или имеющие пересекающиеся диапазоны, теперь удаляются.

### Механизм предотвращения возвращения неудовлетворительных ответов

В случае, если RUNER по тем или иным причинам не смог извлечь сущности определенного типа, процесс обработки продолжится и будут возвращены те сущности, которые удалось идентифицировать, сообщив об ошибке в консоли или логе, в зависимости от настроек журналирования.

## Релиз 1.2

Дата релиза – 1 сентября 2020.

### Распознавание пользовательских именованных сущностей

Данный релиз RUNER качественно расширен возможностями обработки пользовательских сущностей – custom entity. Пользовательские сущности определяются настраиваемыми словарями, поиск в которых осуществляется на базе внутренних механизмов RUNER по нечеткому поиску, применяемому для идентификации намерений пользователя – интенгов.

Представленный механизм позволяет имплементировать сущности без необходимости глубокого понимания внутренних механизмов RUNER, а опираясь только на требования к именованной сущности, диктуемые ее контекстом применения. Логика отражается в объявлении пользовательских сущностей, задаваемых в специальном файле конфигурации. Пользователь определяет возможные значения именованных сущностей и ряд параметров, модифицирующих принципы и условия их поиска в процессе работы RUNER: например, что необходимо обнаружить, как искать, и что возвращать по обнаружению. Параметры представлены в конфигурационном файле custom/entities.conf:

```
[market_segments] # название пользовательской сущности, в фигурных скобках
type: label       # тип словаря
search: fuzzy
case_sensitive: false
path: market_segments.txt # путь к словарю, возможен без кавычек

[market_params]
TYPE: LABEL      # регистр значения не имеет
ordered = true   # значение можно передавать через знак "="
search = fuzzy
case_sensitive: false
path: "./market_params.txt"

[metro]
Type=Value      # отсутствие пробелов допустимо
```

```
case_sensitive: false
ordered: true
search: fuzzy
path: metro_stations.txt
```

Объявление пользовательской сущности начинается с написания ее название в квадратных скобках, в нижнем регистре. Значение параметра может быть записано через символы “=” и “:”, пробелы при этом опциональны. Соблюдение нижнего регистра не обязательно, за исключением имен сущностей и путей к словарям.

Среди обязательных параметров, необходимых при объявлении, присутствуют:

1. Имя сущности, записывается без пробелов, в нижнем регистре. Объявляется в квадратных скобках, за чем с новой строки следуют остальные параметры, описываемые в любом порядке.
2. Путь к словарю “path”. Может быть записан в кавычках и без. Допускаются директории с пробелами в именах, при описании пути записываются без изменений. При описании пути корневой директорией является папка custom, где расположен данный конфигурационный файл.
3. Тип словаря “type”. При анализе текста на данную именованную сущность производится поиск фраз, и, в зависимости от типа словаря в JSON помещается данная фраза в своей оригинальной форме из словаря (value), ее уникальный ключ (key), или неуникальная метка (label). Может иметь следующие значения:
  - a. “label” – словарь, где каждая фраза обладает неуникальной меткой длиной в одно слово и начинающееся с нижнего подчеркивания “\_”;
  - b. “value” – словарь-список, где записаны искомые фразы, каждая фраза начинается с новой строки файла.

Необязательные параметры можно не указывать, в таком случае они имеют значение по умолчанию.

Необязательные параметры, к регистру не чувствительны:

1. Чувствительность к регистру “case\_sensitive”. Возможные значения: true или false (по умолчанию true). Условие поиска, требующее, чтобы регистр слов искомой фразы в анализируемом тексте совпадал с записанным в словаре оригиналом.
2. Порядок записи слов “ordered”. Возможные значения: true или false (по умолчанию true). Условие поиска, требующее, чтобы слова, составляющие искомую фразу в анализируемом тексте, были в той же последовательности, какой они записаны в словаре. Между несколькими словами фразы не может быть других слов.
3. Тип поиска строки “search”. Возможные значения: precise, fuzzy (по умолчанию precise). Применение нечеткого поиска на фразе из словаря в анализируемом тексте. Применяется, когда допускается возможность записи фразы в тексте в ином падеже или форме.

### Расширение RUNER API для поиска заданного набора сущностей

Новый атрибут RUNER API “allowed\_entities” позволяет задать конкретный набор именованных сущностей, требуемых для данного запроса на обработку текста. Атрибут принимает строку, где



требуемые сущности записаны через запятую, в нижнем регистре. Указание пробелов между именованными сущностями, как разделителя между ними, опционально.

Перечисляемые именованные сущности могут быть из числа пользовательских, так и стандартных. Порядок их объявления не имеет значения. К их числу добавлено несколько служебных сущностей, используемых для краткого описания групп сущностей:

- `basic` – объединяет все стандартные сущности RUNER, за исключением сущности `persons`;
- `persons` – используется для распознавания именованных сущностей Персон (соответственно, при отсутствии данного параметра, атомарные токены, связанные с данной именованной сущностью, будут отсутствовать. А именно: `“tkFName”`, `“tkSName”`, `“tkMName”`, `“tkFullName”`, `“tkIName”`);
- `custom` – объединяет все объявленные пользовательские сущности из `entities.conf`;

Если по данному параметру была передана строка, но не содержащая именованных сущностей, то поиск сущностей не будет произведен, а только токенизация, сегментация и выявление намерений (`intent`), если таковые были переданы. В случае отсутствия обращения к данному параметру, будут найдены по умолчанию все стандартные именованные сущности RUNER (аналогично строке `“basic, persons”`).

Использование имен служебных сущностей как имен объявляемых пользовательских сущностей приведет к стандартному поведению служебных сущностей и обнаружению пользовательской именованной сущности с данным названием.

## Информационные запросы к RUNER

В данной версии RUNER теперь доступны запросы для получения информации о лицензии и зарегистрированных пользовательских словарях.

Запрос в API по адресу `/getinfo` возвращает json:

```
{
  "runer_version": "1.2.200901",
  "custom_entities": [
    "market_segments",
    "market_params",
    "metro"
  ],
  "license": {
    "id": "a8a6c1023ac2476aa9e18768205dd64f",
    "Тип": "Development",
    "Лицензиат": "MobilityLab LLC",
    "Производитель": "MobilityLab LLC",
    "Дата выписки (UTC)": "2020-03-02 07:43:20",
    "Дата начала (UTC)": "2020-03-02 00:00:00",
    "Дата окончания (UTC)": "2100-01-01 00:00:00"
  }
}
```

Запрос в браузере по адресу `/serverinfo` возвращает web-страницу с аналогичной информацией о сервере.

Версия экземпляра сервера RUNER: **1.2.200901 release**.

Информация о лицензии экземпляра RUNER:

- id: **a8a6c1023ac2476aa9e18768205dd64f**
- Тип: **Development**
- Лицензиат: **MobilityLab LLC**
- Производитель: **MobilityLab LLC**
- Дата выписки (UTC): **2020-03-02 07:43:20**
- Дата начала (UTC): **2020-03-02 00:00:00**
- Дата окончания (UTC): **2100-01-01 00:00:00**

Зарегистрированные пользовательские сущности `entities.conf`: **market\_segments, market\_params, metro**

## Установщики RUNER для Windows и Linux

На сайте технической поддержки МобилитиЛаб теперь в дополнение в дистрибутиву Windows доступны и дистрибутивы для Linux как пакеты установки DEB и RPM. Теперь RUNER работает и на российских ОС на базе Linux. Требования по версии glibc доступны вместе с дистрибутивами на странице установщиков: <https://support.workspad.com/hc/ru/articles/360013505599>.

## Размещение RUNER на Linux

RUNER на Linux устанавливается в папку `/opt/runer`.

Так как установка пакета осуществляется из под `sudo`, файл лицензии также должен быть размещен в специальной папке `/opt/runer/license` с использованием команды `sudo cp <файл-лицензии откуда> <куда положить>`.

В пользовательском интерфейсе выбора доступных программ ссылки для запуска генератора словарей и сервера RUNER размещаются в разделе Разработка / Develop.

## Релиз 1.1

Дата релиза – 16 июля 2020.

## Определение намерений пользователя

Улучшено определение намерений пользователя – интенгов (intents). Теперь для интента в JSON возвращается дополнительный параметр `score` – эвристическая оценка точности определения намерения, принимающая значение от 0 до 1.

## Расширение сегментации - разбиение текста на параграфы и последующее разбиение на предложения

Единый конвейер обработки текста RUNER расширен специальной процедурой сегментации текста с отметкой конца параграфа в виде специального “пустого” предложения в списке предложений, состоящего только из одного токена '<EOP>' нового типа **tkEOP - End Of Paragraph**.

При этом, разбиение текста на отдельные предложения теперь производится блоками - по параграфам, с учетом выделенных именованных сущностей для более точного определения конца предложения (EOS) в случаях присутствия в них инициалов персон, денежных и количественных величин, дат, а также специального нового словаря сокращений.

## Улучшение токенизации и распознавания именованных сущностей

Добавлена обработка **сокращений** (на основе специального словаря /dicts/shorts.txt с более чем 800 общеупотребимых сокращений).

Токены **tkAlpha** теперь могут включать не только буквы, но и дефисы или тире для более точной обработки сложных имен, фамилий, названий и т.п.

**URL/URI-адреса:** Возможна запись URL/URI **без протокола и в кириллице** (например, workspad.ru, яндекс.рф). Легитимность адреса определяется по наличию корректного имени домена верхнего уровня на базе нового словаря доменов верхнего уровня /dicts/top\_level\_domains.txt.

**Длительность** и другие диапазоны: Учтен символ '-' как вариация минуса в числах или длинное тире в диапазонах (например, для диапазона лет '2020-2023 гг', дат '9-12 июня' и т.п.).

Добавлен предлог 'co' (tkPre) в словарь предлогов /dicts/prepositions.dict.

**Персоны:** Добавлена обработка Инициалов персон. Введен новый тип токена **tkIName** (“Initials Name”) для токена инициалов (например, 'А.' и 'А.Б.'). Введен новый тип токена **tkFullName** для ФИО записанных в формате: один или два инициала в верхнем регистре, после которых следует фамилия, записанные слитно (А.А.Александров). При этом при формировании именованной сущности Person буквы инициалов распределяются по полям first и middle без точки.

**Персоны:** Добавлена обработка часто употребляемых фраз, где имя/фамилия по смыслу таковыми не являются (например, 'Так как' при наличии фамилии 'Так', 'Кроме того' при наличии фамилии 'Кром')

**Персоны:** Добавлен словарь многозначных фамилий ('Московская') и имен ('Израиль') совпадающих с топонимами /dicts/person\_names\_unconventional.txt. В случае, если в тексте встречаются персоны. Состоящие только из одного многозначного имени или фамилии, для таких персон в JSON-представлении персоны введен параметр **score (confidence score)** имеющий значение 0.5. Для всех однозначно определяемых персон параметр score равен 1.

В дальнейшем предполагается использовать параметр **score** и для других именованных сущностей, уверенность в точности определения которых может значительно отличаться от 100% или основываться на вероятностном определении (probability) сущности.

**Проценты:** Распознавание сущности процентов теперь выполняется с записанными словом 'процентами', а не только символа процента '%'.  
**Даты:** Теперь корректно обрабатываются фразы аналогичные "на следующей неделе" со словами "этой", "предыдущей"... Добавлен обработчик фразы "через 2-3 недели"

Ниже приведена таблица типов токенов RUNER с выделенными расширениями:

Тип атомарного токена	Описание
<b>tkAlpha</b>	Алфавитный токен, состоящий только из букв (русского и английского алфавитов в кодировке UTF-8) в нижнем регистре ('слово') или начинающийся с буквы в верхнем регистре ('Слово'), или содержащий все буквы в верхнем регистре ('СЛОВО').  Может содержать в себе точки, дефисы, слеш, и записанные подряд точку и дефис ('.-') между буквами в нижнем или верхнем регистре (ж.-д.), при том, что данное сокращение существует. Если токен состоит из чередующихся букв в верхнем регистре и точки (А.Б.В.), то он становится токеном типа tkTitle, иначе остается tkAlpha ('А.БВ.' будет tkAlpha, 'А.Б.В.' будет tkTitle)
<b>tkAlphaNum</b>	Алфавитно-цифровой токен, не попавший в критерии типов tkNumber и tkNumOrdinal (напр., '1400го' не является порядковым с двумя 'о' перед 'го')
<b>tkAnd</b>	Слово 'и'.
<b>tkCurrencySym</b>	Символ валюты (₽, \$, \$us, €)
<b>tkCurrencyWord</b>	Валюта, написанная словом (рубль, доллар, копейка) в любой падежной форме или склонении, или код валюты (RUB, USD, GBP)
<b>tkDate</b>	Запись из двух или трех чисел, разделенных точкой '.', косой чертой '/' или дефисом '-'. Первое число – это день, второе – месяц, а третье – год. Эти числа могут включать от одной до четырех цифр (01/10/2021, 23.06.87). Если в записи присутствует два числа, и второе состоит из 4 цифр, то такая запись читается как месяц, за которым следует год (06.1987). Если числа два, но второе имеет длину в 1-2 цифры, то они читаются как день и месяц (20.04). При этом, если второе число больше 12, токен читается как месяц, за которым следует год, иначе как день и месяц.
<b>tkDayOfWeek</b>	День недели, записанный словом в любом падеже или склонении (вторника, четвергу, субботе)
<b>tkEmail</b>	Токен электронной почты ( <a href="mailto:runer@workspad.com">runer@workspad.com</a> )

<b>tkEOS</b>	Сокращение от End Of Sentence. Специальный токен, обозначающий конец предложения. По окончании деления текста на предложения он автоматически ставится в конце каждого из них
<b>tkEOP</b>	Сокращение от End Of Paragraph. Служебный токен, обозначающий разделитель между параграфами. Помещается как единственный токен в служебное предложение, расположенное между двумя или после последнего, параграфами
<b>tkMoney</b>	<p>Число, записанное цифрами, с кодом/символом валюты слева или справа (130,00GBP). Может быть положительным или отрицательным и целым или дробным (разделитель целой и дробной части — запятая, допустимая один раз)</p> <p>Если число положительное и валюта записана слева от числа, то справа допустимо сокращение класса разряда числа (тысячи = k, миллионы = M, в верхнем или нижнем регистре) один раз в токене (€120k). Если число отрицательное, то код/символ валюты допустим только справа.</p> <p>Разделение разрядов обозначается точкой или пробелом. Данный разделитель может отсутствовать или встречаться несколько раз (-0,4\$, 74.500,5RUB)</p>
<b>tkMonth</b>	Месяц года, записанный словом в любой падежной форме или склонении (января, июне)
<b>tkNone</b>	Тип токена, к которому приводится токен, если он не прошел проверку ни по одному из критериев для других типов токенов, за исключением tkEOS
<b>tkNumber</b>	Количественное числительное, записанное цифрами без окончания, или с окончанием длиной до 2х букв, с возможным стоящим между числом и окончанием дефисом (12, 18-ти)
<b>tkNumOrdinal</b>	Порядковое числительное, записанное цифрами и стоящим после них окончанием с возможным дефисом, длиной до 2-х букв. Исключения составляют окончания 'ого', 'ому', 'ему' (99-ом, 4ого)
<b>tkNumWord</b>	Количественное числительное, записанное словом (девятиста, пять)
<b>tkNumWordOrdinal</b>	Порядковое числительное, записанное словом (восьмой, сороковых)
<b>tkOr</b>	Слово 'или'
<b>tkPercent</b>	Токен процентов, состоит из положительного/отрицательного целого/дробного числа, со знаком процента справа ('-12%', '153,5%')

<b>tkPhone</b>	Телефонный номер без пробелов, с возможным добавочным номером, через запятую или решетку. (495-974-7979, +7(495)974-7979#112)
<b>tkPre</b>	Слово-предлог ('за', 'около', 'через'), соответствующее предлогам из словаря prepositions.txt
<b>tkPunkt</b>	Токен, являющийся одним из символов окончания предложения, или запятой, или двоеточием
<b>tkQuotes</b>	Атомарный символ открывающихся/закрывающихся кавычек. ("", "", «», „“ — читаются посимвольно)
<b>tkSymbol</b>	Атомарный символ скобок, знаков препинания ASCII, или два идущих подряд дефиса
<b>tkTime</b>	Токен времени, где часы и минуты записаны через двоеточие (12:9, 8:30)
<b>tkTitle</b>	Токен, состоящий из букв (русского и английского алфавитов в кодировке UTF-8), чья первая буква - в верхнем регистре, а остальные в нижнем ('Слово'), или аббревиатура, где все буквы в верхнем регистре, записанные по одной и разделенные точками, но на точку может не заканчиваться ('А.Б.В.', 'А.Б'). Аббревиатуры могут быть записаны или только в кириллице или только в латинице
<b>tkFName</b>	Токен tkTitle, определенный как имя, в любой падежной форме или склонении
<b>tkMName</b>	Токен tkTitle, определенный как отчество, в любой падежной форме или склонении
<b>tkSName</b>	Токен tkTitle, определенный как фамилия, в любом падеже или склонении
<b>tkIName</b>	Токен записанного с точкой инициала в верхнем регистре (А.) или двух инициалов, записанных через точку, слитно (А.А.)
<b>tkFullName</b>	Один или два инициала в верхнем регистре, после которых следует фамилия, записанные слитно (А.А.Александров)
<b>tkURL</b>	Токен универсального указателя ресурса (Интернет-ссылка) или доменного имени, в т.ч. национального домена, например, на русском языке ( <a href="https://workspad.ru">https://workspad.ru</a> , runer.workspad.com, яндекс.рф)

## Расширение поддерживаемых именованных сущностей

Добавлен новый тип именованной сущности – **Период**

Обеспечена "умная" обработка периодов. Если время from > till, то происходит попытка увеличения времени на 12. Корректно обрабатывается фраза 'завтра с 11 до трёх' - в таком случае время будет с 11 до 15. Также обрабатываются 'через 1,5 года', 'через 1,5 часа'.

Поддерживается обработка квартала как периода – в результате отдаются дни начала квартала и соответственно конца, а также номер квартала в терминах календарного года.

## Работа со словарями

Начиная с данной версии продукта пользователи RUNER могут настраивать в экспериментальном режиме словари, используемые RUNER для определения типовых именованных сущностей.

Определены три типа словарей, каждый из которых имеет специальное расширение файла:

- Настраиваемые словари – расширение .txt
- Фиксированные словари – расширение .dict
- Генерируемые словари – расширение .dct

Настраиваемые словари предполагают возможность расширения пользователем. Например, в поставляемом в составе RUNER словаре фамилий не найдена редкая фамилия одного из сотрудников предприятия. В этом случае она может быть добавлена в словарь /dicts/snames\_all.txt. После этого необходимо запустить специальную утилиту **dict\_gen.exe** (расширение .exe в случае использования RUNER на платформе Windows, утилита доступна и в версиях для Linux и Mac, поставляемых по запросу).

Фиксированные (.dict) и генерируемые (.dct) словари не предполагают изменений. Любое изменение их содержания может нарушить работу продукта.

Словари загружаются в оперативную память при запуске RUNER. Если вы самостоятельно внесли изменения в настраиваемый словарь и обработали этот словарь утилитой dict\_gen – для использования обновленного словаря необходимо остановить и заново запустить сервер RUNER для учета внесенных расширений словаря в процессе разбора текста.

**Важно:** Прежде чем изменять тот или иной настраиваемый словарь (.txt) проконсультируйтесь с производителем во избежание нарушения работы продукта.

**Важно:** Словари в папке /dicts являются неотъемлемой частью продукта RUNER, не могут распространяться и использоваться отдельно от продукта RUNER без согласования с производителем и владельцем исключительных прав – компанией МобилитиЛаб.

## Изменения в JSON-результатах анализа

Для оптимизации обработки большого количества запросов и объемов текста в результатах анализа в JSON теперь выводятся только те разделы именованных сущности, для которых найдена как минимум одна именованная сущность. Например, если в тексте не было персон, то в JSON будет отсутствовать блок 'persons', который в более ранних версиях возвращался пустым.

### Расширение конфигурационных настроек сервера

В файле **config.conf** добавлен параметр **log\_level** для контроля детальности вывода результатов в лог. Значение по умолчанию 0 - - не выводить JSON а только время обработки post-запроса, 2 - выводить весь JSON в консоль и лог, значение 1 – зарезервировано.