



## ЛОГИЧЕСКИЙ ДЕТЕКТИВ

Мержанова Анастасия,  
Олейник Иван,  
Сиднев Михаил

### ДЕЛО №2 ПРОИСШЕСТВИЕ ВТОРОЕ. О КСЮШИНОМ КОТЕ

Вашему вниманию представляется ещё один детективный сюжет. Конечно, его можно разгадать без использования математической логики и не прибегая к помощи компьютерных технологий. Но без интереса не может остаться способ разгадки электронного детектива и теоретический способ, на который этот детектив опирается, именно об этом мы Вам и расскажем.

#### ИСТОРИЯ

В квартире живут 4 студентки: Лиза, Света, Галя и Ксюша. Не все они одинаково прилежны, но Света всегда ходит на лекции. У Ксюши есть кот Мурзик. Еду в миску коту накладывает тот, кто первым возвращается вечером с учёбы, и Мурзик сразу съедает весь корм. Одна из соседок решила устроить Ксюше во вторник первоапрельский розыгрыш и спрятала кота в другом помещении. Ксюша должна угадать, кто похитил Мурзика?

У неё есть некоторые предположения, и она хочет отвергнуть ложные из них и подтвердить правильные, так чтобы ответ однозначно указал на похитительницу. Вот что она думает:

«Наверняка, подшутил кто-то из моих соседок».

«Первое, что приходит на ум — это сделала Света в первой половине дня».

«Помнится, что Галя вчера была дома».

«Кто-то успел покормить кота до его пропажи».



Итак, начнём расследование.

## НЕМНОГО О МЕТОДАХ АВТОМАТИЧЕСКОГО ДОКАЗАТЕЛЬСТВА

Если вдумчиво прочитать условие и присмотреться к изображению, можно раскрыть это дело и не прибегая к помощи компьютера. Но в этой статье мы будем подробнее рассматривать именно способ решения данной задачи с помощью программ, автоматизирующих логический вывод.

Начать нужно с формализации условия задачи (так как разговорный язык не является строгим). Предложения на естественном языке могут быть переведены в формальные логические выражения различным образом. Именно поэтому на сайте логического детектива приводится формализация задачи:

- Н – Галя была дома в день похищения;
- Е – Мурзик был покормлен перед пропажей;
- М – Мурзика похитили вчера до обеда;
- С – Света похитила кота;
- G – Галя похитила кота;
- L – Лиза похитила кота;

Кроме того, на сайте логического детектива читатель может найти логические формулы, соответствующие частям текста, из которых можно вынести необходимую для расследования информацию (здесь восклицательный знак (!) обозначает логическую операцию НЕ, амперсанд (&) соответствует логической операции И, знак вертикальной черты (|) обозначает ИЛИ, а импликация, которая связана с высказыванием типа «ЕСЛИ А, ТО В» обозначается привычным образом – стрелкой ( $\Rightarrow$ )).

Приведём таблицу соответствия:

Табл. 1

Свидетельские показания (часть из них опровергается уликами на картинке)	Логические выражения
Одна из соседок решила устроить Ксюше во вторник первоапрельский розыгрыш и спрятала кота в другом помещении	$(S \mid (G \mid L))$
Первое, что приходит на ум — это сделала Света в первой половине дня	$(S \& M)$
Помнится, что Галя вчера была дома	Н
Кто-то успел покормить кота до его пропажи	Е
Данное выражение не соответствует определённой части текста, но вывод сделан из части: «Еду в миску коту накладывает тот, кто первым возвращается вечером с учёбы, и Мурзик сразу съедает весь корм»	$(!E \Leftrightarrow M)$
Данное выражение не соответствует определённой части текста, но является обоснованным (Ведь нельзя похитить кота, не находясь при этом там, где произошло похищение)	$(G \Rightarrow H)$

### Использование улик с картинки

Есть ли противоречия в соответствии условия и картинки? Имеем логическое выражение, в котором говорится о том, что кто-то успел покормить кота до его пропажи, но на картинке миска полна. Это явное несоответствие, поэтому мы имеем право инвертировать выражение (Е), получаем  $!(E)$ . Кроме того, есть информация о том, что кота похитила Света

в первой половине дня, только вот в расписании над её столом написано, что в понедельник у Светы занятия всю первую половину дня. Вспомнив о том, что из текста получена информация, что сегодня вторник, а кота похитили вчера, и Света исправно посещает занятия, снова замечаем явное несоответствие. И выражение  $(S \& M)$  переходит в  $!(S \& M)$ . Не стоит упускать из вида то, что в условии говорится о том, что Галя в день похищения была дома. При этом сегодня 1 апреля, значит, вчера было 31 марта. А на двери Галиной комнаты висит записка, где указано, что она была у родителей, при этом дата, указанная в записке – 31 марта. Вот и третье несоответствие! Значит,  $(H)$  берётся под отрицание и получается  $!(H)$ . Итак, мы опровергли три логических выражения, используя улики на картинке, остальные выражения переносятся на следующий этап нашего расследования без изменений.

### Логический вывод

Из получившегося набора выражений нам нужно получить разгадку. Для того чтобы распутать детектив, нужно доказать виновность одного и невиновность остальных. Каким образом мы можем это сделать? Давайте воспользуемся методом резолюций.

Все полученные нами выражения разложим на элементарные дизъюнкты и будем считать их набором данных, необходимых для расследования.

В ходе доказательства будем действовать методом «от противного»: для доказательства невиновности подозреваемого добавляем к уже имеющимся дизъюнктам выражение, которое говорит о том, что выбранный нами подозреваемый виновен; а для доказательства виновности – добавим выражение, говорящее о непричастности данного подозреваемого к делу. Предположение о виновности/невиновности подозреваемого доказано, если комбинацией резольвент полученного нами набора мы пришли к лжи.

Итак, расследование данного преступления – это поэтапное доказательство невиновности двоих соседа и виновности третьей. Ясно, что верным является один из трёх вариантов (Галя – виновна, а Лиза и Света – нет; Света – виновна, а Галя и Света – нет; Лиза – виновна, а Галя и Света – нет). Если мы возьмёмся за доказательство неправильного предположения, то будем перебирать все возможные варианты, но не придём к лжи. Нас же интересует доказательство верного предположения.

Приведём его:

В ходе преобразования логических выражений в элементарные дизъюнкты получили такой набор:

1.  $(S \mid (G \mid L))$
2.  $(!S \mid !M)$
3.  $!H$
4.  $!E$
5. а).  $(E \mid M)$     б).  $(!E \mid !M)$
6.  $(!G \mid H)$

Итак, первое предположение: «Похититель – не Света», значит, к набору добавляется  $S$ .

7.  $S$

8.  $\text{Res}(2,7)=!M$

9.  $\text{Res}(8,5a)=E$

10.  $\text{Res}(9,4)=0$ , а это противоречие, значит,  $!(S)$ , то есть похититель – не Света.

Далее, второе предположение: «Похититель – не Галя», к набору добавляется  $G$ .

11.  $G$

12.  $\text{Res}(6,11)=H$

13.  $\text{Res}(12,3)=0$ , и снова противоречие, то есть  $!(G)$  - Похититель – не Галя.

Наконец, третье предположение - «Похититель – Лиза», к набору добавляется  $!L$ .

14.  $!L$

15.  $\text{Res}(14,1)=(S \mid G)$

16.  $\text{Res}(15,10)=G$

17.  $\text{Res}(16,13)=0$ , данное противоречие и является последним пунктом доказательства нашего предположения, в котором мы подтвердили, что похититель – Лиза.

Но это лишь теория, как же эти действия проделывает компьютер?

Улучшение текста доказательства на разговорном языке.

В первом выпуске журнала уже описан метод построения текста расследования, теперь расскажем, как улучшить построенный (сгенерированный) текст.

Для дела №1 «О Ксюшином коте» данный метод строит следующий текст:

Знаем, что если Галя не была дома в день похищения, то Галя не похищала кота. Факты говорят о том, что Галя не была дома в день похищения. Делаем вывод, что Галя не похищала кота.

Известно, что если Света похитила кота, то Мурзика не похищали вчера до обеда. Также нам известно, что если Света не похищала кота, то Галя похитила кота или Лиза похитила кота. Делаем вывод, что если Мурзика похитили вчера до обеда, то Галя похитила кота или Лиза похитила кота.

Есть информация, что если Мурзик не был покормлен перед пропажей, то Мурзика похитили вчера до обеда. Знаем, что Мурзик не был покормлен перед пропажей. Заключаем, что Мурзика похитили вчера до обеда.

Знаем, что если Мурзика похитили вчера до обеда, то Галя похитила кота или Лиза похитила кота. Знаем, что Мурзика похитили вчера до обеда. Приходим к выводу, что если Галя не похищала кота, то Лиза похитила кота.

Известно, что если Галя не похищала кота, то Лиза похитила кота. Известно, что Галя не похищала кота. Следовательно, Лиза похитила кота.

Как видно данный текст богат тавтологией: повторяются подлежащие, сказуемые, даже целые предложения. Для того чтобы от этого избавиться был реализован метод, устраняющий некоторые тавтологии. Такие как:

18. Повторение предложений.

19. Повторение подлежащих в разных частях одного предложения.

20. Повторение одинаковых сказуемых (которые можно заменить на одно сказуемое множественного числа).

Каждая из тавтологий устраняется по-своему. Более подробно о том, как это происходит:

### 1. Повторение предложений.

Сгенерированные предложения строятся по определенному правилу:

**<Предложение>\* ::= \*\* <Вводная конструкция> <условие>, <следствие>.**

При этом условная часть всегда начинается со слова «если», а главная со слова «то».

**<условие> ::= если <логическая переменная>**

**<следствие> ::= то <перечисление логических переменных>**

Также следует отметить, что каждый абзац текста состоит из трех предложений:

**<Абзац> ::= <резольвента А><резольвента В><резольвента С=А res В>**

Поэтому если условие и следствие предыдущего абзаца совпадают с условием и следствием первого предложения текущего, то мы просто удаляем это предложение.

### 2. Тавтология подлежащих.



Для решения данной проблемы нам необходимо заменить подлежащее в «следствии» на его местоимение, если оно встречалось в «условии».

Для того чтобы правильно определить местоимение, необходимо знать род подлежащего. Для этого мы заведем новую структуру – словарь, в которую создатель каждой задачи должен внести слова, которые могут быть заменены на их местоимения.

Пример:

```
var dictionary = [
  {
    word: "Лиза",    //слово
    gender: "F",     //род M-мужской|F-женский|N-средний
    words: "",       //множественное число
    type: "N",       //часть речи N-существительное|V-глагол
    pronoun: "она",  //местоимение
  },

```

### 3. Тавтология сказуемых.

Для решения данной проблемы мы также используем словарь. В данном случае необходимо удалить первое вхождение одинаковых глаголов «следствия» и заменить второе вхождение на множественное число. Например: «..., то Галя похитила кота или Лиза похитила кота» превратить в «..., то Галя или Лиза похитили кота». Аналогично решая другие задачи, можно столкнуться с повторным союзом «или», и для этого было необходимо создать дополнительно условия постановки запятой перед повторным союзом «или» при однородных членах. Пример текста и его преобразования в таких задачах: «...то Дэн украл монету, или Джон украл монету, или Шон украл монету, или Рэй украл монету» в «...то Дэн, или Джон, или Шон, или Рэй украли монету».

После всех преобразований можно наблюдать результат работы метода устранения тавтологий:

Знаем, что если Галя не была дома в день похищения, то она не похищала кота. Известно, что Галя не была дома в день похищения. Можно сделать вывод, что Галя не похищала кота.

Известно, что если Мурзик не был покормлен перед пропажей, то Мурзика похитили вчера до обеда. Известно, что Мурзик не был покормлен перед пропажей. Очевидно, что Мурзика похитили вчера до обеда.

Знаем, что если Света похитила кота, то Мурзика не похищали вчера до обеда. В то же время известно, что если Света не похищала кота, то Галя или Лиза похитили кота. Следовательно, если Мурзика похитили вчера до обеда, то Галя или Лиза похитили кота.

Нам уже известно что, Мурзика похитили вчера до обеда. Приходим к выводу, что если Галя не похищала кота, то Лиза похитила кота.

Ранее мы доказали что, Галя не похищала кота. Следовательно, Лиза похитила кота.

Разумеется, в тексте всё еще сохранились другие тавтологии, и человек с легкостью справится с их устранением, однако машина не имеет понятия о литературности языка и обучить её строить приятные для прочтения предложения крайне сложно.



Наши авторы, 2014.  
Our authors, 2014.

**Мержанова Анастасия,  
Олейник Иван,  
Сиднев Михаил,  
студенты 2 курса факультета  
компьютерных технологий и  
информатики СПбГЭТУ «ЛЭТИ».**

