

06, июнь 2016

УДК 004.512

Интерфейс с использованием контролируемого естественного языка

*Гутников И.Е., студент
Россия, 105005, г. Москва, МГТУ им. Н.Э. Баумана,
кафедра «Информационные системы и телекоммуникации»*

*Научный руководитель: Иванов А.М., старший преподаватель
Россия, 105005, г. Москва, МГТУ им. Н.Э. Баумана,
кафедра «Информационные системы и телекоммуникации»
amivanoff@gmail.com*

В настоящее время интерфейсы большинства инженерных программ, таких как системы автоматизированного проектирования, интегрированные среды разработки часто имеют сложный интерфейс.

Одним из путей решения проблемы сложного интерфейса является использование интерфейса, в котором реализован контролируемый упрощенный естественный язык.

Для того чтобы интерфейс был более простым и понятным пользователю, идеальный интерфейс или агент такого интерфейса должен уметь разговаривать с людьми в свободной форме, на естественном языке, так же, как это делают люди. Но данную задачу до сих пор трудно решить в современных программных системах, поэтому приходится использовать дополнительные ограничения, такие как общение на основе шаблонов или специальных меню. Данные ограничения могут значительным образом облегчить реализацию коммуникаций, но они также ограничивают выразительность и гибкость интерфейса.

Компромиссом между ограничениями и сложностью общего понимания естественного языка является разработка интерфейса с использованием контролируемого естественного языка.

Контролируемый естественный язык

Контролируемый естественный язык напоминает обычный язык, но имеет ограниченный набор строгих правил языка, лексики и однозначности смысла. Эти ограничения позволяют успешно обрабатывать контролируемый естественный язык,

избегая неоднозначности смысла слов и неопределенности правил грамматики, и явным образом связывая язык к контекстной информации.

К сожалению, некоторые проблемы с контролируруемыми естественными языками ограничивают повсеместное их использование:

— Контролируемый естественный язык не является точно таким же, как естественный язык, который известен пользователю, поэтому пользователь должен изучить его. Изучение контролируемого естественного языка может занять некоторое время, которого зачастую нет. Также, пользователь может «адаптироваться» к контролируемому естественному языку со временем, но это возможно, только тогда, когда пользователь использует интерфейс с контролируемым естественным языком достаточно долго;

— Контролируемый естественный язык не достаточно гибкий, чтобы адаптироваться к изменениям в базовой программной системе (например, в структурах данных). Изменения в базовой программной системе как правило приводят к тому, что интерфейс, использующий контролируемый естественный язык, как и сам контролируемый язык, устаревает и должен быть адаптирован вручную [1].

Для того что бы преодолеть описанные выше ограничения, необходимо использовать ограничить использование интерфейса на контролируемом естественном языке в определенных рамках и генерировать такой язык на основе модели. Данные действия позволят облегчить создание и обслуживание контролируемого естественного языка.

Ограничения интерфейса на контролируемом естественном языке

Для того чтобы принудить пользователя следовать правилам контролируемого языка, необходимо создать определенные ограничения в пользовательском интерфейсе. Эти ограничения не позволяют пользователю отклоняться от правил лексики контролируемого языка путем анализа вводимых пользователем данных и автоматической адаптацией входных данных к виду, который задан в контролируемом естественном языке. Такие ограничения также оказывают помощь пользователю при использовании контролируемого естественного языка предлагая ему возможные языковые конструкции, слова или выражения [2].

Самой простой реализацией подобного ограничения является метод интеллектуального ввода текста. Этот метод постоянно анализирует вводимый

пользователем текст и содержит набор возможных предложений, основанный на уже введённом тексте и содержит возможные предложения [3].

При использовании такого метода мы достигаем двух целей – ввод текста, построенный в соответствии с правилами языка и немедленная помощь пользователю в использовании контролируемого языка. Пользователь может использовать интерфейс практически ничего не зная о нем и со временем язык интерфейса будет становится более простым и понятным. Пример подобного интерфейса представлен на рисунке 1, где изображена консоль языка Erlang.

```
Erlang/OTP 17 [erts-6.2] [source] [64-bit] [smp:4:4] [async-threads:10] [hipe] [kernel-poll:false] [dtrace]

Eshell V6.2 (abort with ^G)
1> c(mobius.erl),
* 1: syntax error before: '.,'
1> c(mobius),
{ok,mobius}
2>
application          application_controller  application_master
beam_a                beam_asm                beam_block
beam_bool             beam_bsm                beam_clean
beam_dead             beam_dict               beam_except
beam_flatten          beam_jump               beam_lib
beam_opcodes          beam_peep               beam_receive
beam_split            beam_trim               beam_type
beam_utils            beam_validator          beam_z
binary                c                        cerl
cerl_clauses          code                    code_server
compile               core_lib                dict
edlin                 edlin_expand            epp
erl_bifs              erl_bits                erl_distribution
erl_eval              erl_expand_records      erl_internal
erlLint               erl_parse                erl_prim_loader
erl_scan              erlang                  error_handler
error_logger          error_logger_tty_h      erts_internal
ets                   file                     file_io_server
file_server           filename                 gb_sets
gb_trees              gen                       gen_event
gen_server            global                   global_group
group                 heart                    hipe_unified_loader
inet                  inet_config              inet_db
inet_parse            inet_udp                 init
io                    io_lib                   io_lib_format
io_lib_pretty         kernel                   kernel_config
lists                 mobius                   net_kernel
```

Рис. 1. Интерфейс консоли языка Erlang

Это решение накладывает дополнительные требования на обработку естественного языка. Такая обработка должна поддерживать обработку фрагментов предложения и генерировать «продолжения» в реальном времени, пока пользователь печатает предложение. Подобное решение представлено на рисунке 2, где показан интерфейс программы Sublime Text 3.

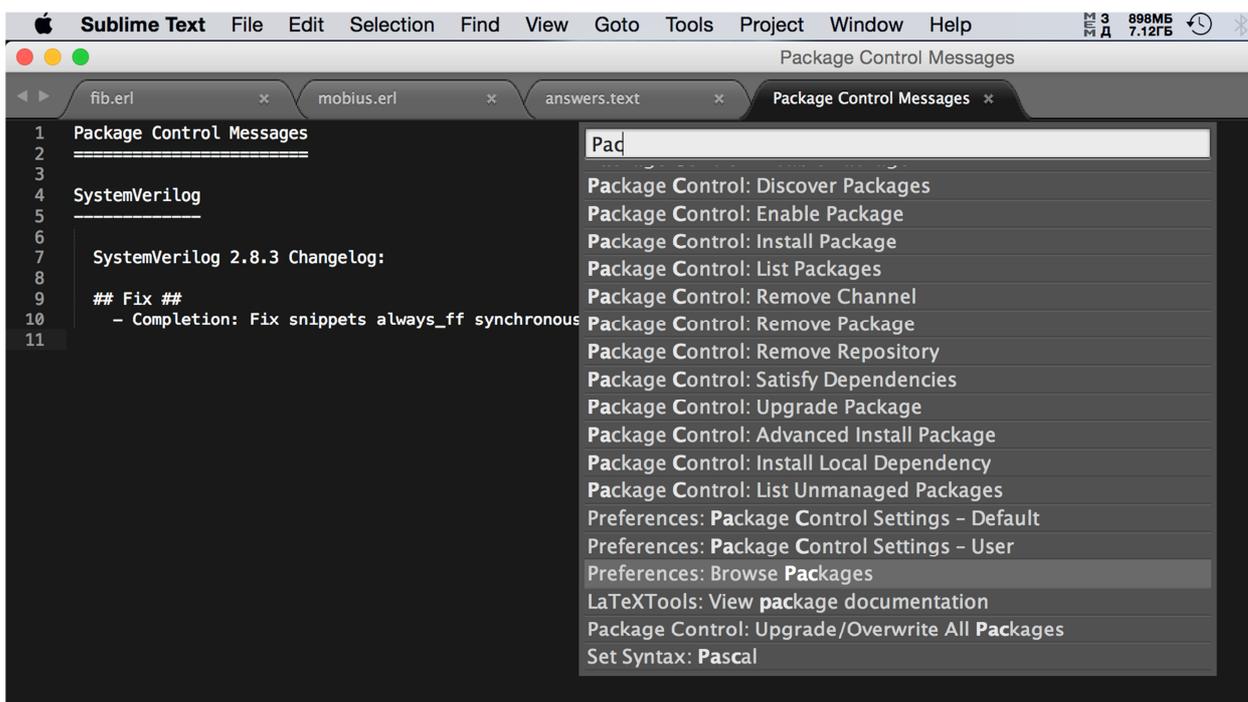


Рис. 2. Пример интерфейса с контролируемым естественным языком

Заключение

Интерфейс на контролируемом естественном языке являются одним из перспективных направлений в построении интерфейсов. Но у такого метода есть ряд недостатков и ограничений, которые можно решить при помощи создания фреймворка, который сможет генерировать контролируемый естественный язык для определённого приложения. В дальнейшем планируется разработать подобный фреймворк.

Список литературы

- [1]. A Controlled Natural Language Interface for Semantic Media Wiki Using the Rabbit Language. Available at: <http://eprints.soton.ac.uk/267190/1/SemanticWikiv7.pdf>, accessed 02.02.2016.
- [2]. CLIF Playground. Available at: <http://project.mit.bme.hu/clif/>, accessed 02.02.2016.
- [3]. Тарасенко С. В., Рязанова Н. Ю. Анализ методов автоматической генерации вопросов на естественном языке // Инженерный вестник. МГТУ им. Н.Э. Баумана. Электрон. журн. 2015. № 12. Режим доступа: <http://engbul.bmstu.ru/doc/829687.html> (дата обращения 02.02.2016).