

[Wiki »](#)

Как производится поиск по телефонной книги распознанной строки

Ответ от ASR

На этапе поиска по телефонной книге уже получен ответ от ASR.
Причем ответ - это не просто строка, а список строк.

Условный пример ответа от ASR:

```
алексей  
алексей лосев  
алексей лысенко
```

Который кастуем в список вида:

```
["алексей лысенко", "алексей лосев", "алексей"]
```

Где **"алексей лысенко"** финальный результат, а остальные промежуточные.

Причина по которой мы учитываем промежуточные варианты проста - в них может быть правильный вариант распознавания. Вероятность этого мала, но она есть.

По какой книге производится поиск

Поиск производится по xml книге, взятой с restfs.

Restfs в свою очередь формирует xml по данным из одного выбранного источника: ssw, mysql, ldap.

При старте автосекретаря производится импорт xml. По ней формируется структура вида ключ-значение, где ключ - имя, значение - телефон.

Кроме этого формируется ещё одна подобная структура, содержащая уникальные имена и фамилии из книги, где ключ - уникальное имя, значение - полное имя.

По этим двум мапам - телефонной книге и уникальным именам и производится поиск.

Алгоритм поиска

1. Берем строку из списка
2. Проверяем, содержится ли данная строка в таблице **hard_aliases**
3. Раскрываем сокращения (алиасы из таблицы **aliases**) и добавляем их в список, по которому будет производиться рекурсивный поиск
4. По каждой строке в сформированном списке делаем поиск
 1. Формируем список с перестановками для строки
 1. Проверяем содержится ли строка (именно целиком, а не как подстрока) в уникальных именах. Если да - заменяем эту строку на полное имя
 2. Ищем строку в телефонной книге. Если номер найден поиск закончен
5. Если в списке из пп. 3 ещё остались строки, то берем следующую

Пример

Имеем ответ от ASR:

["жениа казарцев", "жениа казарцев", "жениа"]

Двигаемся согласно алгоритму:

1. Берем строку из списка
"жениа казарцев"
2. Проверяем, содержится ли данная строка в таблице hard_aliases
В нашем примере её там нет, так что передаем строку дальше как **"жениа казарцев"**
3. Раскрываем сокращения
["жениа казарцев", "божена казарцев", "евгений казарцев", "евгения казарцев"]

4. По каждой строке в сформированном списке делаем поиск

Строки "женя казарцев" и "божена казарцев" пропущены т.к. пройдя всю цепочку ниже номера для них не были найдены

"евгений казарцев"

1. Формируем список с перестановками для строки

["евгений казарцев", "казарцев евгений"]

1. Проверяем содержится ли строка (именно целиком, а не как подстрока) в уникальных именах. Если да - заменяем эту строку на полное имя

не содержится, передаем как есть - **"евгений казарцев"**

2. Ищем строку в телефонной книге. Если не найден берем следующий элемент из списка с перестановками

евгений казарцев -> 4819

5. Если в списке из пп. 3 ещё остались строки, то берем следующую

Здесь мы пропустили строки **"женя казарцев"** и **"божена казарцев"**

Будущие улучшения

В следующем релизе этот алгоритм будет сильно изменен.

Поиск будет осуществляться в отдельном сервисе ru-phonebook **#153491**.

Строка будет анализироваться с помощью статистической модели (будет некоторое количество метрик, влияющие на финальную вероятность).

Если получится, то так же будет анализироваться персональная история пользователя и учитываться в статистической модели.