ITHEA

International Journal INFORMATION MODELS & ANALYSES

2018 Volume 7 Number 4

International Journal
INFORMATION MODELS AND ANALYSES
Volume 7 / 2018, Number 4

IJ IMA is official publisher of the scientific papers of the members of the ITHEA® International Scientific Society. IJ IMA rules for preparing the manuscripts are compulsory.

The **rules for the papers** for ITHEA International Journals are given on www.ithea.org .

The papers should be received by ITHEA® Submission system http://ij.ithea.org .

Responsibility for papers published in IJ IMA belongs to authors.

International Journal "INFORMATION MODELS AND ANALYSES" Volume 7, Number 4, 2018

Edited by the **Institute of Information Theories and Applications FOI ITHEA**, Bulgaria, in collaboration with:
University of Telecommunications and Posts, Bulgaria,
V.M.Glushkov Institute of Cybernetics of NAS, Ukraine,
Universidad Politécnica de Madrid, Spain, Hasselt University, Belgium, University of Perugia, Italy,
Institute for Informatics and Automation Problems, NAS of the Republic of Armenia
St. Petersburg Institute of Informatics, RAS, Russia,

Publisher: **ITHEA®** Sofia, 1000, P.O.B. 775, Bulgaria. www.ithea.org , e-mail: office@ithea.org
Technical editor: **Ina Markova**

Printed in Bulgaria

ISSN 1314-6416 (printed) **ISSN 1314-6432 (Online)**

USING OF "INTERNATIONAL COMPONENTS FOR UNICODE" FOR APPLIED TASKS OF COMPUTATIONAL LINGUISTICS (CASE OF STRUCTURAL TEXT ANALYSIS)

Yuriy Koval, Maxim Nadutenko

(in Russian)

Abstract: *The present article deals with the peculiarities of character encoding in information-communication systems technologies. Article provides historical information about different systems, approaches and standards of character encoding starting from ASCII to Unicode. Reasons and conditions of the development of each standard have been provided accordingly. Unidoce has been defined as common system for character encoding, counter-compatibility with ASCII and characters encoding method of Unicode have been illustrated. International Components for Unicode (ICU) has been discovered as common instrument for structural text analysis. List of functionality of ICU has been provided as well as spheres of using of ICU.*

The present article provides short description of Swift programming language and interfaces of ICU which are used in this programming language for dealing with regular expressions. Logic for interlanguage text transformation in the Swift described. Detailed process of integration of ICU for using with Swift programming language has been provided.

The present article contains information about applications of text boundary analysis, types of boundaries and principles of text processing by BreakIterator ICU classes, which all listed in the Standard Annex #29 (Unicode text segmentation). Unicode Common Locale Data Repository has been defined as a main source of locale information for specific regional rules, dedicated for specifying the rules for text boundary analysis.

Icu4c-swift framework has been used for manipulation of ICU logic with Swift programming language. Such text boundaries as words and sentences have been marked with specific markers. Correlation between time of marker insertion, text size in memory, quantity of symbols in text has been provided in the corresponding tables and graphs. Special attention has been paid to the usage of Swift.String.index(_:offsetBy:) function. Correct usage of such function can increase the speed of program execution dramatically.

Advantages and disadvantages of using the International Components for Unicode with Swift programming language have been analyzed. The emphasis was made on using specific functions of Swift with International Components for Unicode for text analysis.

***Keywords**: character encoding, Unicode, structural text analysis, computational linguistics.*

Использование “International Components for Unicode” в прикладных задачах компьютерной лингвистики (на примере структурного анализа текста)

Юрий Коваль, Максим Надутенко

***Abstract**: В данной статье рассматриваются особенности кодирования символьных данных в системах информационно-коммуникационных технологий. В качестве стандарта для кодирования символов был выбран Unicode, а International Components for Unicode как основной инструмент для структурного анализа текста. Были проанализированы и показаны преимущества и недостатки использования International Components for Unicode совместно с языком программирования Swift. Предложены рекомендации по применению функционала Swift для работы с текстом с помощью International Components for Unicode.*

Ключевые слова: *кодирование символов, Unicode, структурный анализ текста, компьютерная лингвистика*

***ITHEA Keywords:** D.2.3 Coding Tools and Techniques, I.2.7 Natural Language Processing*

Введение

Текст состоит не из слов, а из словосочетаний. При взаимодействии человека с текстом, границы в нем помогают адекватно воспринимать последовательность символов и интерпретировать эту последовательность в собственном когнитивном аппарате как информацию. Именно эти границы помогают «увидеть» слова и сложить их в словосочетания, присвоив последовательности символов семантику. Поскольку задачи обработки текста сейчас все больше передаются от человека компьютеру, возникает проблема в адекватном обучении компьютера определять границы в тексте. Для человека, который работает с текстом на компьютере, в свою очередь, вызовом является правильный выбор системы для обработки символов. От правильного выбора такой системы зависит достижение целей, которые ставит перед собой человек, обрабатывая текст, ведь результат работы различных систем обработки символов может отличаться.

Так, практически любой специалист по компьютерной лингвистике в процессе профессиональной деятельности сталкивается с проблемой кодирования текстовой информации и токенизации текста, то есть, разбиения текста на слова, предложения и тому подобное. Например, символ Emoji, с изображением семьи из мужчины, женщины, мальчика и девочки в одной системе, будет представлен как отдельные символы человека мужского пола, человека женского пола, ребенка мужского пола и ребенка женского пола в другой системе из-за различий в кодировке символов. Другой пример можно взять из структурной разметки текстов, где чтобы выделить прямую речь, диалоги, слова автора, части предложения и другие необходимые для дальнейшей обработки текста единицы, необходимо сначала выполнить разбиение текста на предложения, слова и тому подобное. Хотя вышеупомянутые задачи и имеют много имеющихся на сегодняшний день путей решения, но все равно являются нетривиальными, а конкретные решения зависят от использования конкретных систем обработки естественного языка, языков программирования и даже операционных систем.

Целью статьи является исследование использования преимуществ International Components for Unicode (ICU) на практике, а именно применение ICU совместно с языком программирования Swift и выявления положительных и отрицательных сторон использования инструментов ICU для задач структурной разметки текста, предоставление практических рекомендаций для разбиения текста на слова и предложения. ICU является библиотекой программных средств консорциума Unicode предназначенной для обработки текста, символьных данных, решения проблем интернационализации, временных поясов, календарей, дат и др. [2]

Поскольку каждая платформа имеет свои требования и стандарты функционирования, имеет смысл использовать кроссплатформенное решение для работы с текстом, которое будет вести себя одинаково независимо от стандартов конкретной платформы, или разработать один стандарт интерфейса, который будет иметь нативную реализацию подобно BLAS (Basic Linear Algebra Subprograms). Здесь и выходит на передний план ICU, которая выступает единственной системой для адекватной обработки символьных данных и которая по умолчанию используется различными платформами, операционными системами и языками программирования.

Кроме "источника" стандартизированных подходов к кодировке символьных данных, группа разработчиков ICU предоставляет и постоянно совершенствует алгоритмы для работы с текстом, в частности алгоритмы токенизации. Конечно, для решения специфических задач может быть целесообразной реализация собственного программного решения, но тогда возникнет необходимость реализации функционала, который уже содержится в ICU и прошел испытание временем.

Другим аргументом в пользу использования ICU явилась способность библиотеки обрабатывать тексты, написанные на разных языках. Разработка необходимого функционала для работы, скажем, с кириллицей не является слишком сложной задачей, но если возникает необходимость в обработке текстов на латинице, китайских иероглифов, то сложность системы значительно возрастает, и требуется поиск решения множества различных подзадач с большим количеством исключений из правил.

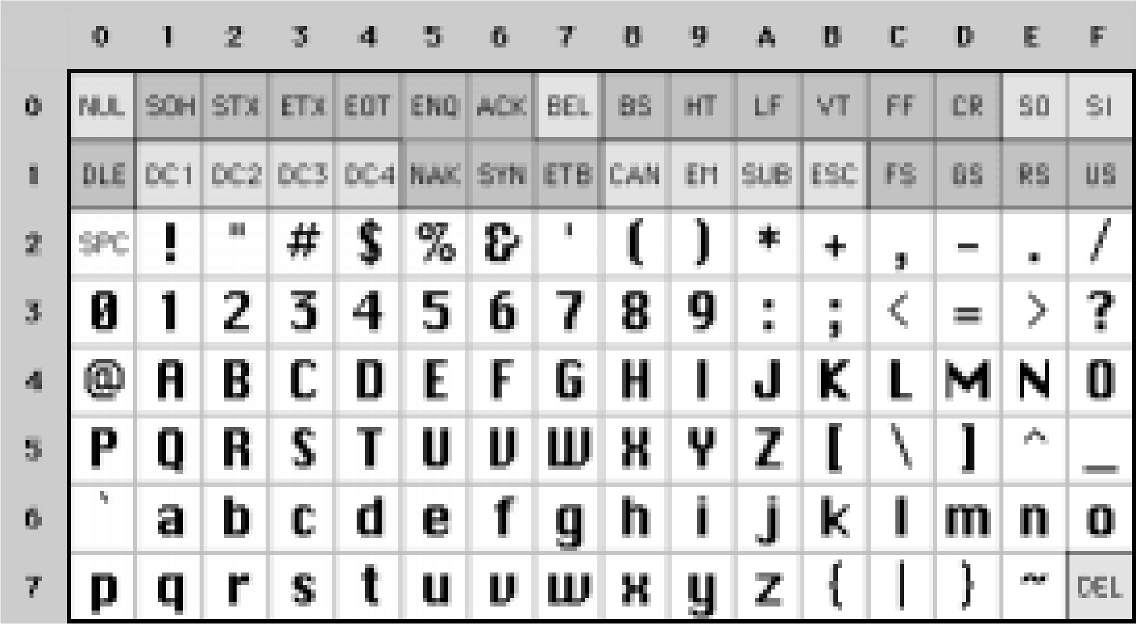
International Components for Unicode написана на низкоуровневом языке программирования С, что позволяет получить более высокую скорость работы библиотеки по сравнению с другими библиотеками, для которых не проводилась фундаментальная многолетняя работа по оптимизации использования памяти и улучшению быстродействия.

Краткие сведения о International Components For Unicode

Исторические сведения о International Components for Unicode

Как указано на сайте разработчика: «ICU - это фундаментальный, широко используемый набор библиотек C/C++ и Java, обеспечивающих поддержку Unicode и Globalization для программных приложений. ICU портативен и обеспечивает одинаковые результаты для приложений, выполняющихся на различных платформах и написанных на языках C/C++ или Java» [2]. Разработке данной библиотеки предшествовал ряд исторических событий, связанных с представлением символьных данных в компьютерных системах.

Первым стандартом для размещения набора символов была таблица ASCII (American Standard Code for Information Interchange, Американский стандартный код для информационного обмена), которая содержала 33 сервисных символа с кодами от 0 до 31, а также 95 печатных символов ASCII с кодами от 32 до 126 (рис.1).

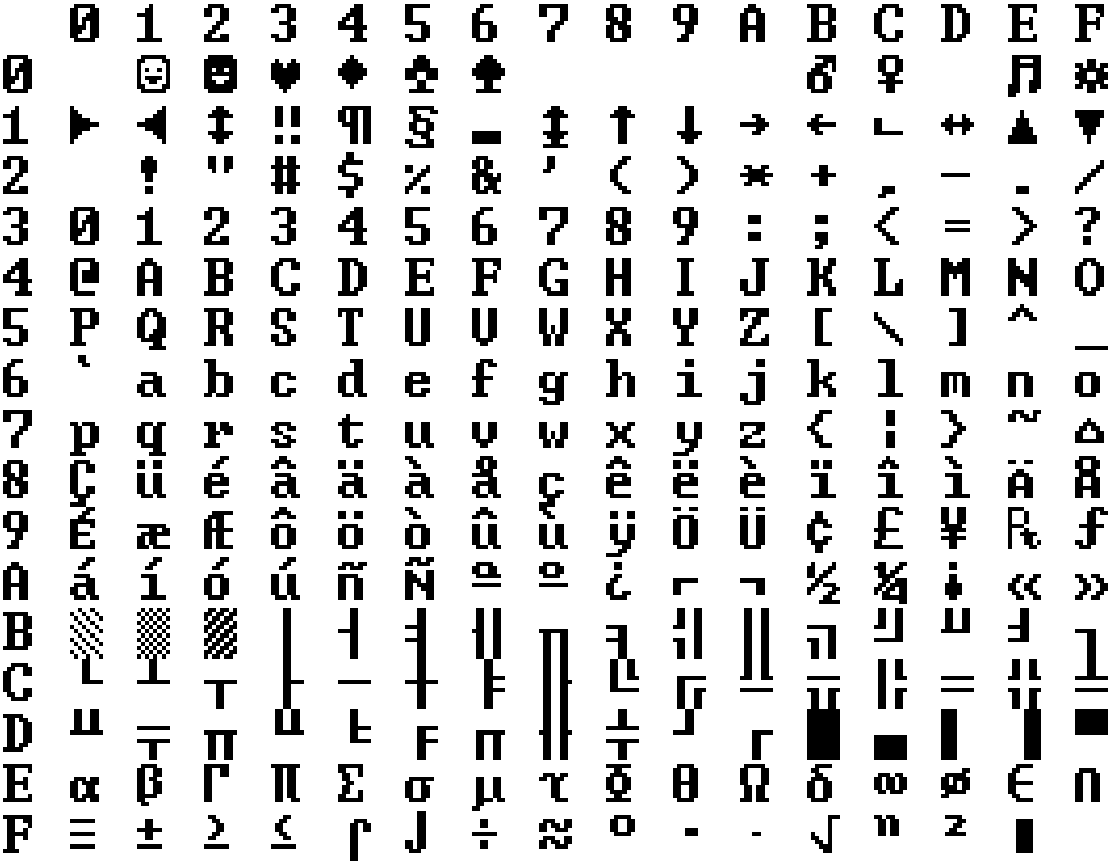


	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NUL	SOH	STX	ETX	EOT	ENQ	ACK	BEL	BS	HT	LF	VT	FF	CR	SO	SI
1	DLE	DC1	DC2	DC3	DC4	NAK	SYN	ETB	CAN	EM	SUB	ESC	FS	GS	RS	US
2	SPC	!	"	#	$	%	&	'	(	)	*	+	,	-	.	/
3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
6	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7	p	q	r	s	t	u	v	w	x	y	z	{	|	}	~	DEL

Рис. 1. ASCII - Американский стандартный код для информационного обмена

Многие решения на стадии разработки функционала для работы с символьными данными принималось исходя из требований, которые диктовались аппаратным обеспечением той или иной платформы [1]. Например, символу Delete в таблице ASCII соответствует код x'7F ", из-за необходимости выбить все отверстия в колонке перфокарты, чтобы сигнализировать, что колонка должна быть проигнорирована.

ASCII можно было вместить в 7 бит, а большинство компьютеров того времени использовали 8-битную систему. Все происходило хорошо до тех пор, пока не появлялось необходимости в использовании символов не английского алфавита. Так как компьютеры использовали 8 бит, а вся таблица занимала 7 - во многих одновременно возникла идея задействовать целый бит в собственных целях (а это коды от 128 до 255). Это, конечно, создало определенный хаос в передаче информации от одного компьютера другому, так как коды после 128 в одной системе отличались от кодов на тех же позициях в другой. Тогда IBM-PC ввела так называемые «кодовые страницы» (code pages), которые впоследствии были переименованы в наборы символов (character set) OEM (Original Equipment Manufacturer), которые покрыли символы большинства европейских языков, использующих латинский алфавит. Кроме символов алфавита, наборы символов IBM-PC OEM содержали в себе символы для рисования (линии, двойные линии, наклонные линии и т.д.) рис. 2.



	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0		☺	☻	♥	♦	♣	♠					♂	♀		♫	☼
1	►	◄	↕	‼	¶	§	▬	↨	↑	↓	→	←	∟	↔	▲	▼
2		!	"	#	$	%	&	'	(	)	*	+	,	-	.	/
3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
6	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7	p	q	r	s	t	u	v	w	x	y	z	{	¦	}	~	⌂
8	Ç	ü	é	â	ä	à	å	ç	ê	ë	è	ï	î	ì	Ä	Å
9	É	æ	Æ	ô	ö	ò	û	ù	ÿ	Ö	Ü	¢	£	¥	₧	ƒ
A	á	í	ó	ú	ñ	Ñ	ª	º	¿	⌐	¬	½	¼	¡	«	»
B	░	▒	▓	│	┤	╡	╢	╖	╕	╣	║	╗	╝	╜	╛	┐
C	└	┴	┬	├	─	┼	╞	╟	╚	╔	╩	╦	╠	═	╬	╧
D	╨	╤	╥	╙	╘	╒	╓	╫	╪	┘	┌	█	▄	▌	▐	▀
E	α	ß	Γ	π	Σ	σ	µ	τ	Φ	Θ	Ω	δ	∞	φ	ε	∩
F	≡	±	≥	≤	⌠	⌡	÷	≈	°	∙	·	√	ⁿ	²	■	

Рис. 2. Набор символов IBM-PC OEM

С момента, когда персональные компьютеры стали покупать за пределами США, все необходимые для обмена информацией символы локальных алфавитов стало возможно разместить в позиции выше 128 (первые 128 формально было договорено выделять для ASCII). Это и стало моментом создания стандарта ANSI, с различными системами кодов, которые уже здесь получили название "кодовые страницы". Например, в Израиле использовалась кодовая страница 862, в тот момент как в Греции – 737, то есть каждая национальная операционная система MS-DOS содержала в себе специфическую для региона страницу кодов. Существовали даже страницы "межнациональной" языка – эсперанто. Но для того, чтобы объединить в одном компьютере, скажем, украинский и немецкий языки, нужно было писать собственную программу для отображения различных символов.

Что касается систем письменности азиатский языков, которые имеют тысячи символов, поместить их в 8 бит явно не представлялось возможным. Поэтому использовалась система под названием DBCS (double byte character set) - набор двухбайтных символов, где иногда один символ занимал один байт, иногда - два. Чтобы исключить перевод каретки на байт, лежащий в пределах одного символа, не рекомендовалось проходить по символам обычным итератором (как вперед, так и назад). Взамен предлагалось использовать функции вроде Windows.AnsiNext и Windows.AnsiPrev. В основном символы обрабатывались исходя из правила "один символ занимает один байт", которое работало хорошо до того момента, когда приходилось передать текст с одного компьютера на другой, что стало насущной проблемой с началом использования Интернета.

Для того, чтобы сохранить единый функционал в старых и новых системах, а также из-за ограниченной вместимости таблицы символов ранних вычислительных систем, группа разработчиков системы Unicode наложила много ограничений на таблицы кодировок и на сам функционал обработки символьных данных.

Понимая необходимость плавного и наименее болезненного перехода на новую систему кодирования, разработчики Unicode сохранили соответствие с фактически действующим на то время стандартом ASCII. На пример, в ASCII символ “H” имел позицию 48. В Unicode он обозначается такой последовательностью как “U+0048”, что называется единицей кодирования (code point). Если посмотреть на кодирование слова “Hello”, то можно увидеть четкую аналогию между единицами кодирования и позициями в ASCII:

U+0048 U+0065 U+006C U+006C U+006F

Что касается символов, которые не входят в английский алфавит, то эту проблему разработчики Unicode разумно решили, выделив несколько единиц кодирования для интерпретации одного символа. То есть ограничение в 8 бит было снято и появилась возможность представить любой символ, выделив на него (при необходимости) несколько байт. Так, большинство символов Emoji и китайской системы письменности могут занимать до четырех байт.

После создания стандарта для хранения символов и унификации их кодирования, стала возможной адекватная передача и однозначная интерпретация текстовой информации, написанной на различных языках. После этого возникла потребность в автоматизации обработки последовательностей символов в тексте, которые образуют слова, предложения, даты и прочее. Более того, возникла необходимость делать это с учетом особенностей обработки такого рода информации в различных культурах и отдельных странах. Решения для этих проблем вышли в свет в 1999 году с выпуском International Components for Unicode (конечно, не окончательные, но необходимые для функционирования систем глобализации и интернационализации в различных языках программирования и информационных продуктах).

Чтобы лучше понять возникновение системы Unicode и проекта International Components for Unicode, а также предпосылки создания наборов стандартных символов, которые использовались и продолжают использоваться в компьютерных системах и алгоритмов обработки текстов, стоит рассмотреть этот процесс в хронологическом порядке:

- 1986-1987 разработка шрифтов для китайского языка в Xerox, Apple присоединяется к ANSI X3L2 и работает над универсальным набором символов;
- Февраль 1988 – в Apple начинают работать над 16-битной системой кодирования "High Text";
- Апрель 1988 – в Apple разработаны первые прототипы кодировки текста в Unicode;
- Сентябрь 1988 – Apple покупает базу данных азиатских символов (CJK – Chinese, Japanese, Korean) для работы с базой данных Han;
- Февраль 1989 – начало регулярных встреч между Sun, Adobe, HP, NeXT, Pacific Rim Connections для работы над Unicode;
- Август 1989 – Xerox и Apple объединяют базы данных Han для совместной работы над азиатскими символами;
- Октябрь 1989 – Unicode представляют для Microsoft и IBM на фоне сотрудничества Apple и Microsoft над TrueType;
- Май 1990 – презентация Unicode на глобальной конференции разработчиков Apple;
- Ноябрь 1990 – презентация Unicode на конференции IEEE;
- Январь 1991 – создание Unicode Consortium;
- Декабрь 1991 – создание базы данных UniHan;
- Июнь 1992 – публикация второго выпуска Unicode Standard Version 1.0;
- Начало 1999 – выход в свет "IBM Classes for Unicode", позже переименованного в International Components for Unicode [18].

Более подробную информацию можно найти в Chronology of Unicode Version 1.0 [17].

Итак, мы видим, что историю кодирования символов можно свести к утверждению трех основных стандартов: ASCII, ANSI и Unicode, последний из которых стал тем, который вобрал в себя все символы алфавитов и систем письменности на Земле.

Усилия, вложенные в создание Unicode и International Components for Unicode, явились основой для начала нового этапа в обмене информацией в "Сети сетей", унифицировав в одном месте набор символов, содержащий каждую из общепринятых систем письма (знаков) на планете. Таким образом, создание Unicode в начале 90-х годов XX века (а именно в июне 1992 [17]), сыграло ключевую роль не только в удобстве обмена информацией в Интернете, но и в переходе человечества ко второй волне сетевой эпохи развития.

Сферы применения International Components for Unicode

International Components for Unicode применяется в самых разных областях для обработки текстовой информации. Так, на сайте проекта ICU приводятся следующие основные (не полный перечень) функции, которые эта система предоставляет [1]:

- сравнение символов (в соответствии с выбранными правилами);
- представление множества символов юникода;
- сравнение порядка кодовых единиц (code points, code units);
- итерация по символам юникода;
- локаль (региональная специфика);
- сервисы даты и времени (календарь, временные зоны и т.д.);
- преобразование внутреннего представления даты в текстовое представление времени, атрибуты календаря и т.д.;
- работа с регулярными выражениями;
- "двусторонняя" обработка текста (обработка текста как с последовательностью символов слева направо, так и справа налево);
- анализ границ в тексте (определение позиций слов, предложений, абзацев в тексте и т.д.).

В данной статье больше внимания будет уделено токенизации и анализу границ в тексте, но стоит помнить, что основное, для чего следует использовать ICU, так это для идентификации символов, потому что она является наиболее полным "складом" кодовых единиц юникода, включая Emoji и CJKV (Chinese, Japanese, Korean, Vietnamese).

Очевидно, что многие языки программирования используют в своей основе логику ICU для работы с некоторыми задачами из приведенного выше списка, но в основном этот функционал ограничен и рекомендуется, при необходимости решить более узкоспецифическую проблему, использовать именно ICU [2], а не логику, взятую из конкретного языка программирования.

Итак, можно сделать вывод, что хотя и приведен далеко не полный список возможностей, которые предоставляет ICU, но из него понятно, что эта логика используется в большинстве систем, работающих с датами и календарями, локалью, кодированием символов систем письменности различных языков (включая системы с различным направлением чтения), что и обеспечивает набор средств для решения проблемы интернационализации и глобализации информационных технологий.

Интеграция ICU с системами обработки естественного языка

Интеграция со Swift programming language

Swift programming language – один из часто используемых высокоуровневых языков программирования на сегодняшний день, занимает 15 место в рейтинге TIOBE [5]. Представленный на World Wide Developers Conference в 2014 году и разработанный группой, возглавляемой Крисом Латнером, как язык программирования для операционных систем macOS, iOS, watchOS и tvOS [6], сейчас широко используется не только для выполнения своих первоочередных задач, но и для написания программ под операционную систему Linux, реализации back-end логики и т.д. [8]. Именно после перевода Swift programming language в открытый доступ, Swift начал использоваться для более широкого спектра задач.

Swift, как и многие другие языки, использует ICU для работы с текстом. То есть часть функционала ICU поставляется внутри стандартных библиотек, а конкретнее в Foundation framework. Например, Apple Inc. [9] приводит такие интерфейсы ICU, используемые в Foundation Framework для работы с регулярными выражениями (regular expressions):

parseerr.h
platform.h
putil.h
uconfig.h
udraft.h
uintrnal.h
uiter.h
umachine.h
uregex.h
urename.h
ustring.h
utf_old.h
utf.h
utf16.h
utf8.h

Олег Бегемана, разработчик и член комиссии по развитию Swift programming language, на своем веб-сайте [10] приводит пример, как ICU используется в Foundation Framework для текстовых трансформаций. В частности, он говорит о транслитерации, которая наглядно иллюстрирует, что ICU является прекрасным инструментом, когда нужно работать с символьными данными алфавитов разных языков:

```
import Foundation
let shanghai = "上海"
shanghai.applyingTransform(.toLatin, reverse: false)
// → "shàng hǎi"//результат после трансформации
```

Swift содержит библиотеку для анализа естественного языка с помощью методов машинного обучения, которая также решает задачи токенизации (Tokenization) [7]. Однако, Swift не содержит интерфейсов ICU, необходимых для выполнения задач структурной разметки текста, а именно его разбиения на предложения и слова. Из-за этого появляется необходимость использования логики ICU, к которой приходится обращаться конкретным объектам, написанным на Swift. Как готовое решение проблемы была использована библиотека icu4c-swift [11], в которой реализован доступ к ICU через такие объекты-обертки как CharacterBreakCursor, LineBreakCursor, RuleBasedBreakCursor, SentenceBreakCursor, WordBreakCursor и другие.

Если же есть необходимость в использовании ICU через Swift напрямую, то это можно сделать, импортировав необходимые интерфейсы в собственноручно созданный модуль, а саму библиотеку ICU установить в операционную систему как отдельный компонент, отличный от того, который уже используется системой (если есть необходимость использовать определенную версию ICU и системная версия не содержит нужных интерфейсов). То есть выглядит это следующим образом:

1. Создание "упаковки" модуля для установки ICU. Файл "упаковки" выглядит следующим образом:

```
  Let package = Package (name: "icu4c-swift", pkgConfig: "icu-uc", providers: [ .Brew("icu4c"),
.Apt("libicu-dev"), ])
```

2. Создание непосредственно файла модуля:

```
module ICU4C {
header "umbrella.h"
link "icucore"
export *
}
```

3. Создание файла для импорта необходимых интерфейсов с ICU, так называемого "зонта" для интерфейсов:

```
//umbrella.h
#import <unicode/icudataver.h>
#import <unicode/parseerr.h>
#import <unicode/platform.h>
```

После выполнения этих шагов можно использовать ICU-логику в собственных объектах. Важно помнить, что ICU включает в себя множество функционала, который является стандартом и используется в различных языках программирования. Не является исключением и Swift. Те функции, которые использует сам язык программирования, называются приватными и их использование "напрямую" может рассматриваться с стороны Apple как несанкционированное проникновение в нативное API, поэтому, соответственно, Apple может отказать в публикации приложения в AppStore на этапе его рассмотрения. К такому API, кроме прочего, относятся такие компоненты как ubrk_current, ubrk_first, ubrk_next, используемые в ICU для разбиения текста. Выходом из такой ситуации может быть использование не системного ICU, а статически скомпилированной библиотеки.

Итак, мы видим, что Swift programming language содержит большое количество логики ICU для работы с символьными данными и текстом в библиотеке Foundation, но если нужно разбивать текст на предложения, слова и т.д., то можно использовать недостающую в Swift ICU-логику, которую можно задействовать как через прямой доступ к системной ICU, содержащейся в операционной системе, так и собрав отдельный модуль, что безопаснее, благодаря гарантированного не использования нативного API, а также работе с конкретной версией библиотеки.

Использование International Components For Unicode для структурной разметки текста

Разбиение текста на слова

Анализ границ текста с помощью ICU (определение лингвистических границ при форматировании и обработке текста) включает в себя [4]:

1. определение позиции соответствующих точек для обрамления текста в слово для помещения между специфическими отступами при отображении или печати;
2. определение начала слова, выбранного пользователем;

3. подсчет знаков, слов, предложений, абзацев;
4. определение, как далеко переставить курсор при нажатии на клавишу стрелки (некоторые символы занимают более одной позиции в тексте, а некоторые символы не отображаются вообще);
5. создание списка уникальных слов в тексте;
6. капитализицзацию первой буквы слов;
7. определение отдельной единицы в тексте, слове (десятое слово второго абзаца и т.д.).

Классы BreakIterator в ICU были созданы именно для подобных задач. Существует четыре типа границ в тексте, в отношении которых применяются итераторы ICU:

1. границы знаков;
2. границы слов;
3. границы разрыва линий;
4. границы предложений.

Каждый тип границ определен в соответствии с правилами, описанными в Приложении номер 29 к Стандарту Unicode [12], а также с алгоритмом разбиения строк того же Unicode [13].

Итератор символьных границ определяет границы согласно правилам границ графемного кластера из вышеупомянутого Приложения номер 29. Границы определены таким образом, чтобы соответствовать классификации "что такое символ" самого пользователя, то есть базовой единицы системы письменности для конкретного языка, символ которой часто может занимать более одной единицы кодирования (code point) в системе Unicode. В качестве примера на странице инструкции по использованию итераторов разбиения ICU [4] приводится буква Ä, которая может занимать как одну единицу кодирования, так и две, когда в первой сохраняется буква A, а во второй хранится ее умлаут (диакритический знак в немецком и других языках, имеет вид двух точек над буквой). В обоих случаях это будет расценено как один символ.

Итератор словесных границ определяет позиции границ слов. Примерами применения итератора словесных границ являются: выделение слова при двойном щелчке клавишей мыши на нем, поиск полных слов, подсчет количества слов в тексте.

Границы слов также определены в Приложении 29 к стандарту Unicode, но в случае для слов они дополнены словарями для китайского, японского и других языков. Важной и, наверное, основной особенностью итератора словесных границ является то, что им принимаются во внимание алфавиты и традиции правописания различных языков.

Для разработки программы для структурной разметки текста мы использовали библиотеку ICU4C-swift, где класс WordBreakCursor является оберткой над итератором словесных границ ICU.

Если же говорить о быстродействии, то очень наглядно демонстрируется быстродействие работы самой ICU по сравнению с высокоуровневыми структурами в Swift. Прохождение курсора по границам слов в тексте занимает не так много времени, как вставка символов и поиск позиции в тексте с помощью функции String.index (_: offsetBy :). Сложность функции указана в документации [14] как O(n), но если использование данной функции еще возможно для небольших объемов текста, то с увеличением количества слов в тексте, время выполнения программы возрастает до критического уровня (Рис. 3):

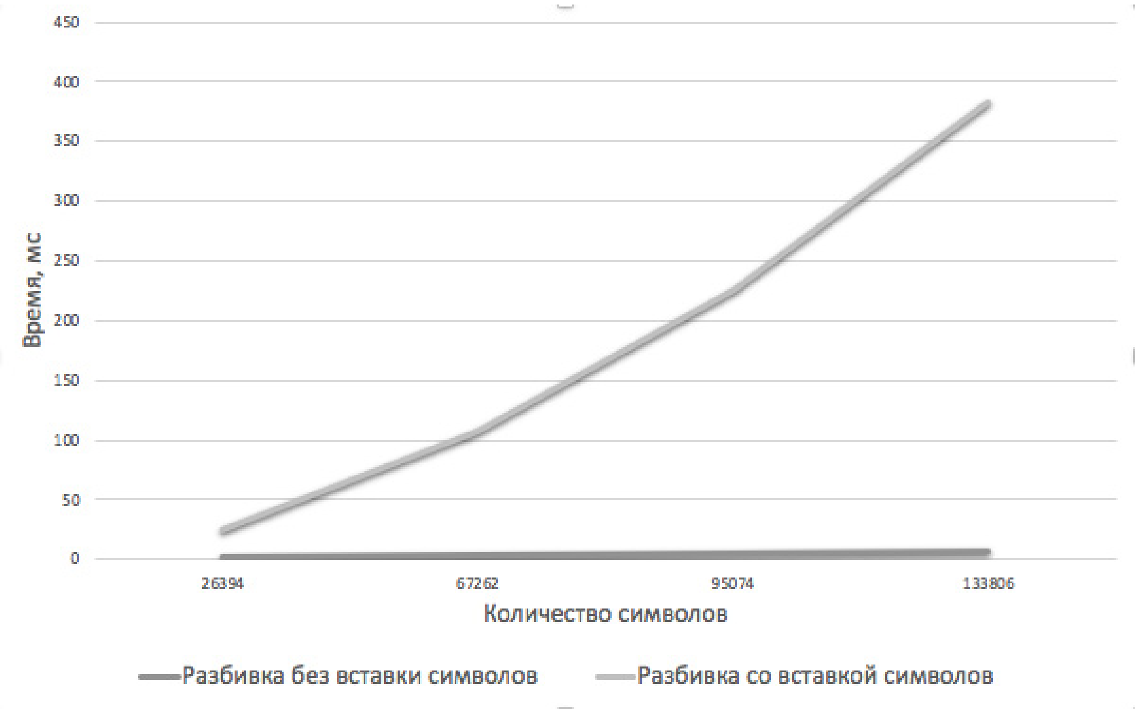


Рис. 3. Зависимость времени разбиения на слова от количества символов в тексте

Для анализа использовался текст Конституции Украины, который был условно поделен на четыре части. При этом время разбиения без вставки символов выросло с 5,9 миллисекунд до 28,6 миллисекунд при увеличении количества символов с 26394 до 133806. То есть с увеличением количества символов в тексте в 5,07 раза, время разбиения выросло в 4,85 раза, что не является удовлетворительным результатом, но приемлемо для дальнейшего совершенствования написанной программы.

Что касается разбиения текста на слова со вставкой символов структурной разметки, то время выросло с 5509,3 миллисекунд до 130395,1 миллисекунд при соответствующем росте количества символов. То есть с увеличением количества символов в тексте в те же 5,07 раза, время выросло в 23,7 раза, что говорит о том, что использование данного алгоритма никак невозможно без его улучшения и изменения.

Таблица 1. Зависимость времени разбиения текста на слова от количества символов в тексте

Количество знаков	Размер текста, байт	Время разбиения без вставки символов, мс	Время разбиения со вставкой символов, мс
26394	53590	5,9	5509,3
67262	136534	14,5	33662,6
95074	193082	19,9	66060,6
133806	271536	28,6	130395,1

Из таблицы 1 видно, что наибольший прирост времени наблюдается в момент вставки символов. При рассмотрении этой операции было проанализировано время, которое выделяется на каждый ее этап, так как это время изменяется при изменении количества символов в обрабатываемом тексте. Из графиков на рис. 4 видно, что время вставки символов является относительно устойчивым – среднее значение равно $3,12 \times 10^{-8}$ миллисекунд. Что касается времени определения позиции для вставки символов, то эта величина возрастает пропорционально росту количества символов в тексте начиная от 5×10^{-9} до $6,23 \times 10^{-6}$ (то есть почти в тысячу раз). Причины: функция String.index (_: offsetBy :) выполняет важную для программиста работу – определяет позицию в тексте, учитывая количество code points, которые выделяются на один символ в тексте (как это было показано в примере с буквой Ä). То есть если текст состоит из десяти символов и необходимо вставить новый символ после символа номер 7, то функция String.index (_: offsetBy :) определяет позицию в тексте, в которой необходимо выполнить вставку символа, и возвращает именно количество символов, а не code points. Даже, если для отображения символа выделяется 3-4 юникодовских code points (например на флаги в символах Emoji), функция распознает Unicode последовательность и интерпретирует ее как один символ, то есть именно так, как видит перед собой пользователь. Это очень полезно, в первую очередь не только потому, что не нужно распределять code points по символам "вручную", но и из-за невозможности вставки нового символа между code points, что преобразует его в другой символ и разрушит имеющийся. Но это имеет свою цену и требует прохода через весь текст, в котором нужно определить позицию для вставки символа. Итак, если каждый раз вызывать функцию String.index (_: offsetBy :) для вставки символа в n позицию от начала и каждый раз проходить линейно по тексту от начала до n, что, в свою очередь, будет занимать много времени на больших объемах текста с большим количеством вставок, то нужно искать подходы к оптимизации логики обработки текста. Основной целью такой оптимизации должно быть уменьшение текста, который должен быть обработан функцией String.index (_: offsetBy :).

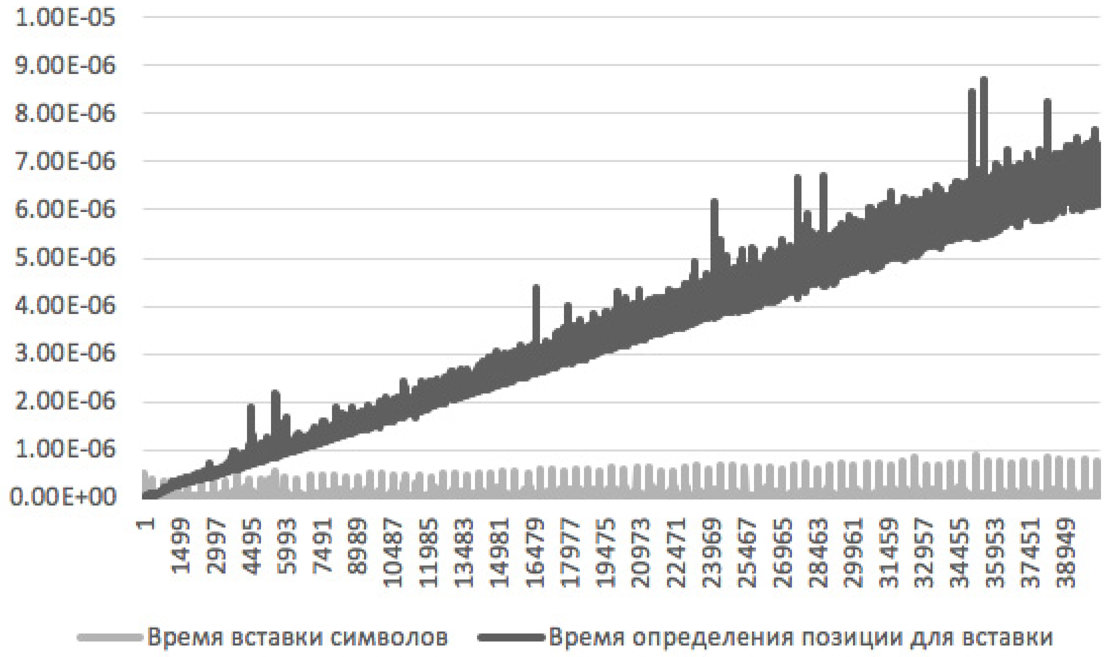


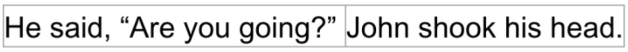
Рис. 4. Время определения позиции для вставки и вставки символов, миллисекунд

После оптимизации алгоритма, применяемого для разбиения текста на слова, время работы программы снизилось с более чем двух минут до 800 миллисекунд, то есть скорость обработки текста увеличилась почти в 170 раз.

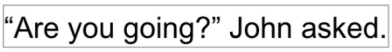
Итак, нами было исследовано разбиение текста на слова с помощью языка программирования Swift и ICU и получен неудовлетворительный результат, причина которого заключалась в использовании функции String.index (_: offsetBy :) ненадлежащим образом, что и было исправлено. Можно сделать следующий вывод: сочетание Swift с ICU увеличивает быстродействие, что является преимуществом такой системы.

Разбиение текста на предложения

Разбиение текста на предложения используется для таких прикладных, ориентированных на пользователя задач, как, например, выделение предложения после тройного щелчка клавишей мыши на любом фрагменте предложения, на котором находится указатель мыши; выделение части текста, в которой более одного слова. Определение границ предложений даже в простом тексте не является однозначно решенной задачей. Например, точка, которую принято считать концом предложения в украинском правописании, может быть знаком в десятичной дроби, а кавычки в прямой речи в английских текстах могут указывать как на конец предложения, так и на окончание прямой речи без окончания основного предложения. Подобный пример приведен в Приложении 29 к стандарту Unicode [12]:



He said, “Are you going?” John shook his head.



“Are you going?” John asked.

Символы <?, “, пробел, прописная буква> в первом предложении примера являются указателями на конец предложения, а во втором первом предложении примера расположены в середине предложения. Без семантического анализа достаточно сложно правильно определить границы предложений, и даже после семантического анализа часто остаются неопределенности, но обычно, подход, реализованный в алгоритме разбиения текста на предложения ICU, работает удовлетворительно.

Очень важным моментом при определении границ предложений является их локализация, которая очень сильно влияет на результат работы алгоритмов ICU. Активация или деактивация какой-то части логики может быть выполнена как вручную разработчиком, который использует ICU, так и выполняться автоматически с помощью локализации. Для этого консорциумом Unicode был создан «Репозиторий данных общей локализации» [15], в котором можно найти информацию о том, как та или иная логика работает в зависимости от текущей локализации. Репозиторий данных общей локализации (CLDR - Unicode Common Locale Data Repository) широко используется корпорациями, продукты которых зависят от региональных стандартов. Вот краткий список организаций, который приводится на сайте CLDR:

- Google (Web Search, Chrome, Android, Adwords, Google+, Google Maps, Blogger, Google Analytics, …)
- IBM (DB2, Lotus, Websphere, Tivoli, Rational, AIX, i/OS, z/OS,…)
- Microsoft (Windows, Office, Visual Studio, …)

Кроме локализации в Приложении 29 к стандарту Unicode [12] также приведены знаки членов группы Sentence_Break (границы предложения), которыми обычно являются знаки препинания, перевод каретки и тому подобное. Перечень группы Sentense_Break можно найти в таблице 4 Приложения 29 к стандарту Unicode. Интересно, что там приведены и знаки препинания группы SContinue, сигнализирующие о том, что предложение не закончилось, например, двоеточие, запятая, дефис и тому подобное. Полный перечень этой группы можно найти в той же таблице 4.

Также важную роль в итераторе границ предложений ICU играют правила их идентификации. По сути они являются повторяющимся сочетанием членов группы Sentence_Break и служат для задания макросов (набора инструкций) для правила определения границ предложений. Например, макрос "прервать предложение после терминаторов (элементов, сигнализирующих остановку в контексте), но включать закрытия пунктуации, конечные пробелы и любые разделители абзацев" обозначается как множество SB9 ... SB11. Полный список макросов приведены в таблице 4a "Sentence_Break Rule Macros" в Приложении 29.

Как и в случае с итератором для слов, итератор для разбиения предложений был реализован с помощью библиотеки ICU4C-swift, где класс SentenceBreakCursor является оберткой над итератором границ предложений в ICU. Быстродействие логики ICU опять же показало себя достаточно хорошо, чего нельзя сказать о высокоуровневой логике из библиотеки Swift programming language для поиска позиции в тексте и для вставки текста в конкретную позицию, о которой упоминалось в разделе о разбиении текста на слова. Для анализа был использован тот же текст Конституции Украины. Результаты оказались хотя и значительно лучшими, но подобные предыдущим: с увеличением количества символов в тексте в 5,07 раз, время разбиения текста на предложения без вставки символов выросло в 3,78 раза, что даже несколько лучше, чем в случае со словами (таблица 2, рис. 5). Но время разбиения текста на предложения, со вставкой символов, выросло в 15,9 раз, что немного и лучше, чем в случае со словами, но все равно говорит о том, что высокоуровневую логику со Swift нужно реализовать по-другому. Важно понимать то, что не сама логика в Swift является медленной, а конкретная реализация с ее использованием нуждается в доработке.

Таблица 2. Зависимость времени разбиения текста на предложения от количества символов в тексте.

Количество знаков	Размер текста, байт	Время прохода по тексту, мс	Время со вставкой символов, мс	Количество предложений
26394	53590	1,9	24	474
67262	136534	3,6	107,6	1151
95074	193082	5,5	225,8	1651
133806	271536	7,2	382,4	2224

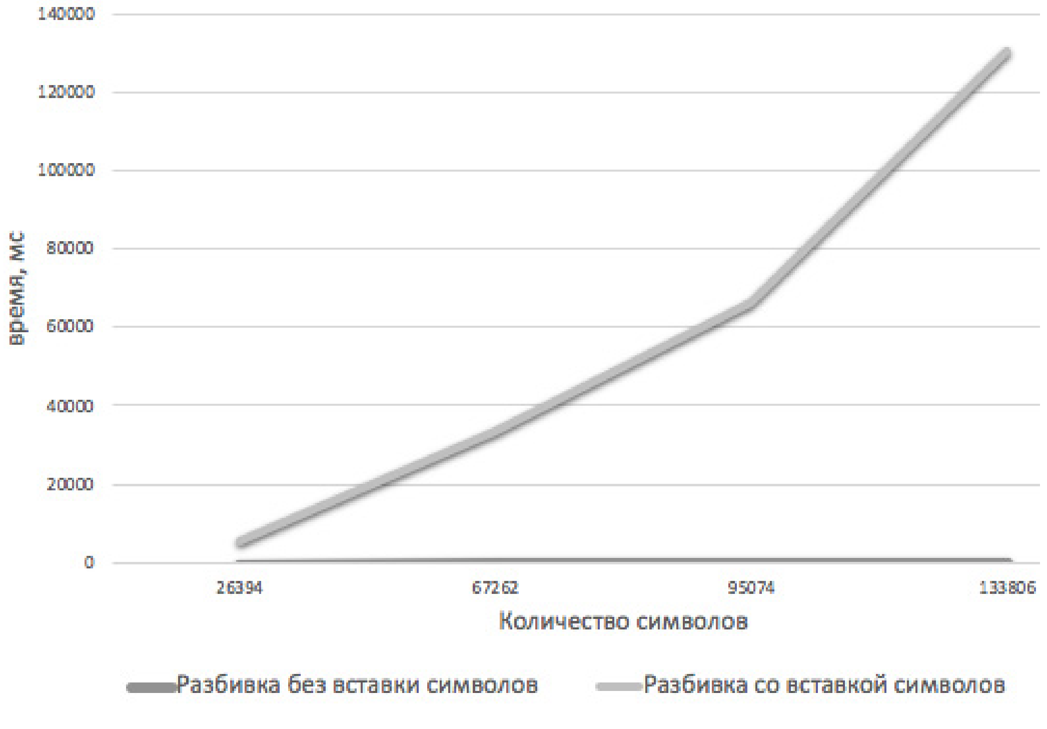


Рис. 5. Зависимость времени разбиения на предложения от количества символов в тексте

Подобно работе с определением границ слов в тексте, границы предложений размечаются определенными сигнальными знаками, используемыми для дальнейшей работы с размеченным текстом. Каждая вставка символов в текст – операция, требующая много времени. Поэтому алгоритмы разметки текста на предложения также были изменены для того, чтобы определять позицию для вставки в наименьшем отрезке текста. После модификаций время со вставкой символов уменьшился с 382 миллисекунд до 10 миллисекунд, что существенно улучшило работу программы.

Для тех, кто использует Swift programming language для работы с текстами, рекомендуется определение позиции символа в тексте (Swift.String.Index) на минимальной длине текста. Если в качестве входного параметра выступает число, показывающее расстояние от определенной позиции в тексте к желаемой позиции вставки, то время поиска желаемой позиции растет пропорционально величине этого числа.

В Swift (версии 4) текст представлен как "текстовая величина, состоящая из совокупности символов Unicode" [16]. Структура String в Swift является объектом абстрактного типа коллекции, которая позволяет работать с текстом как с массивом (или множеством) символов, поэтому при смещении каретки рекомендуется смещать ее на минимальное расстояние и сохранять прежнюю позицию каретки после конкретного элемента массива.

Итак, подобно предыдущему разделу, было исследовано разбиение текста на предложения с помощью языка программирования Swift и ICU и получен неудовлетворительный результат. Скорость разбиения, конечно, была выше, чем при разбиении на слова, но все равно недостаточно высокой для обработки текстов больших объемов. Ситуация изменилась к лучшему после усовершенствования логики формирования индекса для вставки символа разметки. Снова можно сделать вывод, что сочетание Swift с ICU дает более высокое быстродействие, что также является преимуществом такой системы и для разбиения на предложения.

Выводы

В данной статье были рассмотрены исторические сведения об International Components for Unicode, система Unicode, сферы применения ICU, использование ICU в языке программирования Swift, а также приведены результаты испытаний использования ICU для разбивки текста на слова и предложения.

Историю кодирования символов в компьютерных системах можно свести к утверждению трех основных стандартов: ASCII, ANSI и Unicode, последний из которых стал таким, который вобрал в себя все символы алфавитов и систем письменности на Земле, что сыграло важную роль для адекватного обмена информацией в сети Internet.

Логика ICU используется в большинстве систем, работающих с датами и календарями, локалью, кодированием символов систем письменности различных языков (включая системы с различным направлением чтения), что и обеспечивает набор для решения проблемы интернационализации и глобализации информационных технологий.

Язык программирования Swift использует логику ICU для работы с текстом, локалью и т.д, но поддерживает ограниченный набор функций, поэтому авторы рекомендуют использовать ICU как статически скомпилированную библиотеку, что будет удобно и безопасно при публикации программы в AppStore.

Разбиение текста на слова и предложения является достаточно сложной задачей с большим количеством правил и исключений, многие из которых реализованы в ICU, что и говорит о целесообразности использования этой системы. На языке программирования Swift удобно работать с текстом из-за того, что в нем он представлен как коллекция символов, состоящие из code-points. Поэтому разработчик работает с текстом, который он «видит». Важно правильно реализовывать формирования Swift.String.Index для вставки символов разметки, ведь неправильная реализация может увеличить время выполнения программы в сотни раз, как это видно из таблиц и графиков, приведенных в разделе «Использование International Components For Unicode для структурной разметки текста». Для разбиения текста использовались итераторы ICU, а логика программы была реализована на Swift.

Авторами были выявлены негативные аспекты конкретной реализации (как делать не следует) и даны рекомендации по минимизации времени выполнения программы (как необходимо делать).

Итак, если подытожить недостатки использования ICU, то они следующие:

1. требует много усилий для установки на OS Linux и для использования с языком программирования Swift;
2. специфическая реализация структуры String в Swift хоть и уберегает от опасности попасть между кодовых единиц Unicode одного символа, но требует формирования специфического индекса в тексте для позиционирования между символами;
3. при публикации продукта в AppStore его могут отклонить из-за использования приватных интерфейсов Apple, которые являются интерфейсами ICU.

Преимущества использования ICU:

1. кроссплатформенность;
2. содержит наборы символов, что является результатом работы над кодированием различных языков;

3. содержит фундаментальный функционал для работы с символьными данными, текстами (в частности, токенизации);
4. быстродействие.

Наличие единого "хранилища", которое могло бы вместить в себя все символы всех языков мира – серьезная проблема, с которой столкнулось человечество в процессе информационной революции. Отсутствие решения такой проблемы для систем информационно-коммуникационных технологий была бы равноценной коммуникации этих систем между собой "на разных языках". К счастью, Unicode не только стал этим единственным хранилищем, но и обеспечил единый механизм кодирования символов, и предоставил в определенной степени системам обмена информацией возможность идентичного отображения этой информации для человека с обеих сторон сети как до, так и после кодирования этой информации для ее передачи. International Components for Unicode, в свою очередь, на базе Unicode обеспечил дальнейшие шаги по направлению к интернационализации и глобализации, разработав стандарты и правила для работы с локалью календари и датами, границами в тексте и т.д., что обеспечило значительное ускорение перехода от информационной к сетевой эпохи человечества.

ЛИТЕРАТУРА

1. Feature Comparison Chart [Электронный ресурс] // International Components for Unicode – Режим доступа к ресурсу: http://site.icu-project.org/charts/comparison.
2. ICU-TC Home Page [Электронный ресурс] // International Components for Unicode – Режим доступа к ресурсу: http://site.icu-project.org/home#TOC-Who-Uses-ICU-
3. How Unicode relates to prior standards such as ASCII and EBCDIC [Электронный ресурс] // IBM Knowledge Center – Режим доступа к ресурсу: https://www.ibm.com/support/knowledgecenter/en/ssw_ibm_i_72/nls/rbagsunicodeandprior.htm.
4. ICU User Guide. Boundary Analysis [Электронный ресурс] // International Components for Unicode – Режим доступа к ресурсу: http://userguide.icu-project.org/boundaryanalysis
5. TIOBE Index for January 2019 [Электронный ресурс] // TIOBE Software Quality Assessment Company. – 2019. – Режим доступа к ресурсу: https://www.tiobe.com/tiobe-index/.
6. Swift programming language [Электронный ресурс] // Apple Inc. – Режим доступа к ресурсу: https://developer.apple.com/swift/.

7. Natural Language Framework [Электронный ресурс] // Apple Inc. – Режим доступа к ресурсу: https://developer.apple.com/documentation/naturallanguage.

8. Server-side Swift - Perfect [Электронный ресурс] // Perfectly Soft – Режим доступа к ресурсу: https://perfect.org/.

9. Lower Level Text-Handling Technologies [Электронный ресурс] // Apple Inc. – Режим доступа к ресурсу: https://developer.apple.com/library/archive/documentation/StringsTextFonts/Conceptual/TextAndWebiPhoneOS/LowerLevelText-HandlingTechnologies/LowerLevelText-HandlingTechnologies.html.

10. ICU Text Transforms in Foundation [Электронный ресурс] // Ole Begemann's website – Режим доступа к ресурсу: https://oleb.net/blog/2016/01/icu-text-transforms/.

11. ICU for Swift [Электронный ресурс] // Tony Allevato – Режим доступа к ресурсу: https://github.com/allevato/icu-swift.

12. UNICODE TEXT SEGMENTATION [Электронный ресурс] // Unicode® Standard Annex #29 – Режим доступа к ресурсу: http://www.unicode.org/reports/tr29/#Word_Boundaries.

13. UNICODE LINE BREAKING ALGORITHM [Электронный ресурс] // Unicode® Standard Annex #14 – Режим доступа к ресурсу: http://www.unicode.org/reports/tr14/.

14. Swift programming language documentation [Электронный ресурс] // Apple Inc. – Режим доступа к ресурсу: https://developer.apple.com/documentation/swift.

15. Unicode Common Locale Data Repository [Электронный ресурс] // Unicode – Режим доступа к ресурсу: http://cldr.unicode.org/.

16. String type in Swift programming language [Электронный ресурс] // Apple Inc. – Режим доступа к ресурсу: https://developer.apple.com/documentation/swift/string.

17. Chronology of Unicode Version 1.0 [Электронный ресурс] // Unicode – Режим доступа к ресурсу: http://www.unicode.org/history/versionone.html?fbclid=IwAR2n_hc7ji3OHne0eJJyzLE-zmeoKZQfZ2tB0Y2LGo4XTZgFlN3KykEULlI.

18. ICU Technical Committee [Электронный ресурс] // ICU - International Components for Unicode. – 1999. – Режим доступа к ресурсу: http://site.icu-project.org/projectinfo.

Информация об авторах

Коваль Юрий Владимирович *– аспирант, Украинский языково-информационный фонд НАН Украины, Киев, Украина. Тел. +380 (93) 202 2906. E-mail: yurii.smith@gmail.com.*

Сфера научных интересов: прикладная и компьютерная лингвистика, лексикография, системы обработки естественного языка (NLP), машинное обучение (ML), iOS и Swift разработка.

Надутенко Максим Викторович *– к.т.н., заведующий отделом информатики, Украинский языково-информационный фонд НАН Украины, Киев, Украина. E-mail: maxkrb@gmail.com.*

Сфера научных интересов: структурная, прикладная, математическая и компьютерная лингвистика, лексикография, когнитивные информационные технологии, лексикографические системы, системы профессионального взаимодействия в лингвистике, информационно-поисковые системы, онтологические системы и их применение в процессах поддержки принятия решений.

RESEARCH OF THE INTELLECTUAL SYSTEM OF KNOWLEDGE SEARCH IN DATABASES

Oleksii Vasilenko, Oleksandr Kuzomin, Bohdan Maliar

***Abstract**: The aim of the work is to research a modification for Rete pattern matching algorithm for medical expert rule-based systems. Developed modification uses the power of cloud system in order to shift the load from user's CPU and RAM.*

The work use Java programming language for implemented algorithms and support software. To provide high computing power used Amazon Elastic Compute Cloud – Amazon Linux AMI which included Java OpenJDK Runtime Environment.

***Keywords**: data mining, knowledge search, intellectual system, databases.*

***ITHEA Keywords**: E.2 Data Storage Representations, F.2 Analysis of algorithms and problem complexity.*

Introduction

The task of the differential medical diagnostic system is to determine the diseases that the patient may be ill, based on observations of his symptoms. Depending on the type of medical data there are two main approaches to medical diagnostics: diagnostics using the methods of probability theory and mathematical statistics based on objective statistical information; diagnostics using artificial intelligence based on subject information, i.e. knowledge and experience of a group of doctors.

One of the many areas of artificial intelligence is an expert system. The expert system has the following advantages: high efficiency; performance; reliability; accessibility to understand. To improve the efficiency of the system, knowledge base flexibility is required.

The system is focused on the possible expansion of the list of diseases, symptoms and interrelations between them by the criterion of the expediency of their use. A rule-based system is used to store and manipulate knowledge to interpret information in a useful way. A medical expert rule-based system is a software that is designed to help doctors around the world to choose the correct diagnosis based on a cluster of symptoms based on the list of knowledges provided by contact with a patient [Giarratano, 1994].

The common approach for rule-based system search is to analyze these knowledges and compare them with the knowledge base. The knowledge base consists of human-crafted rule sets.

Therefore, the goal of the work is to improve Rete algorithm for medical expert system by using the cloud as computing power.

Methods and models for problem resolution

Semantic Networks is the knowledge representation model most closely related to natural language. A semantic network is an oriented graph with vertices that correspond to symptoms, syndromes, etc. regarding to illness, concept or diagnostic situation, with other situations that can be identified by different methods that characterize the relationship between the objects of the study. The advantages of semantic networks include great expressive possibilities; visibility of the knowledge system, which is presented graphically; proximity to natural language, which is used when filling in the medical history (ontologies and precedents of “close” ones by the signs of the disease) compliance with the modern electronic representation of patients and their diseases; easy adjustment [Jackson, 1990].

The negative points of using the network model are the following facts:

- the model does not give a clear idea about the structure of the subject area, which corresponds to it, therefore, the formation and modification of such a model is very difficult;
- network models are passive structures, for the processing of which requires a special apparatus of formal withdrawal and planning;
- the complexity of the search and conclusion on semantic networks; the presence of multiple relationships between network elements;
- the presence of multiple relationships between network elements.

A few words according to the usage of frame structures for diagnostic modeling. In its organization, the frame model is a lot like a semantic network. A frame is a collection of nodes and relationships organized hierarchically, where the upper nodes are general concepts, and the lower nodes are more particular cases of these concepts. Frames reflect a typical pattern of action in a real situation. Each node is an attribute - a slot containing a certain concept and value of this concept, or how the value of a slot can be the name of another frame, or a slot can contain the name of a procedure with which a specific value can be calculated, or a range of values can be found in a slot. A set of frames can be networked, while the frame properties are inherited from top to bottom, the top frame contains more general information that is valid for the entire hierarchy of frames. The use of frames in the formalization of the medical subject area gives quite effective results.

Negative aspects when using frames of knowledge representation models include the difficulty of adapting the model when introducing new knowledge, the need to use sufficiently large amounts of memory that are necessary for storing model elements, as well as the lack of simple mechanisms for managing knowledge output.

The basic idea of building logical models of knowledge representation is that all information necessary for solving applied problems is considered as a set of facts and statements, which are represented as formulas in a certain logic. Knowledge is reflected by a combination of such formulas, and the acquisition of new knowledge is reduced to the implementation of inference procedures [Kuzomin and Vasylenko, 2010].

The positive qualities of logical models include the fact that the basis should be the classical apparatus of mathematical logic, whose methods are well studied and have a formal justification; the presence of sufficiently effective procedures for withdrawal; knowledge bases can only be stored using a variety of axioms, and all other knowledge can be obtained from them by the rules of inference. The main drawbacks of the logical method of knowledge representation are the considerable time spent on building a chain output; the inability to effectively describe the rules on exceptions; the need to describe a large number of rules and statements when modeling a real medical disease, which may be contradictory.

Ontologies are another way of describing the subject area, the basic concepts of this area, their properties and the connections between them. Practically all models of ontologies contain concepts (concepts, classes, entities, categories), properties of concepts (slots, attributes, roles), relationships between concepts (connections, dependencies, functions) and additional restrictions. Ontology, together with many individual instances, constitute the knowledge base, describes the facts based on the generally accepted meaning of the dictionary used. The advantages of ontologies are determined by the possibility of sharing by people or software agents for a common understanding of the structure of information; the ability to reuse knowledge in the subject area; the ability to separate knowledge in the subject area from operational data; ability to analyze knowledge in the subject area.

As an example of using the semantic network model for solving medical diagnostics problems, we can use the CASNET system (Causal-Associational NETwork). On the basis of logical models, such systems as MYCIN - a system for diagnosing bacterial infections or a system INTERNIST - a diagnostic system in the field of general therapy are built. Given the level of development of information computer technologies, ontologies have found their application in many subject areas, in particular in medicine, the SNOMED dictionary or the Unified Medical Language System can be noted, which are used to formalize common terminology and annotation in the field of medicine.

Formally, the task of medical diagnosis can be presented as a classification task, which consists in the fact that in order to match the set of input parameters a specific disease.

The basic approaches that are used to solve the problem of medical diagnosis can be grouped as follows:

1. Logical approach.
2. Statistical approach.
3. Bionic approach.

The logical approach to making decisions in medicine is quite common, as it is a direct reflection of the doctor's considerations. The reasoning of the physician during the diagnostic process must be assured, consistent and evidence-based. That is, to determine a violation, which is characterized by a complex of symptoms, it is necessary to use the laws of formal logic. You can make a diagnosis using inductive reasoning associated with the prediction of the results of observations based on past experiences. But the inductive method in the diagnosis of the state, since the logic of knowledge and considerations is deductive in nature. Therefore, the deductive method is usually used in determining the diagnosis, that is, the process of deductive inference from a certain set of suspected diseases of the required diagnosis.

A variation of the logical approach is the method of decision trees or classification trees. To use this approach, the rule base of the form is composed: "If> Then" in the form of a hierarchical structure, which is a tree. To determine the class of the disease, it is necessary to answer the questions in the nodes of this tree, starting from its root, and thus make the transition to the next question. The positive aspects of this approach are the clarity of the method and clarity. But in practice, a number of questions arise that limit the use of this approach in solving the problem of classification. One of the problems associated with the choice of the next node. To solve it, various algorithms are used, which often give a too detailed tree structure and can lead to errors. Despite the various features of this approach, the logical scheme of questions and answers of the diagnostic process in the form of decision trees has been successfully applied in practice.

The group of statistical methods includes the Bayesian approach, methods of discriminant analysis, and conclusion based on precedents. The use of the Bayes theorem in determining the class of the disease is a common approach due to its simplicity, clarity and simple mathematical calculations, but it has a clear disadvantage - a large database of archival data is needed for the likelihood of a diagnosis to meet reality. Also, a negative point is the fact that a rare disease or non-standard symptoms are difficult to correctly identify.

Discriminant analysis is characterized by the presence of many calculations, the emergence of various kinds of connections between symptoms, the elucidation of the effect of the relationship of signs on the result of diagnosis, usually pulls in difficulties in solving a medical problem.

The case-based deduction method (Case Based Reasoning) is an approach based on the use of previous cases and experience. To make a decision, the search for similar precedents that took place in the past is first carried out, and then a measure of proximity between the new and all found cases for which solutions are determined is calculated. This approach has a number of limitations: a system based on precedents, built on knowledge, which is "drawn out" of experts, which means that there is a measure of subjectivism; knowledge gained from people needs to be assessed, verified and assessed for their credibility; lack of solving such problems for which there are no precedents; arbitrariness in the choice of a measure of proximity; a sharp decline in performance with a large set of input parameters.

The bionic approach is a process of artificially reproducing those structures and processes characteristic of the human brain. The advantages inherent in this approach are quite large: the ability to adapt (learning and self-learning) the parallelism of information processing; robustness (resistance to individual failures). Neural networks (NN) are used in solving problems of medical diagnostics, problems of classification (clustering), approximation, forecasting. Sometimes the applicability of the neural network approach is limited due to some drawbacks: a large amount of archival data is needed to train the created neural network; The factor of subjectivity in creating the NN structure is very important - the number of layers, the number of neurons in each layer and the activation function are selected by an expert, which, in turn, introduces additional uncertainty; Also, when training an NN, there is a possibility that the selected structure will not give adequate results when using some input data; Sufficient sensitivity to the input data - a training pattern on which NN learns must be properly prepared to eliminate the additional weight of individual input parameters, leading to unreliable learning outcomes.

Thus, the task of medical diagnostics is a rather complicated task due to the fact that patient data is poorly structured and have a different character. Some of the necessary information relating to the patient is usually missing, which introduces additional difficulties in the processing of medical data; some of the information is qualitative, since it is determined by the doctor, that is, there is a share of subjectivity; some of the information reflects the results of the analyzes, which means that it is necessary to take into account the randomness factor due to measurement errors. To date, there are several approaches to work with uncertainty in the tasks of medical diagnostics. The probabilistic approach is an approach where unknown factors are statistically stable and therefore represent random variables or random events. In this case, all the necessary statistical characteristics must be determined: the laws of distribution and their parameters, functions, or probability density distributions, which, in turn, introduces additional difficulties.

Another approach to accounting for uncertainty is the use of the theory of fuzzy sets, which allows you to take into account the inaccessible or no data, or the data are of an exclusively qualitative nature. However, the application of this approach is associated with a number of difficulties, for example, with determining the type of membership function, as well as difficulties with the subsequent processing of fuzzy data.

Resolution of the problem

The medical expert system of differential diagnostics is a system for determining diagnostic hypotheses based on the medical knowledge of a group of doctors and the facts of the patient's symptoms found. Diagnostic hypotheses are possible diseases (expert judgment) that a patient suffers from [Kuzomin and Vasylenko, 2010]. Based on diagnostic hypotheses, it is possible to determine the possible specialty of a doctor, according to which the patient should be treated [Kuzomin and Vasylenko, 2014].

The formal model of the system can be represented as a tuple:

$$MESDD = \langle WM, KB, UI, IE, EM, KA \rangle, \tag{1}$$

where WM - the working memory; KB - medical knowledge base; UI - user interface; IE - control output diagnostic solutions; EM - explanation of performance information; KA - the acquisition of medical knowledge.

In the mode of acquiring knowledge, expert doctors fill the system with medical knowledge, which allow it to independently solve the problems of finding a diagnostic solution in the mode of medical consultation. In the mode of medical consultation, the user-patient is involved in communicating with the system, who is interested in the effective diagnostic information and explanatory information of the result.

The key concept of the system is the knowledge base. For the presentation of knowledge in the system, a combination of frame and fuzzy knowledge bases was chosen. The frame knowledge base is presented to describe the current state of the field of diagnostics, that is, a quantitative assessment of each disease based on knowledge from the knowledge base and evidence of symptoms. A fuzzy knowledge base is presented to describe the dynamic known in the transitions between the states of the diagnostic area, that is, a cause-effect relationship that relates a disease to symptoms in its symptom complex. Using procedural knowledge and the inheritance of frame properties, you can implement a mechanism for controlling the output of a diagnostic solution.

Frame knowledge base can be represented as a tuple:

$$KB = \langle FC, FSM, FSD, FSS, FSC, \{FIM_i\}, \{FID_j\}, \{FIS_k\}, \{FIC_h\} \rangle \tag{2}$$

where FC - a frame class; FSM - specialty frame prototype; FSD - a prototype of the disease; FSS - symptom frame prototype; FSC - prototype frame of the symptom complex; $\{FIM_i\}$ - a set of frames-instances of specialties; $\{FID_j\}$- a set of frames-instances of diseases; $\{FIS_k\}$ - a set of frame-instances of symptoms FSC_h - a set of frame-symptom complexes.

Formally, a fuzzy rule can be represented as a tuple:

$$FR = \left\langle NFR, \left\{ \left\langle FSMS_i, SF_i \right\rangle \right\} \rightarrow FSMD, CF \right\rangle , \tag{3}$$

where WFR is the name of a fuzzy rule; $FSMS_i$ is a fuzzy statement of a symptom variable; SF_i is the coefficient of symptom specificity in a symptom complex; $FSMD$ is a fuzzy statement of a variable disease; CF is the coefficient of confidence of the likelihood of the disease.

Formally, a fuzzy statement of a single variable can be represented as a tuple:

$$FSM = \langle LV, LT, M \rangle, \tag{4}$$

where LV is a linguistic variable; LT is the linguistic term of the variable M - modifier, which correspond to the words “very”, “more or less”, “no”, etc.

Formally, a linguistic variable can be represented as a tuple:

$$LV = \langle NLV, TSLV, ULV, GLV, MLV, TLV \rangle \tag{5}$$

where NLV is the name of the linguistic variable; $TSLV$ - term set of linguistic variable; ULV - the domain of definition of each $TSLV$ element; GLV - syntactic rules, often in the form of a formal grammar, generating the name of linguistic terms; MLV - semantic rules that define the membership functions of linguistic terms generated by the syntactic rules of GLV ; TV is a type of linguistic variable (symptom or disease).

Formally, the linguistic variable term can be represented as a tuple:

$$LT = \langle NLT, MF \rangle, \tag{6}$$

where NLT is the name of the linguistic term; MF - a variable membership function of a linguistic term. As the membership function, the following functions are used:

$$\mu_{LT}(u) = \frac{1}{1 + \left(\frac{u-b}{c} \right)^2}, \tag{7}$$

where b and c are the settings: b is the coordinate of the maximum of the function; c is the coefficient of concentration-stretching function.

In a system based on a combination of frame and fuzzy models, symptom slots are treated as weekends, and disease slots are targeted. When generating additional questions, procedures-methods are initiated that implement the opposite conclusion to determine the initial values of possible symptoms. When assigning the initial values of the slots, the demons procedures that are responsible for the direct inference and perform a fuzzy inference to obtain the target values of the disease slots will work.

To highlight the final diagnosis for the disease, 3 main criteria are used:

- the most minimal range of reliable solutions;
- the most maximum current integrated assessment of the disease;
- the maximum area of unreliable decision.

When implemented, the system checks the suitability of the fuzzy-production rules of each disease for each fact of symptoms in the working memory, if necessary, performs them and proceeds to the next disease, returning to the beginning when all diseases are exhausted. To ensure speed with a large knowledge base and a large number of facts in the working memory, the Rete algorithm is used. This algorithm sacrifices the amount of memory for speed, so the calculations must be carried out in cloud systems, it will simplify the load on the doctor's working machine.

When using the Rete algorithm, the medical knowledge base translates into a network of Rete (or a prefixal tree), in the end nodes of which there are, on the one hand, procedures-demons attached to output slots, and on the other, procedures-methods for obtaining values of target slots with the truth of the premise of fuzzy-production rules, information about which is stored in the intermediate nodes (α and β-memory). When symptoms enter the working memory, output slots are assigned a value, and only a small part of the network connected to it is updated.

At the time of assignment, not all rules are known under conditions of uncertainty. Therefore, it is impossible to build a single network for all the rules. Such a modification of the Rete algorithm is called the fast Rete algorithm.

The following components should be stored in the modified Rete network:

- an activation list in which parent slots are stored, i.e. slots of prototype frames;
- the context of the activity, which stores references to the current frames that caused the activation, that is, instance frames.

When changing the value of some source slot, it is in the premise that all the associated demons are activated, which directly try to calculate the value of the target slot in the conclusion. When calculating using the fuzzy inference algorithm, β-memory stores intermediate results, and the slots are used as α-memory. Unification of the rule with values in the working memory is not carried out as such, but is replaced by implicit inheritance unification, which is achieved by calling the daemon procedures of all parent frames with passing the current frame (triggering activation) as the call context. Thus, the network is implicitly formed by demons attached to the slots, rules associated with them, and a fuzzy conclusion, in the nodes of which intermediate results of calculations are stored.

Comparison of the symptom complexes with the available facts from the working memory is carried out after the approval of the incoming symptoms. As a result, at the stage a conflict set consists of potential diseases according to the following criteria:

- a potential disease corresponds to a symptom complex, in which the symptoms coincide with the symptoms entering the working memory;
- a potential disease corresponds to the age and sex of the patient, possibly suffering from this disease.

In case all the symptoms of a disease are present in the working memory, then such a disease will occur in a variety of diseases under consideration.

Conflict resolution is performed to select one or more of the most appropriate diseases from the conflict set. The result of this phase is the set of active diseases and determines the order of their implementation.

Conflict resolution is based on the principle “first come, first served”, i.e. priority over the choice of the first of the active diseases to trigger. In addition, the following criteria are used to select appropriate diseases:

- novelty. Active diseases correspond to the manifestations of symptoms that enter the working memory as the most. For this, it is necessary to provide the facts with a special attribute of time spawning;
- specificity. Active disease corresponds to a symptom complex with many facts of manifestations coming into the working memory of symptoms.

You can select a conflict resolution criterion or define a queue of several criteria. In addition, worn fuzzy-production rules should not be applied to existing facts.

To improve the effectiveness of teaching a fuzzy-production model of knowledge representation, it solves the problem of forming a knowledge base, the use of a genetic algorithm is proposed. To do this, the first is to put a method of encoding / decoding a fuzzy-production knowledge representation model, which defines some parameters (membership function parameters; coefficients of specificity and confidence), which are combined into a single vector. The value of one parameter lies in a certain neighborhood, which can be divided into 2-16 intervals. Then, to encode the slot number, you can use a 16-bit value in the gray code, in which the neighboring numbers have fewer positions.

To create the initial chromosome population, 100 chromosomes are randomly generated from the initial initialization of gene values in each neighborhood using the Gaussian method. Then, using the composition operation, combine a set of genes into a single chromosome to assess the fitness of chromosomes.

Each chromosome from the population is mapped to an assessment of its fitness of chromosomes in the population, the calculation of which is performed on the basis of training samples and vectors of model parameters. The learning process is considered complete if the condition that the resulting estimate is greater than the threshold value is satisfied. Selection of chromosomes. In a selection procedure based on the principle of a roulette wheel, the larger the sector on the roulette wheel (that is, the corresponding assessment of chromosome fitness), the higher the chance that this particular chromosome is chosen, which later on after performing the decomposition operation, genetic operators are used to create populations.

Application of genetic operators to chromosomes. In genetic algorithms, the crossover operator (90% probability) is responsible for the transfer of parents' genes to descendants. The mutation and inversion operators (probability 10%) are designed to maintain the diversity of chromosomes in a population.

The formation of a new population. Productive chromosomes must be placed in the population after performing the composition operation. To reduce the population to the original number of chromosomes, a reduction operator is used.After stopping the work of the genetic algorithm, a trained model comes out, approximating data from a training sample with a given accuracy and forms a knowledge base consisting of a system of fuzzy-production rules.

Creating a knowledge base.

At (Fig. 1)–Disease stores general information about acute renusitis (GDS).

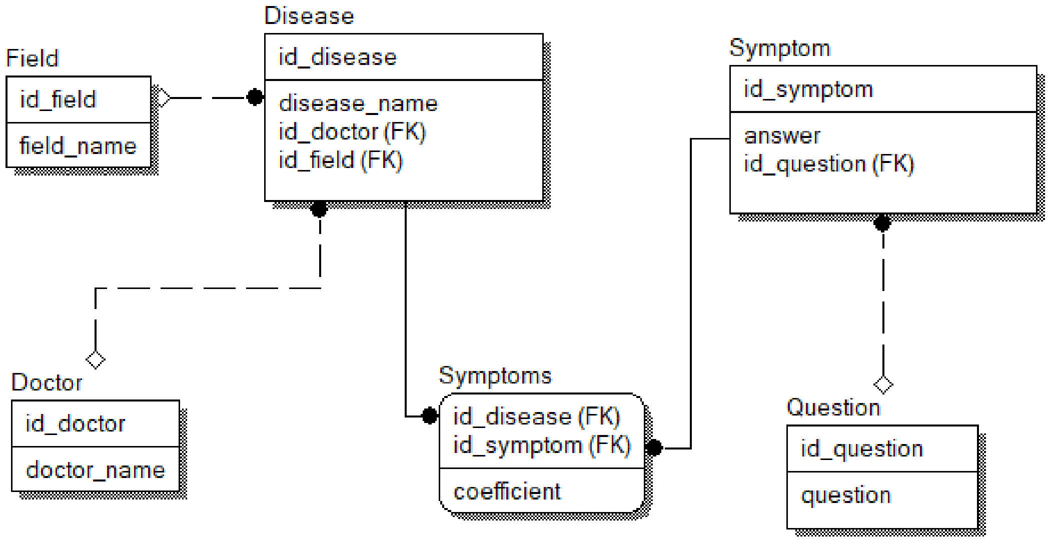


Figure 1–Knowledge Base MES

Field is a grouping of diseases by region in accordance with the International Classification of Diseases (ISS). Symptom (Fig. 1) stores the symptoms, the separate table contains the questions that the MES forms during the analysis of fuzzy knowledge. For example, the question “The nature of nasal breathing" and in the Symptom table are stored the diagnoses referring to this question with the answers "free", "moderately difficult", "difficult", which are stored in the answer field. Each option is treated as a separate diagnosis. If the question can be answered yes / no, then the answer is stored null. The Symptoms table connects the symptoms with the diagnosis, there is a coefficient field that takes into account the weights — an estimate for different diagnoses; if one is more important than the other, put more weight.

Conclusions

The aim of the work is to begin research for possible improvements of the Rete algorithm for medical expert system.

During the work was implemented modification for Rete algorithm. The key idea of the work is that it can be used as for daily routine operations in every medical institution. Unfortunately, we do not have enough time to present them in the work.

In the work, we finished only theoretical part of our goal. Right now, we implemented this modification with the help of Java language code and Amazon Cloud Services. The next step will be to implement this modification to newer variants of Rete algorithm and to find similar algorithms which can be modified in this way also.

Bibliography

[Jackson, 1990] Jackson Peter. Introduction to Expert Systems. Addison-Wesley Pub, Vol.2, 1990.

[Giarratano, 1994] Giarratano Joseph, Riley Gary. Expert Systems: Principles and Programming. PWS Publishing Co, Vol.2, 1994.

[Forgy, 1982] Forgy Charles. Rete: A Fast Algorithm for the Many Pattern/Many Object Pattern Match Problem. Artificial Intelligence, pp 17-37, 1982.Authors' Information

[Kuzomin and Vasylenko, 2014] Kuzomin, O.Ya., Vasylenko, O., Obespechenie bezopasnosti ispolzovaniia baz dannyh v usloviiah chrezvychainyh situazii. International Journal "Information Technologies Knowledge", Vol. 8, Num. 2. 2014. pp. 173-187.

[Kuzomin and Vasylenko, 2010] Kuzomin, O.Ya., Vasylenko, O., Analiz estestvenno iazykovyh obiektov I predstavlenie znanii. Vostochno-Evropeiskii zhurnal peredovyh technologii, Vol. 6/2(48). 2010.

Authors' Information

***Oleksii Vasylenko** – Aspirant of Kharkiv National University of Radioelectronics; Kharkiv, Ukraine;*

***e-mail**: ichbierste@gmail.com tel.: +380 63 841 66 23*

***Major Fields of Scientific Research**: General theoretical information research, Knowledge Discovery and Engineering, Business Informatics.*

***Prof. Dr.-hab. Oleksandr Kuzomin** – Informatics chair of Kharkiv National University of Radio Electronics; Kharkiv, Ukraine Ukraine;*

***e-mail**: kuzy@daad-alumni.de tel.: +38(057)7021515*

***Major Fields of Scientific Research**: General theoretical information research, Decision Making, Emergency Prevention, Data Mining, Business Informatics.*

***Bohdan Maliar** – Master student in Information and Communication technologies of Kharkiv National University of Radio Electronics; Kharkiv, Ukraine Ukraine;*

***e-mail**: bohdan.maliar@gmail.com*

***Major Fields of Scientific Research**: Big Data, Data Mining, Data Analyses.*

RESEARCH OF MEDICAL DIAGNOSTIC DATA SEARCH METHODS

Oleksii Vasilenko, Oleksandr Kuzomin, Oleksandr Shapoval

Abstract: *Improving the efficiency of diagnosis and treatment is the most important task of health care. One of the solutions to this problem is the complex automation of the collecting, storing and processing of medical information, for which medical information systems are being created. As well as advisory assistance to the doctor for the diagnosis of diseases, for which medical expert systems are created. The aim of the work is to research the methods of modeling and creating medical expert systems, the analysis of their disadvantages and the way of their solutions. Targeted research relies on the Bayesian trust networks.*

Key words: *medical expert systems, diagnosis, medical knowledge base, syndromes, sym*

Introduction

The general problem of identifying the diagnosis of a person's illness primarily and foremost lays on the presence of uncertainty, inaccuracy and insufficient volume of biomedical data and knowledge, difficulties in formalizing knowledge, excessive variety of making decisions regarding diagnosis, modeling a patient's condition, modeling the clinical thinking of a physician, choosing treatment methods, etc.

The object of the research is the processes of modeling the diagnosis of clinical medicine, the development of methods for the analysis and synthesis of biomedical data and the identification of knowledge from them and the creation of medical expert systems (MES) of clinical medicine.

The subject of the research is the models and methods of analysis of medical data and knowledge, the reliability of storing and processing of complex biomedical Big Data and the development of medical expert systems for the diagnosis of clinical medicine.

In medicine, it is important to find accurate methods for describing the data for research, as well as estimating and monitoring the process of diagnosis. The best way to the accuracy and logical considerations in solving any problem is to use a mathematical approach. This approach is might be chosen regardless of how difficult and complex the issue is. If we deal with a large number of interdependent factors, symptoms, syndromes, signs of illnesses that exhibit significant natural variability, then there is only the one effective way to describe the complex scheme of their effects – to the use of the appropriate statistical method. If the number of factors or the number of data categories is very large, then it is desirable or even necessary to use data processing methods such as Data Mining, in order to obtain the desired results in a short time, it is necessary to create and use medical expert systems (MES).

Research of the subject area and the choice of methods and models of medical diagnosis

It is common knowledge that the medical expert system is a set of programs that performs the functions of an expert in solving medical diagnosis problems. Perform an analysis of some well-known modern MES:

1. The system of medical diagnostic Diagnos.ru.
2. Diagnostic decisions of the expert system "ЕСБАД".
3. Expert system "МУТАНТ" (MUTANT), which was created by the staff of the Electronic Computing Center of Moscow University.
4. Automated system of early diagnosis of hereditary diseases "ДИАГЕН" (DIAGEN), which allows identifying more than 1200 forms.
5. Dendral - analysis of mass spectrometry data.
6. Mycin - diagnosis of infectious diseases of the blood and the antibiotics recommendations.
7. STD Wizard - an expert system for recommending and selecting medical analyzes (diagnostics).

As you can see in the analysis, it is possible to make three main conclusions:

1. Scope of application of each MES is objectively due to a narrow list of diseases, that is the representation of databases and knowledge of diseases.
2. On average, the MES have from 56% to 90% of the correct diagnoses.
3. The main principles for the development of the MES are production rules, which give a very large percentage of correct diagnoses.

These conclusions provide a basis for use in the development of MES to consider the possibility of using production rules in the creation of the MKB (medical knowledge base) for the development of MES. In addition, the data (Tab. 1) determine the most effective directions:

1. Using electronic medical data (EMD).
2. Building a knowledge base (KB) on the rules of products of the type "IF - THEN".

The analysis of the using of modern researches in the MKB development that were presented above provides a basis for the selection, modification or combination of methods for analyzing medical data for the diagnosis of diseases. When developing models and methods of diagnosis, one should take into account rather high results in the development of modification of statistical analysis methods and the application of Data Mining methods, which are often been supplemented by the use of fuzzy, hybrid neural networks in the modeling of complex applications. In addition, special attention must be paid to the very efficient use of belief networks that use the theory of subjective probabilities, are based on the Bayesian method, the Wald method, or, as it is still called, the method of sequential statistical analysis, the diagnostic tables of Sano, the method of linear discriminant functions, etc.

The use of subjective expectations in Bayesian networks is the only alternative in practice, if it is necessary to take into account the views of experts (e.g. physician) about the possibility of an event occurring to which the notion of repeatability is used and it is impossible to describe it in terms of a set of elementary events.

With these methods, one can calculate the distribution of probabilities in expert systems, which is a more convenient method for calculation, rather than assume with the help of statistical functions. With Bayesian theory, one can calculate the probability of judgments that are not certain. Such probability is determined by the level of confidence in the truth of a judgment. However, no matter how good this method seems, there are some negative aspects in using this theory. For example, in many cases it is psychologically difficult for an expert to remain within the framework of a strict mathematical apparatus of probability theory, which in its nature is objective. It is necessary to break the strict conditions of equality of units of probability sum of all possible states, especially in their large numbers. In many cases, actually observing evidence is confirmed not by any particular result (or hypothesis), but immediately by a certain set that does not allow determining the probability of each of them. If the expert estimates values that have a rather obscure meaning, whose properties in many cases do not coincide with the usual representations, it can being confronted with the fact that its answers will not provide useful informing about the estimated values.

Expert systems that use the theory of subjective probabilities are in great demand among expert systems for finding solutions. These expert systems allow you to get a right answer to various questions in a narrow subject area.

The most effective areas of data analysis and knowledge discovery in the proposed study, that was determined in (Tab. 1), are:

1. Method of using "micro situations" and theory of utility.
2. Fuzzy and hybrid neural networks.
3. The use of ontology, case studies and disease knowledge.

To develop models and methods of diagnosis, we need to make some clarifications regarding what clinical thinking is. Having determined it, we can more accurately develop a diagnostic algorithm for diseases. In addition, the basic logical connections of concepts reflecting the subject area and necessary to refine the development of models for clinical diagnosis should be used.

Clinical thinking in accordance with is the process of thinking physician from the moment of meeting with the patient or receiving the first preliminary information until the recovery or death of the patient. The result of the clinical thinking of a doctor is the formulation of clinical diagnosis, treatment plan and its practical implementation. Therefore, when we have plenty of previous examples of diagnoses (in ontological or precedent presented in the database and knowledge base) as the results of practical and positive examples of clinical thinking of doctors, we will be able to develop algorithms for automatic diagnosis of the disease.

The general version of the stages of creating a criminal diagnosis based on clinical thinking is depicted on.

With the development of medical science, a scheme for formatting a clinical diagnosis was developed:

1. The main disease.
2. Complications of the main disease.
3. Concomitant diseases.

In addition, it is advisable to take into account the following stages of the diagnosis (Fig. 1):

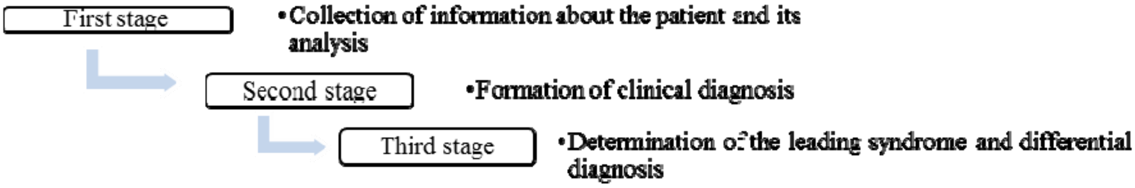

First stage
•Collection of information about the patient and its analysis
Second stage
•Formation of clinical diagnosis
Third stage
•Determination of the leading syndrome and differential diagnosis

Figure 1 – Diagnosis search algorithm

Table 1 – An example of the forming of clinical analysis

The main disease	Measles, typical, medium-grained form, period of rashes, abrupt course of the disease.
Complications of the main disease	Laryngitis
Concomitant diseases	Atypical dermatitis, localized form, remission period

As a variant of the stage of forming a clinical diagnosis (Fig. 1), it is expedient to bring the following:

1. Nosological form.
2. Degree of gravity of the process.
3. Stage, period of illness.
4. Degree of compensation.
5. Phase of the process (active, inactive).
6. The nature and localization of the pathological process.
7. Etiology of the disease.

However, these clarifications do not affect the concrete formation of the clinical diagnosis only complement and might be take into account in developing the methods for diagnosing diseases. The basis of the rules of fuzzy ES work is the concept of linguistic variables. Each of them has a set of values - fuzzy variables that form its term set. The linguistic variable L is characterize by the following set of properties:

$$\mathrm{L} = (\mathrm{X}, \mathrm{T}(\mathrm{X}), \mathrm{U}, \mathrm{G}, \mathrm{M}), \tag{1}$$

where X is the name of the variable; $\mathrm{T}(\mathrm{X})$ - term-set of a variable X, that is, the set of names of the linguistic values of a variable X, each of these values being a fuzzy variable x' with values from a universal set U_3 with a base variable u; G is a syntactic rule that generates the names x' of the values of the variable X; M - a semantic rule that matches each fuzzy variable x' with its meaning $\mathrm{M}(\mathrm{x}')$, that is, the fuzzy subset $\mathrm{M}(\mathrm{x}')$ of the universal set U.

Fuzzy variable is characterized by $\langle \mathrm{x},\mathrm{U},\mathrm{X} \rangle$, where x - the name of the variable; U - universal set; X - fuzzy subset of the set U, which is a fuzzy constraint on the value of a variable $\mathrm{u} \in \mathrm{U}$, conditioned X.

It is proposed to analyze the data and knowledge that might be used for diagnosis. First, this study relies on a broad view of electronic medical data (EMD) about patients and illnesses.

It is also important to consider that for the problem under consideration in this study it is necessary to process a large amount of data. That is, Medical Big Date (MBD). However, for reliability of work of MES and reception of high-quality MKB it is necessary to analyze and provide high reliability of storage MKB.

In addition, it is necessary to develop the architecture and structure of the MES. This research direction relies on the development of UML models and diagrams that provide the development of an adaptive or self-organized MES. The software (software) of the MES must be developed on the basis of the use of modern MBD processing technologies such as OOP (Object Oriented Programming), Map Reduce, Hadoop, etc.

For the study of illness probability calculation, the Bayesian theorem is used - a theorem, which, based on circumstances, describes the probability of an event. In our example, we use its Bayesian interpretation, that is, the probability measures the measure of confidence. Bayesian theorem therefore links the degree of confidence in the statement before and after considering the testimony. There is described the formula of probability as follows:

$$P(C \mid E) = \frac{P(E \mid C) * P(E)}{P(C)}, \tag{2}$$

where $P(C \mid E)$ – probability that effect is caused;

$P(E \mid C)$ – probability that the effect will appear when the cause appears; $P(C)$ – probability of the cause; $P(E)$ – probability of the effect.

If the number of effects is greater than one, then the coincidence of the probability class are summed up:

$$P(C \mid E_1, E_2, ...) = \frac{P(E_1 \mid C) * P(E_1) + P(E_2 \mid C) * P(E_2) + ...}{P(C)}, \tag{3}$$

where $P(C \mid E_1, E_2, ...)$ – probability that the cause causes this effect,

$P(E_{1,2,...} \mid C)$ – probability that the effect will appear when the cause appears,

$P(C)$ – probability of the cause,

$P(E_{1,2,...})$ – probability of the effect.

Thereof considering all causes as equiprobable, in order to find $P(E)$ and $P(C)$ the following formula is used:

$$P(E), P(C) = \frac{1}{n}, \tag{4}$$

where n is the number of elements of this type (for example, symptoms, or syndromes, etc.).

This way, you can always complete statistics when adding new data to the system by simply adding new information to the already calculated vertices of the nodes. Using Formula (3) as:

$$P(C \mid E_1, E_2, ..., E_n) = P(C \mid E_1, E_2, ...) + \frac{P(E_n \mid C) * P(E_n)}{P(C)}, \tag{5}$$

where $P(C \mid E_1, E_2, ...)$ – preliminary probability of the node,

$\frac{P(E_n \mid C) * P(E_n)}{P(C)}$ – adding new information to the system,

$P(E_n \mid C)$ – probability that a new effect will appear when the cause appears,

$P(E_n)$ – probability of the effect.

Example of probability distribution

For example, we have input data (Table 2)

Table 2 – The values of a priori and conditional probabilities for hypotheses

$p(\ )/i$	1	2	3
$p(H_i)$	0,59	0,39	0,02
$p(E_1 \mid H_i)$	0,49	0,89	0,39
$p(E_2 \mid H_i)$	0,09	0,79	0,99

In this case, the initial hypothesis characterizes the event associated with determining the reliability of some disease:

- H_1 – "pneumonia";
- H_2 – "bronchitis";
- H_3 – "tuberculosis".

Events (conditionally independent evidence) that support the initial hypothesis are:

- E_1 – "high temperature";
- E_2 – "coughing".

In the process of gathering facts of probability, hypotheses will increase if the facts approve them or decrease if they disprove them. Suppose we have only one fact E_1 – "high temperature" (that is with probability equal to one has come a fact E_1 – "high temperature"). Having received E_1 – "high temperature" we compute the a posteriori probability for hypotheses according to the Bayesian formula for one fact:

$$p(H_i \mid E_1) = \frac{p(E_1 \mid H_i) * p(H_i)}{\sum_{k=1}^{3} p(E_1 \mid H_k) * p(H_k)}, \quad i = 1,2,3. \quad . \tag{6}$$

As follows:

- $p(H_1 \mid E_1) = \frac{0,49 * 0,59}{0,49 * 0,59 + 0,89 * 0,39 + 0,39 * 0,02} = 0,4489;$
- $p(H_2 \mid E_1) = \frac{0,89 * 0,39}{0,49 * 0,59 + 0,89 * 0,39 + 0,39 * 0,02} = 0,5390;$
- $p(H_3 \mid E_1) = \frac{0,39 * 0,02}{0,49 * 0,59 + 0,89 * 0,39 + 0,39 * 0,02} = 0,0121. \quad .$

We do a check, the sum of a posteriori probabilities as a result should give to one:

$$p(H_1 \mid E_1) + p(H_2 \mid E_1) + p(H_3 \mid E_1) = 1. \tag{7}$$

That is, $0,4489 + 0,5390 + 0,0121 = 1$.

After E_1 – "high temperature" the confidence in the hypothesis H_1 – "pneumonia" and H_3 – "tuberculosis" reduced, when it increased to H_2 – "bronchitis". In cases where there are facts confirming both event E_1 – "high temperature" and event E_2 – "coughing", then the a posteriori probability of the initial hypothesis can also be calculated by the Bayesian rule:

$$p(H_i \mid E_1E_2) = \frac{p(E_1E_2 \mid H_i) * p(H_i)}{\sum_{k=1}^{3} p(E_1E_2 \mid H_k) * p(H_k)}, \quad i = 1,2,3. \tag{8}$$

Because of conditional independence of high temperature and coughing, we can rewrite Bayesian formula as:

$$p(H_i \mid E_1E_2) = \frac{p(E_1 \mid H_i) * p(E_2 \mid H_i) * p(H_i)}{\sum_{k=1}^{3} p(E_1 \mid H_k) * p(E_2 \mid H_i) * p(H_k)}, \quad i = 1,2,3. \tag{9}$$

As follows:

– $p(H_1 \mid E_1E_2) = \frac{0{,}49 * 0{,}09 * 0{,}59}{0{,}49 * 0{,}09 * 0{,}59 + 0{,}89 * 0{,}79 * 0{,}39 + 0{,}39 * 0{,}99 * 0{,}02} = 0{,}0845;$

– $p(H_2 \mid E_1E_2) = \frac{0{,}89 * 0{,}79 * 0{,}39}{0{,}49 * 0{,}09 * 0{,}59 + 0{,}89 * 0{,}79 * 0{,}39 + 0{,}39 * 0{,}99 * 0{,}02} = 0{,}8904;$

– $p(H_3 \mid E_1E_2) = \frac{0{,}39 * 0{,}99 * 0{,}02}{0{,}49 * 0{,}09 * 0{,}59 + 0{,}89 * 0{,}79 * 0{,}39 + 0{,}39 * 0{,}99 * 0{,}02} = 0{,}0251.$

Equivalence probability check:

$$p(H_1 \mid E_1E_2) + p(H_2 \mid E_1E_2) + p(H_3 \mid E_1E_2) = 1. \tag{10}$$

That is, $0{,}0845 + 0{,}8904 + 0{,}0251 = 1$.

Initial ranking was H_1 – "pneumonia", H_2 – "bronchitis" and H_3 – "tuberculosis", and all three remained after receiving the facts E_1 – "high temperature" and E_2 – "coughing". Herewith bronchitis more likely than pneumonia and tuberculosis. This indicates that having coughing and high temperature the probability of the disease bronchitis much bigger than the probability of the disease pneumonia or tuberculosis.

However, realistically, the spread of probabilities occurs in stages with the summation of individual facts and their impact on the probabilistic probability of the receiving the individual values E_i. It proceeds by using the a priori and a posteriori probability, thus:

1. Define $p(H_i)$ – a priori probability of events H_i.
2. For the received facts E_i set down $p(E_i \mid H_i)$.
3. Considering the Bayesian theorem calculate $p(H_i \mid E_i)$ depending on the outcome E_i, that is, we calculate the a posteriori probability of the event H_i.

4. Now you can mark the current a posteriori event probability H_i, as a new a priori probability Hi. Therefore, $p(H_i)$ equals $p(H_i \mid E_i)$ depending on the value E_i

5. Then choose a new fact for consideration and proceed to step two.

Consider an example, the fact E_2 – "coughing" entered the system. Then:

- $p(H_1 \mid E_2) = \dfrac{0{,}09 * 0{,}59}{0{,}09 * 0{,}59 + 0{,}79 * 0{,}39 + 0{,}99 * 0{,}02} = 0{,}1394;$
- $p(H_2 \mid E_2) = \dfrac{0{,}79 * 0{,}39}{0{,}09 * 0{,}59 + 0{,}79 * 0{,}39 + 0{,}99 * 0{,}02} = 0{,}8087;;$
- $p(H_3 \mid E_2) = \dfrac{0{,}99 * 0{,}02}{0{,}09 * 0{,}59 + 0{,}79 * 0{,}39 + 0{,}99 * 0{,}02} = 0{,}0519.$

Check:

$$p(H_1 \mid E_2) + p(H_2 \mid E_2) + p(H_3 \mid E_2) = 1. \tag{11}$$

That is, $0{,}1394 + 0{,}8087 + 0{,}0519 = 1$.

We take the resulting probability as a new a posteriori probability of hypothesis H_1, H_2 and H_3, so:

- $p\left(\tilde{H}_1\right) = 0{,}1394;$
- $p\left(\tilde{H}_2\right) = 0{,}8087;$
- $p\left(\tilde{H}_3\right) = 0{,}0519.$

Now, if any additional fact like E_1 – "high temperature" arrives, then the new a posteriori probability of the hypothesis calculates only on the evidence that arrives again:

- $p(H_1 \mid E_1 E_2) = p\left(\tilde{H}_1 \mid E_1\right) = \dfrac{0{,}49 * 0{,}1394}{049 * 0{,}1394 + 0{,}89 * 0{,}8087 + 0{,}39 * 0{,}0519} = 0{,}0845;$
- $p(H_2 \mid E_1 E_2) = p\left(\tilde{H}_2 \mid E_1\right) = \dfrac{0{,}89 * 0{,}8087}{049 * 0{,}1394 + 0{,}89 * 0{,}8087 + 0{,}39 * 0{,}0519} = 0{,}8905;$
- $p(H_3 \mid E_1 E_2) = p\left(\tilde{H}_3 \mid E_1\right) = \dfrac{0{,}39 * 0{,}0519}{049 * 0{,}1394 + 0{,}89 * 0{,}8087 + 0{,}39 * 0{,}0519} = 0{,}0250.\ .$

Check:

$$p\left(\tilde{H}1 \mid E_1\right) + p\left(\tilde{H}2 \mid E_1\right) + p\left(\tilde{H}3 \mid E_1\right) = 1. \tag{12}$$

That is, $0{,}0845 + 0{,}8905 + 0{,}0250 = 1$.

From the example given, you can see that the iterative procedure for the sequential probability distribution during the receiving of new facts allows you to get the results similar to the results of the Bayesian rule for two simultaneously received facts. The value of the hypothesis H_2 – "bronchitis" is most likely then H_1 – "pneumonia" and H_3 – "tuberculosis".

Conclusion

There are ES, built on objective and subjective views on the concept of the probability of an event. Knowledge bases of the ES accumulate human knowledge. Therefore, interpretations based on subjective trust are best suited to represent experts' knowledge in the light of probabilities. As a result, most of the modern EUs who use the theory of probabilities are "Bayesian".

The use of the Bayesian strategy in the ES implements with using the Bayesian formula of "inverse probabilities", that is, the estimation of conditional probabilities of hypotheses. In the presence of several signs (symptoms), such calculations are simply realized in the assumption of statistical independence of the characteristics, which is far from always corresponds to reality. However, practice shows that such an approach, despite its obvious mathematical incorrectness, is quite applicable, since it usually leads to correct conclusions.

Expert systems that use the theory of subjective probabilities are widely used in medicine as well as in other areas where it is necessary to determine the probability of occurrence of a certain event clearly and meaningfully. The theory of subjective probabilities subordinate directly to the Bayesian theory. It is used to evaluate a specific task, analyzing it, giving a solid answer, and making predictions for the future.

During the calculations of the distribution of probabilities in expert systems with given hypotheses, H_1 - "pneumonia", H_2 - "bronchitis", H_3 - "tuberculosis", characterizing the event associated with the definition of some disease, the result was obtained, indicating the probability the appearance of bronchitis, more than two other specified diseases.

Bibliography

[*Oleksandr Kuzomin. 2017*] DEVELOPMENT OF INTELECTUAL MODELS AND METHODS OF EXPERT SYSTEMS OF CLINICAL MEDICINE.// Oleksya Vasilenko, Tatyana Tolmachova International Journal “Informattion Technologes&Knolegedge”. Volume 11, Namber 2, 2017. PP. 186-199 ISSN 1313-0455 (printed), ISSN 1313-048X (online), [*Oleksandr Kuzomin. 2014*] Data loss minimization in situation's centrums databases // Oleksandr Ya. Kuzomin, Oleksii Vasylenko. Chairman IDRC Davos 2014 - Global Risk Forum GRF Davos - Davos – Switzerland. PP 153-154.

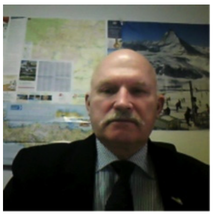
[*Oleksandr Kuzomin. 2017*] METHODS AND MODELS FOR BUILDING A DISTRIBUTED MOBILE EMERGENCY MONITORING SYSTEM.Oleksandr Kuzomin, Oleksii. Vasylenko, 17th International Multidisciplinary Scientific Geoconference SGEM 2017. Conference Proceedings, Informatics Geoinformatics. Volume 17. ISSUE 21 PP 433 – 440. ISBN 978-619-7408-01-0, ISSN 1314-2704. DOI: 10.5593/sgem2017/2.1,

Authors' Information

Oleksii Vasylenko *– Aspirant of Kharkiv National University of Radioelectronics; Kharkiv, Ukraine;*

e-mail: *ichbierste@gmail.com tel.: +380 63 841 66 23*

Major Fields of Scientific Research*: General theoretical information research, Knowledge Discovery and Engineering, Business Informatics.*



Prof. Dr. Oleksandr Kuzomin *– Informatics chair Innovation-marketing department of Kharkiv National University of Radioelectronics; Kharkiv, Ukraine;*

e-mail: *kuzy@daad-alumni.de tel.: +38(057)7021515*

Major Fields of Scientific Research*: General theoretical information research, Decision Making, Emergency Prevention, Data Mining, Business Informatics.*

Oleksandr Shapoval *– Master student in Information and Communication technologies of Kharkiv National University of Radio Electronics; Kharkiv, Ukraine*

Major Fields of Scientific Research*: General theoretical information research, Knowledge Discovery and Engineering, Business Informatics.*

INTELLECTUAL MODELS AND MEANS OF THE BIOMETRIC SYSTEM DYNAMICS OF RINOSINUSITE

Oleksii Vasilenko, Oleksandr Kuzomin, Tatyana Khripushina

Abstract: *The object of the research is the processes of modeling the diagnosis of clinical medicine, the development of methods for the analysis and synthesis of biomedical data and the identification of knowledge from them and the creation of medical expert systems (MES) of clinical medicine.*

The purpose of this work is to research and compare existing MES, to formulate requirements and to develop their own MES, and to present the results of a study on the presentation of medical data in the MES. The developed algorithm for implementing the medical expert system was used to create a software tool for medical diagnosis.

Keywords: *Medical Data, Expert System, diagnosis, Graph.*

Introduction

Now in Ukraine there is a very big problem of keeping a person's life. The problem of acute inflammatory diseases of the upper respiratory tract, acute rhinosinusitis (GMS) in particular, is one of the most urgent in modern clinical medicine. In recent years, there has been an increase in the frequency of nasal and sinus diseases, which manifests itself as an increase in both absolute (morbidity and prevalence) and relative (share in the structure of otorhinolaryngology) indicators. In Ukraine, the prevalence of acute rhinitis, rhinosinusitis and rhinopharyngitis has reached 489.9 cases per 10,000 population, and the incidence−5-15 cases per 1000 population depending on the season. Such patients make up 60-65% of ambulatory patients of otorhinolaryngologists.

To address the above issues, appropriate studies are needed to improve the quality of treatment. In particular, it is necessary to develop the direction of the study of diagnosing GDS high-quality, without errors and high level of clinical thinking of the doctor. Diagnosis of GDS is based on clinical data. The physician's conclusion is based on patient complaints, anamnesis, symptoms and signs of the disease, the data of the medical examination and includes a subjective assessment of the severity of the disease by the patient. So it is very important that in the early stages of the disease have the most accurate diagnosis. This is possible with the presence of a doctor's intellectual system for conducting a primary examination of a patient and anamnesis. In general, the physician makes decisions regarding the diagnosis of GDS in conditions of multiple uncertainty of the signs of illness, symptoms and multicritality of the requirements for the diagnostic process, which confirms the relevance of my research.

In addition, according to the order of the Ministry of Health of Ukraine of February 11, 2016, № 85 was introduced the "Unified clinical protocol of primary, secondary (specialized) and tertiary (highly specialized) medical care acute rhinosinusitis."

Development of models and diagnostic tools

According to the chosen object of research in the work models and methods of development of medical clinical diagnosis (MIC) of GDS are investigated.

Among the modern methods of research in medicine is the direction that uses the intellectual methods of system analysis of the development of MCD. In addition, the publications have long been the information that is sufficiently well applied by medical expert systems (MES).

This study uses statistical diagnostic materials for patients that were published [ ] and were collected directly in hospitals (Fig.1.1). It should be emphasized that electronic medical records (EHR) for patients are often used in the MES. This allows clinicians to increase access to large volumes of medical data collected during diagnosis of a patient [2.5].

In addition, the classification codes of ICD 10 [] were used. In particular, we have the following definitions for the following diseases: acute sinusitis–J01; acute sinusitis–J01.0; acute frontitis–J01.1; acute etiomyiditis–J01.2; acute stenoiditis–J01.3; acute pansionitis–J01.4; another form of acute sinusitis–J01.8; acute sinusitis unspecified–J01.9

Acute rhinosinusitis–is a sharp inflammation of the mucous membrane of the nasal cavity and paranasal sinuses (Fig.1). The code for MIC-10 acute rhinosinusitis has–J01, but when identifying the pathogen, an additional code should be indicated (V95-V97).

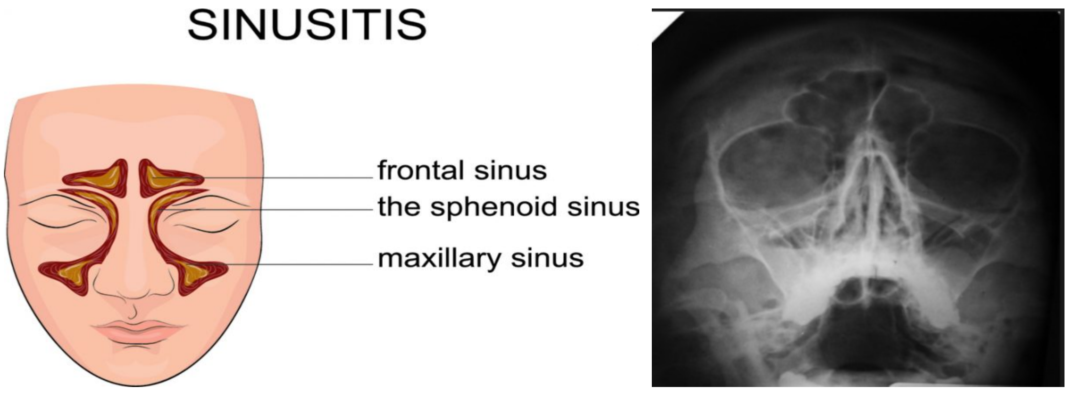


Figure 1 -The layout of cavities relative to the sinusoid and CT scaling

According to the epidemiology we have the following [Lukovkina A., 2013]:

- on average, the episode of sinusitis develops annually in 1 person out of every 7;
- Among women, morbidity is higher than among men;
- The peak in the incidence is from the age group of 45 to 74 years.

According to the etiology, we have the following [Lukovkina A., 2013]:

- respiratory viruses (influenza viruses, parainfluenza, rhinoviruses, adenoviruses, respiratory syncytial virus, enteroviruses, coronaviruses);
- bacteria (pneumococcus, hemophilic stem, moraxelle);
- fungi (in patients with immunodeficiency, for example, HIV, or with poorly controlled diabetes).

Regarding the classification of the disease, depending on the pathogen (V95-V97) we have the following varieties:

- acute viral rhinosinusitis;
- acute bacterial rhinosinusitis.

Depending on the location, the following varieties are distinguished:

- sinusitis (defeat of the maxillary sinuses);
- etiomyiditis (damage to the sinuses);
- frontitis (lesions of the frontal sinuses);
- Stenoiditis (damage to the sphenoid sinuses)

Depending on the course, the following varieties are distinguished:

- acute (<4 weeks);
- subacute (4-12 weeks);
- chronic (> 12 weeks).

Risk factors for acute rhinosinusitis [Lukovkina A., 2013], [Kuzomin, 2017]:

- acute viral infection of the upper respiratory tract. The most frequent OVIVDP encounters OVRs. Thus, according to the results of computed tomography, the objective signs of sinusitis are determined in 87% of cases of acute respiratory viral infections on the second to third day and in 39% of cases on the seventh day, 2,3 of the disease. OBRS is found much less frequently (0.5-2% of all cases OVIVDP1);
- allergic rhinitis;
- Any anatomical features or pathological changes that interfere with the drainage of cavity of the sinuses: nasal polyps, distortion of the nasal septum, etc.;
- violation of mucociliary clearness (local protection of the mucous membrane of the respiratory organs from external influences) as a result of smoking, cystic fibrosis;
- odontogenic infections;
- swimming;
- immunodeficiency (HIV infection, chemotherapy, prolonged use of corticosteroids, etc.).

Clinical manifestations and symptoms for acute rhinosinusitis (viral and bacterial) have the following:

- nasal congestion;
- mucosal purulent discharge from the nose (purulent character of excrement itself is not evidence of secondary bacterial infection);
- head and facial pains, which increase when inclining forward;
- slight increase in body temperature;
- hyposmia or anosmia;
- bad breath.

Diagnosis of acute bacterial rhinosinusitis is based on the presence of one or more three criteria [Kuzomin, 2015]:

- Sustained symptoms of sinusitis lasting 10 or more days without clinical improvement;
- severe OVIVDP symptom (fever> 39 ° C and purulent nasal discharge or pain in the face) for at least three days from the onset of the disease;
- worsening of symptoms (headaches, nasal discharge, fever) after an AED that lasted 5-6 days and was initially accompanied by an improvement in the condition.

Diagnosis of the disease takes place in the following sequence: interviewing a patient during an objective examination by a doctor and fixing a complaint and anamnesis.

During an objective survey it turns out:

- pain in palpation of the characteristic points of the projection of the paranasal sinuses;
- rhinoscopy;
- purulent discharge in the region of the middle nasal passage;
- diffuse swelling of the mucous membrane, narrowing of the middle bowel movement;
- hypertrophy of the nasal concha.

X-ray examination and computed tomography of subcutaneous sinuses can not differentiate OVRs from the OBP. These studies are shown only when suspected of intracranial or orbital complications [Lukovkina A., 2013].

In the following cases, the study of choice is the CT with contrast: reduction of visual acuity; diplopia; periorbital edema; severe headache or altered mental status. CT is more informative than X-ray examination, and can be informative in recurrent sinusitis or with preservation of symptoms, against the background of treatment (in this case, it is sufficient CT without contrast) [Lukovkina A., 2013].

In one study, the thickening of the mucous membrane of the paranasal sinuses in CT was found in 42% of healthy people with no symptoms. Such changes can not be the basis for the diagnosis of sinusitis without a corresponding clinical picture.Sowing of aspirates from sinus to culture is indicated only in those cases where there is no positive dynamics in response to antibiotic therapy or complications [Lukovkina A., 2013]. Sedation or nasal discharge is not recommended, as the results are unreliable.

Data mining methods used in medicine can be divided into several groups in accordance with the tasks solved by them (Tab. 1): predicting the course of the disease, the effect of a drug or group of drugs, the mortality rate; examination - diagnosis based on a set of symptoms; classification - clarification of the diagnosis; the search for associations is the search for hidden dependencies between various patient health indicators [36]. Consider further the basic methods of data mining used for processing medical information.

For the design of the expert system [Kuzomin, 2017] a fairly narrow branch of medicine was chosen - otorhinolaryngology, in particular, the problem of non-sense. The program allows to differentiate diseases such as:

- acute rhinitis (simple undead);
- acute sinusitis (inflammation of the sinuses of the nose);
- allergic rhinitis (runny nose and indigestion associated with allergies);
- acute respiratory viral infections (colds).

Table 1– The tasks of the intellectual analysis of data in medicine and the methods used to solve them

Purpose of the analysis	**Methods with a teacher**	**Methods without a teacher**
Forecasting	The smallest method squares Logistic regression Neural networks Adoption trees making SVM Splines	–
Survey	Decision tree	Main method components Clustering Link Analysis
Classification	Decision tree Neural networks Discriminant analysis Busing Naive Bayes Classifier	Clustering Self-organizing cards Kohonen
Association Search	–	Factor analysis Apriori algorithm

In this EU , the rules of diagnostics are implemented, which, depending on a given situation, will ask the user the necessary questions and receive a response in a strictly prescribed form (it will be necessary to choose the number of the corresponding answer). Further diagnostics will be conducted taking into account the previous answers to the questions given to the user. That is, the intellectual dialogue with the MES is supported on the basis of the analysis of the text content of the symptoms. As a result of the work of an expert system with a fairly high probability, it is possible to conduct a differential diagnosis between diseases associated with non-life. The MES analysis [https://www.med2000.ru/library2/diagnostika3.htm] gives two main conclusions:

1. On average, the MES have from 56% to 90% of the correct diagnoses.

2. The main principles for the development of the MES are production rules and the use of network neurons,Bayesian networks trust networks that give a very large percentage of correct diagnoses.

Development of intelligent diagnostic system

The theory of fuzzy sets was used to diagnose the GDS. A fuzzy expert system (FES) was developed based on the rules. He used the selected factors in the first stage and modeled the behavior of a specialist for diagnosis of GDS. Fuzzy expert systems are basically systems based on knowledge using fuzzy logic, fuzzy IF-THEN rules and membership functions (MF) (37). On (Fig.2) is shown the general architecture of a fuzzy expert system that displays the flow of data throughout the system (38). Its main structure includes four main components:

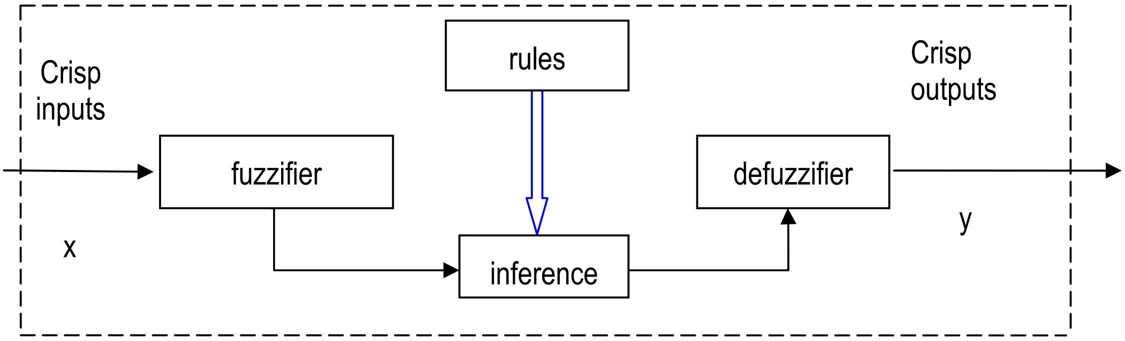


Figure 2 – Known common architecture of a fuzzy expert system [Kuzomin, 2017].

- Fuzzifier, which interprets clear input data (classical numbers) into fuzzy values;
- An inference mechanism that uses the fuzzy reasoning function to produce a fuzzy inference (in the case of Mamdani's inference);
- Knowledge Base, which includes a set of fuzzy rules and a set of membership functions that display fuzzy sets of linguistic variables;
- Defuser, which interprets fuzzy output into clear values [Kuzomin, 2017].

The first stage in the development of a fuzzy expert system is the definition of input and output variables. Differential diagnosis was conducted with nasopharyngitis, adenoiditis, allergic rhinitis, migraine, the presence of foreign bodies in the nasal cavity, pathology of the teeth. In complex cases, differential diagnosis with rare conditions, such as: central nervous system damage, skull bone pathology, facial pain syndrome, vasculitis, invasive fungal sinusitis, nasal liquorice should be performed.

System development

For a accurate definition and understanding of the system processes, a sequence diagram was used. For the role creation operation, the user initially administers the search queries at the patient's address and displays the search results if that user is found, then after confirming the entry, the client application passes the server to the address that the user has and the new role for him. After that the server sends editing queries to the user with this address and if the operation was successful, then the client application sends a successful response (Fig.3). For a clinical analysis operation, the doctor first introduces a list of symptoms that have been detected and finds the patient at his initials. Then, this list of confirmed symptoms with the patient ID is passed to the server, where he first finds the symptoms that are denied to ours. Further, the user ID finds the recorded history, and to what knowledge they lead. Then, from the database, a graph of illnesses that includes these symptoms and signs of history are loaded. All illnesses that are denied leave the further analysis, if the disease is not overlooked, it is believed to be probable with documented facts. Then in a disease with the greatest probability is characterizing its symptom, which contains a confirmatory question, which will be sent to the doctor to support the dialogue. Calculated illnesses and the next question with the disease ID will be sent to the doctor to maintain a dialogue. The Diagnosis Diagram is presented on (Fig.4).

An analysis of the presentation of symptoms and signs of disease in diagnosis

From the preliminary consideration of the above problems to solve the problems posed in this study, it is necessary to use methods of system analysis, system engineering intellectual modeling and algorithmization taking into account uncertainty, multi-criteria and fuzzy logic.

Hierarchical relations reflect the stages Diagnostic algorithm of the diagnostic procedure: data entry, data analysis, syndrome diagnosis, and design of the results.

These important remarks should be taken into account in this study. Thus, the diagnostic process is a complex multilateral procedure, which includes various stages of the analysis of clinical information [4,5]. In the most generalized form, the diagnostic algorithm can be represented in this way (Fig.5).

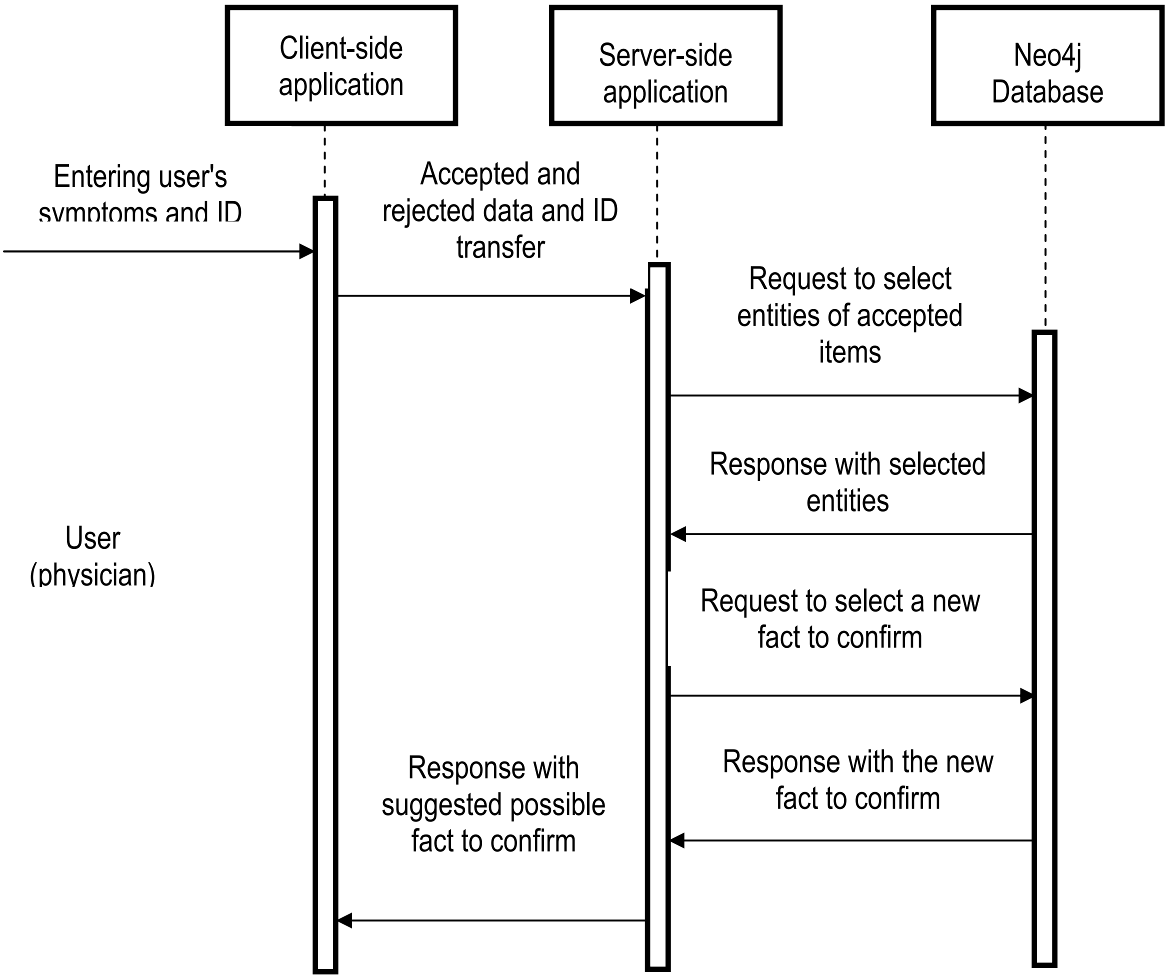

Client-side application
Server-side application
Neo4j Database
Entering user's symptoms and ID
Accepted and rejected data and ID transfer
Request to select entities of accepted items
Response with selected entities
User (physician)
Request to select a new fact to confirm
Response with the new fact to confirm
Response with suggested possible fact to confirm

Figure 3 - Chart of

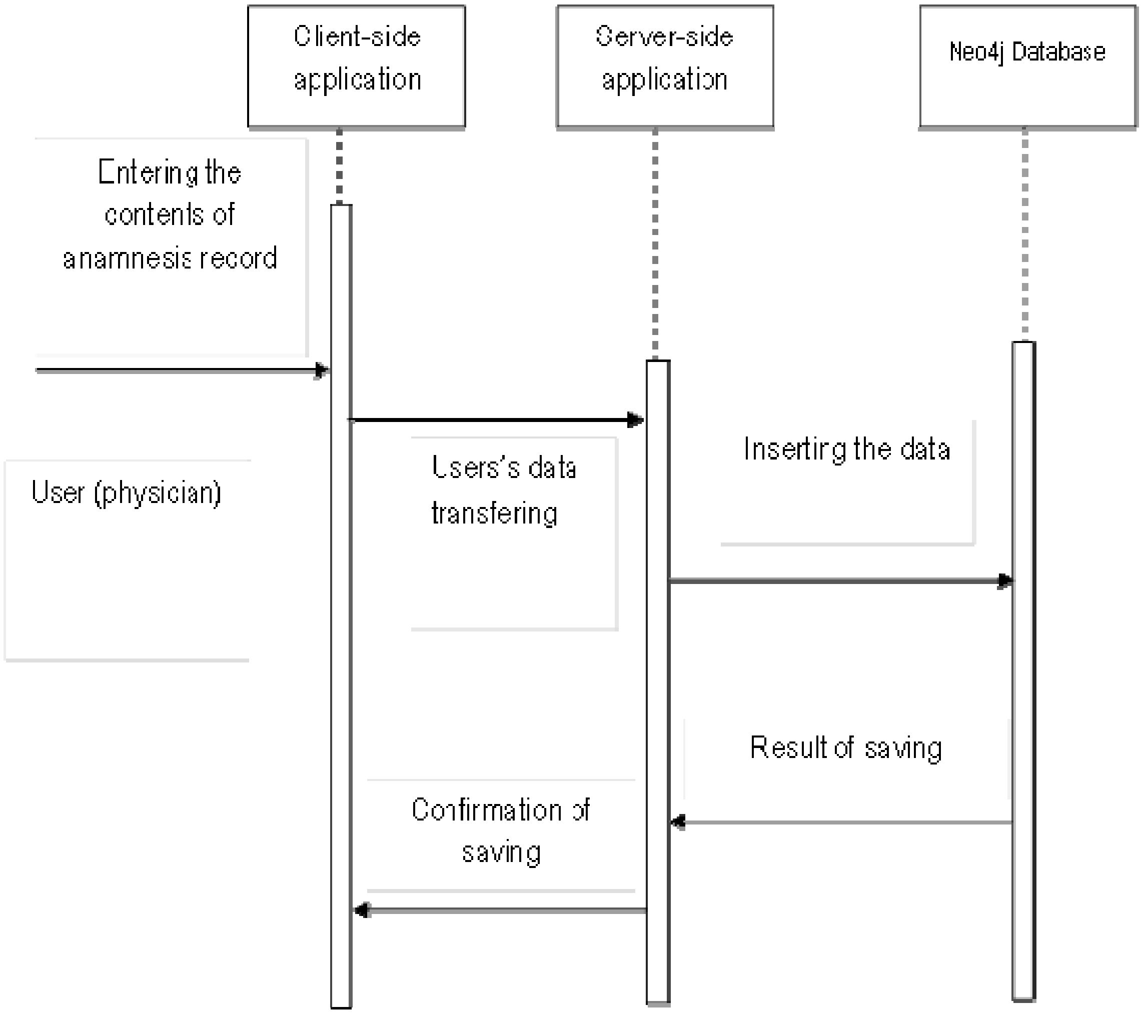


Figure 4 - Diagram of adding a history record

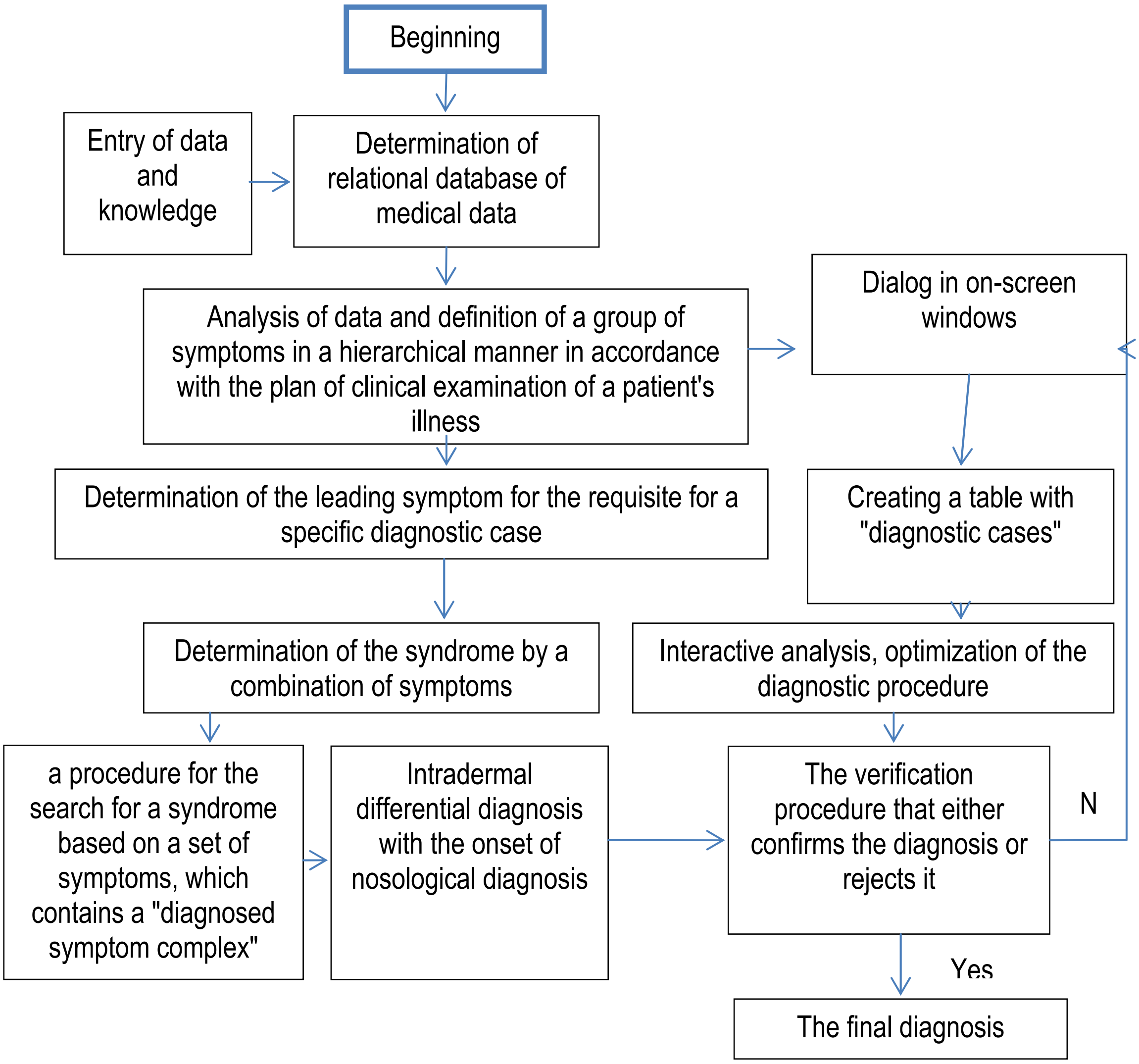

Beginning
Entry of data and knowledge
Determination of relational database of medical data
Analysis of data and definition of a group of symptoms in a hierarchical manner in accordance with the plan of clinical examination of a patient's illness
Dialog in on-screen windows
Determination of the leading symptom for the requisite for a specific diagnostic case
Creating a table with "diagnostic cases"
Determination of the syndrome by a combination of symptoms
Interactive analysis, optimization of the diagnostic procedure
a procedure for the search for a syndrome based on a set of symptoms, which contains a "diagnosed symptom complex"
Intradermal differential diagnosis with the onset of nosological diagnosis
The verification procedure that either confirms the diagnosis or rejects it
N
Yes
The final diagnosis

Figure 5 - Diagnostic algorithm

Conclusion

The resolving of the set of tasks allowed to get the following results:

- For the first time a method has been developed for the distribution of clinical medical conditions into dangerous and safe classes, which determine the most important factors of influence on the clinical condition of the patient, which makes it possible, through the measure of proximity of microsituations (with the greatest influence of parameters), to propose a methodology for predicting the patient's condition;
- For the first time, a reliable backup method for storage of biomedical Big Data has been developed, which in practice provided high reliability in comparison with existing methods;
- Method of constructing a structure of expert systems has been acquired the further development for clinical medicine, which is distinguished by the repeated use of the ontology of successful results and provides an opportunity to increase the reliability and speed of processing of input data, as well as the efficiency of decision-making;
- Situational model of clinical medicine has been improved for analyzing the patient's condition, which, unlike existing approaches, uses situational presentation of a crisis situation based on the three "doctor who manages the influence or decision for re-use of ontology-patient", which allows to predict dangerous and safe situations for a patient faster and with greater accuracy than existing models.

Bibliography

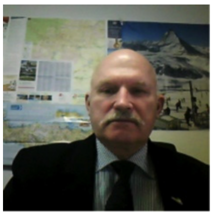
1. [Lukovkina A., 2013], the Big Medical Diagnostics Encyclopedia. 4000 symptoms and syndromes // A. Lukovkina /. Izd. Scientific book. 2013 . P. 363
2. [*Oleksandr Kuzomin. 2017*] DEVELOPMENT OF INTELECTUAL MODELS AND METHODS OF EXPERT SYSTEMS OF CLINICAL MEDICINE.// Oleksya Vasilenko, Tatyana Tolmachova International Journal “Informattion Technologes&Knolegedge”. Volume 11, Namber 2, 2017. PP. 186-199 ISSN 1313-0455 (printed), ISSN 1313-048X (online),

Authors' Information

Oleksii Vasylenko – *Aspirant of Kharkiv National University of Radioelectronics; Kharkiv, Ukraine;*

e-mail: *ichbierste@gmail.com tel.: +380 63 841 66 23*

***Major Fields of Scientific Research**: General theoretical information research, Knowledge Discovery and Engineering, Business Informatics.*



Prof. Dr.-hab. Oleksandr Kuzomin – *Informatics chair of Kharkiv National University of Radio Electronics; Kharkiv, Ukraine Ukraine;*

e-mail: *kuzy@daad-alumni.de tel.: +38(057)7021515*

***Major Fields of Scientific Research**: General theoretical information research, Decision Making, Emergency Prevention, Data Mining, Business Informatics.*

Tatiana Khripushkina – *Master student in Information and Communication technologies of Kharkiv National University of Radio Electronics; Kharkiv, Ukraine*

e-mail: *khripushinka@gmail.com*

***Major Fields of Scientific Research**: General theoretical information research, Knowledge Discovery and Engineering, Business Informatics.*

FORMING MEDICAL DATABASE AND KNOWLEDGE FOR DIAGNOSTIC DISEASE

Oleksii Vasilenko, Oleksandr Kuzomin, Vladislav Shvets

***Abstract**: The paper presents the system for intelligent analysis of clinical information. Authors describe methods implemented in the system for clinical information retrieval, intelligent diagnostics of chronic diseases, patient's features importance and for detection of hidden dependencies between features. Results of the experimental evaluation of these methods are also presented. **Background**: Healthcare facilities generate a large flow of both structured and unstructured data which contain important information about patients. Test results are usually retained as structured data but some data is retained in the form of natural language texts (medical history, the results of physical examination, and the results of other examinations, such as ultrasound, ECG or X-ray studies). Many tasks arising in clinical practice can be automated applying methods for intelligent analysis of accumulated structured array and unstructured data that leads to improvement of the healthcare quality.*

***Keywords**: data mining, knowledge search, intellectual system, databases.*

Introduction

In general, intelligent medical data processing systems have the following applications [Baranov A.A, 2016]: prediction, classification of clinical cases (diagnosis), search of similar clinical cases, observation of patients' condition.

One of the main areas of application of the intellectual system of medical data analysis is the differential diagnosis of the patient's condition: the discovery of the disease, its stage, the nature of the course of the disease. It is necessary to provide for a step-by-step diagnosis of a patient's illness with a precise diagnosis of the patient at each step.

Another important area of application is the prognosis of changes in the clinical condition of the patient in the application of different types of interventions and in the absence of interventions.

Another important area of application of the system is the tracking of dangerous - critical - changes in patient health indicators.

In addition to the main areas of application, the system will have to monitor and automatically evaluate medical personnel's actions.

Data analysis

For medical diagnostics, the methodology for using data analysis as a working tool is intensively developed during the creation of medical knowledge databases. This work is carried out in many research centers [Baranov A.A, 2016], while the ways of solving individual issues have been slightly modified, but the principled approach to building knowledge bases organized in the form of a hierarchical tree of concepts is maintained. Recently, in order to simplify the work of the doctor, a filling was added to the primary card (questionnaire) that was implemented on the computer screen. The doctor fills in a questionnaire with as simple features as possible, and the process of forming concepts is carried out using intelligent computer solutions (Fig. 1).

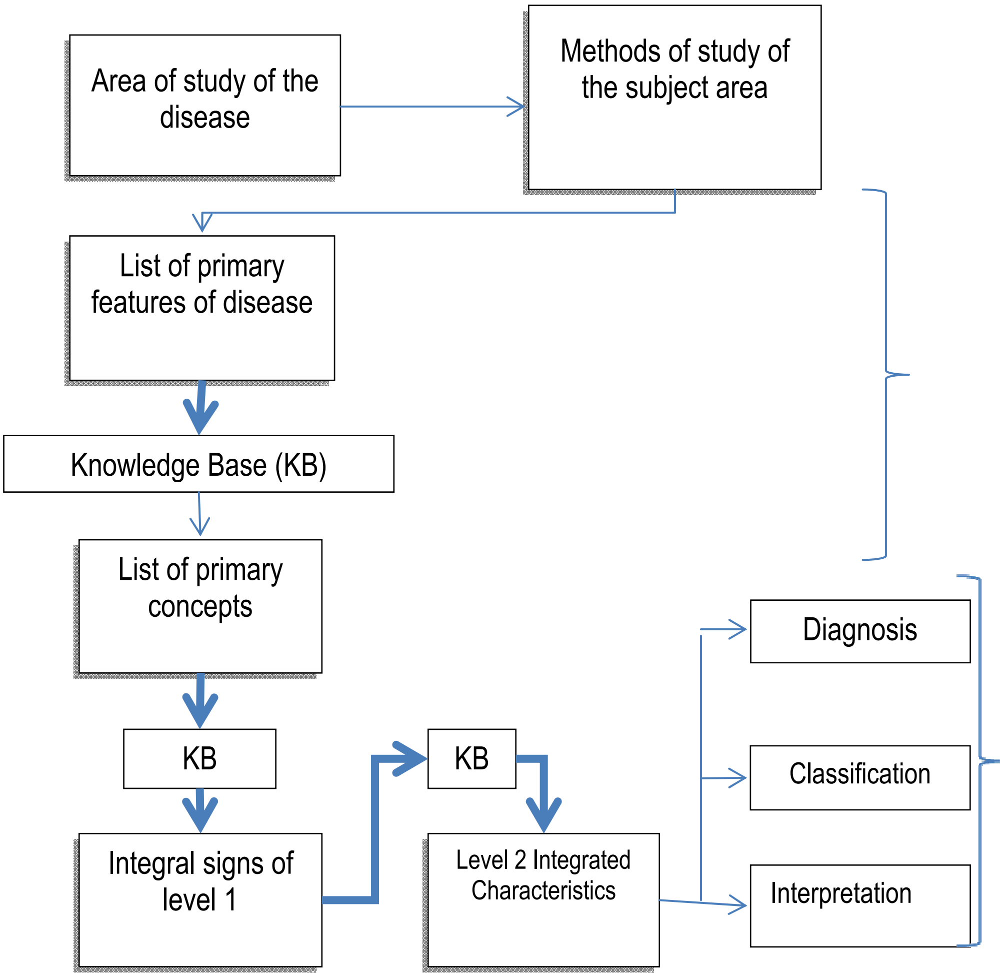

Area of study of the disease
Methods of study of the subject area
List of primary features of disease
Knowledge Base (KB)
List of primary concepts
KB
KB
Integral signs of level 1
Level 2 Integrated Characteristics
Diagnosis
Classification
Interpretation

Figure 1 - The basic scheme of the work of a doctor and a cassette recorder

At the first stage of knowledge discovery, the area and direction of the study, within or outside which is expected to be further developed, that is, the "Field of study of the disease," is determined. This can be a study of only one process or mechanism, for example, a diagnosis of a disease or a group of diseases, one description of the history of the disease or the study of the possibilities of one method.

At the next stage, it is recommended [Kuzomin and Vasylenko, 2014] to analyze internal knowledge blocks, that is, to identify "Methods or aggregates of techniques that examine one side of the investigated phenomenon, process, disease."

This procedure is performed expertly on the basis of objective possibilities and subjective assessments. Next, for each block of the method of research or a set of techniques that were intended to clarify one process, a list (list) of signs of "integral features of the 1st level" is being compiled, as, for example, mechanical features are considered, for example, breathing when detecting sinusitis.

This list includes all the data from the history of the disease, which may subsequently be included in the computational or logical decision-making procedures selection of the diagnosis. Thus the space of the primary features of "Integral signs of the 1st level" is formed.

In general, the analysis of poorly structured medical data contained in the history of the disease allows us to come to the assertion that we have: all the diversity of numbers, curves, images, clinical findings in different forms, in different formats. First of all, it must be transformed into two main types: numbers and texts. A specialist working with images and curves provides a doctor with a conclusion on the appropriate level of medical technology.

The text, officially included in the history of the disease, is both a medical and a legal document. In principle, the database can be formed in any way. It can contain non-formalized source texts, output values, curves, images (if computing allows). In this case, the work of a cryptographic recorder with a specialist on the formation of the primary space of characters may be conducted through a dialogue through the display screen. During the dialog, you can call the screen images, curves, texts from the database that are needed for analysis. If the computer does not have such a database, the work is carried out directly with the texts of medical stories, curves and images on various material media (paper, film, photography).

The next step is to move from the list of signs to the primary concepts that will form the basis of the information matrix "medical object - a sign" and serve as the basis for building medical advisory systems. The process of transition from the signs to the primary concepts is complex and time-consuming. In the course of it it is necessary to do some iterations and apply different approaches.

Therefore, the task of this study is to bring numbers and texts into a single metric information. This problem is solved in this way. When working with numbers, all numerical sampling is based on the numerical scale from the minimum to the maximum values. Then on this scale, the limits of the norm are allocated. The task of determining the limits of the norm is one of the most difficult medical problems. Here you should refer to the materials of publications [Baranov A.A, 2016].

Within the norm, it is possible to allocate three basic gradations: a typical core and two boundary forms that correspond to the notions of the upper and lower limits of the norm. Some researchers [Kuzomin and Vasylenko, 2014] distinguish more gradations (minnorma, optonorma, maxinorm, etc.). It is most effective to use the so-called dependency functions, which corresponds to the fuzzy nature of the data [Kuzomin and Vasylenko, 2014].

Of course, here it is also necessary to use the terminology commonly accepted by clinicians. Naturally, the boundaries between the levels of concepts are rather conditional, but their allocation is important for the following reasons: among a number of well-known doctors, there is the opinion that, given the relativity of the concepts of "norm" and "pathology", it is necessary to consider these processes as unified [Kuzomin and Vasylenko, 2014].

The real difficulties in determining the limits of the norm and pathology are well known, but still confuse the concept of "norm" and "pathology" can not be. The art of the doctor in many respects lies in the assessment of these shaky boundaries. In their definition, an important role is played by a conscious or unconscious idea of the other aspects of the body's vital functions or manifestations of the disease. Such well-used cytochemical parameters of blood relative to the norm allowed to more clearly understand what is happening in pathology [Baranov A.A, 2016].

In the presence of sufficient material, the methods of data analysis help to investigate the limit values, identify them with subclinical forms of diseases. Therefore, in this study, taking into account the above-mentioned features of medical data, existing methods need to be modified in the direction of more reliable results.

The next procedure for analyzing medical data is that the numerical series is transformed into a conceptual series [Baranov A.A, 2016]. Expert way the whole scale is divided into intervals (graduations), each grade is given a meaningful description.

When working with texts, cognitive science is faced with the fact that the doctor determines the same phenomenon in different words, depending on a number of subjective and objective reasons. This discrepancy can be observed even within the same conclusion of the same doctor. Consequently, the cognitive scientist must, "having traveled along the trail" of the expert doctor, compile a dictionary of his definitions, and then try to find out from the physician their contents, build chains of definitions and rank them according to the increasing severity of the process.

For example, a cytologist, describing the sputum, can describe its color, note the presence of blood streaks, the presence of erythrocytes among other cells. In this case, describing the color as rusty, noting the presence of erythrocytes or blood streakscitologist actually describes the same phenomenon: the presence of pulmonary hemoptysis.

As a result, it is possible to come to the chain of concepts and transform the unformed non-rendered description into ranks of the ranked concepts (absence of hemoptysis, small hemoptysis, severe hemoptysis, etc. in an ordered conceptual series.) (Table 1) gives a number of examples of the construction of primary concepts.

Table 1 - A series of examples of constructing primary concepts for GDS

Primary (notions that form) signs	Primary notions	Rules for the formation of primary concepts
Number of neutrophils in the field of vision (NF)	The sputum is slippery Slime-purulent sputum Purulent-slippery scrotum The sputum is slippery	NF up to 30 NF from 30 to 100 NF from 100 to 200 NF 200 and more
Sputum color	Hemorrhage	Sputum is light, no streaks of blood, red blood cells are absent.
The presence of red blood cells	Small hemopus	Sputum is light, without streaks of blood, red blood cells.
The presence of streaks of blood	Intact hemopus	Blood prickles in sputum or rusty sputum
Residual volume of lungs (ZO), residual volume of lungs (ZEL), age	Sharp increase in non-inflammatory parts of the lungs	ZO / ZEL more than 0.5 at the age of 30 years ЗО / ЗЕЛ more than 0.6 at the age from 30 to 40 years ЗО / ЗЕЛ more than 0,63 at the age over 40 years

Thus, the concept of the primary concepts as a unit of knowledge, which in the context of this task is not subject to further specification or division, is formed. From the above, we have that the primary concepts are formed by the cogneologist in conjunction with the expert doctors on the basis of the primary features that are the conceptually constructive elements. In this case, the first part of the knowledge base contains the definition of primary concepts, their clinical interpretation.

So, in the end, after these steps, we can have information that can:

• Be oriented to a generally accepted interpretation without any changes;

• have a clarifying character with the addition of additional elements to achieve an appropriate level of accuracy;

• have a look of illustration with a demonstration of a typical painting. New terms may be introduced (especially when describing new or little developed research methods).

To move from primary features to primary concepts requires a certain knowledge base. It includes a set of logical rules of the form

"IF, TERMS OF DEFINITION"

and may include a set of easiest computing procedures. So, to determine the concept of "lengthening the period of filling the right ventricle" according to jugular phlebography, primary characteristics are required: the duration of the spacing of the spacecraft on the ECG and the spacing of the UA on the phlebogram. In addition, the knowledge base should contain a regression equation that describes the relationship between QC and CA in healthy people, the magnitude of the standard deviation relative to this regression line "b". Then the process of determining the concept of "extension of the period of filling the right ventricle" involves calculating the deviation of the observed UA from the proper at this CK in the particles "b". As a result, the primary concepts become formalized and form the basis of the table "object - a sign." The software support of this part of the knowledge base should provide ease in the modifications of the definitions of the primary concepts for setting the system to work conditions in a particular medical institution.

The second part of the knowledge base is formed as a result of constructing the concepts of higher levels of abstraction of integral features (classifications). Under the integral sign is the complex concept of a high level of abstraction, which is based on a set of primary concepts, constructed on a strictly formal basis, using multidimensional methods of automatic classification. In some cases, the concept may be a logical refinement of existing medical classifications that have had the character of typologies or may lead to a new classification [Kuzomin and Vasylenko, 2014].

Often the integral sign coincides in its content with the medical concept of clinical syndrome [Kuzomin and Vasylenko, 2010]. Thus, the proposed methodology in [Kuzomin and Vasylenko, 2014] considers the construction of knowledge bases of consulting or expert systems as a process of bringing the totality of data and knowledge in the selected subject domain into an ordered hierarchical structure. "Raw" data from the history of the disease through the knowledge base of the first level transform into the conceptual series. At the next stage, the concepts that are primary and considered as data and using the knowledge base of the next level are transformed into integral signs, actually closes the results and their interpretation within one block of research. At the next stage, these integral signs are considered as data and using a higher level knowledge base used to classify, construct diagnostic findings, predict the course of diseases, and so on.

This methodology facilitates the interpretation of known knowledge and pushes a specialist to heuristic decisions in terms of explaining pathophysiological mechanisms, processes and dependencies. As to the use of this methodology in our study, all the main stages of data analysis will be the same, but their essence will have development in the direction of intellectualization using neurons structural algorithms.

In addition, several important points to be borne in mind when solving the problems that were formulated in the works of leading researchers in the field of computer-aided diagnostics:

- First, during the care of each patient in the clinic, the growth of electronic medical records (EHR) [Baranov A.A, 2016] and clinicians increase access to large volumes of data that not all doctors have the right, timely and qualitative use of this information.
- Secondly, BD - the big data requires the organization of their management, ensuring the reliability of storage, constant updating, connection with search robots.
- Thirdly, CDS, which is a computer support for clinical decisions that can change the dynamics of information at the patient's bed and have the opportunity to improve the quality of treatment.
- At the same time, the growth of the use of electronic medical records (EHR), which allows clinicians to increase access to large volumes of data collected during the care of each patient. This combination has led to some overload of the physician with information that challenges the physician's ability to focus on the necessary information, adjust this information to clinical practice standards, and use a combination of clinical data and medical knowledge to assist the patient with the best available medical evidence.

A side effect of the EHR population with patient data is the creation of large data warehouses that contain the accumulation of various clinical data. An analysis of these sets of data collections can give an idea of the nature of the disease and may indicate which of the available diagnostic and therapeutic approaches are likely to yield the desired results. Moreover, it is the information that is needed to support the creation of computer-assisted clinical decision support (CDS), which can change the dynamics of information development at the patient's bedside.

In order to use this information to develop CDS applications, you need to increase resources for obtaining data from data warehouse, analyzing it, and creating decision support tools that can help maintain patient care. This usually requires the collaboration of clinicians, database analysts, statisticians / data miners and software developers. This commitment of large resources is a key obstacle to the widespread use of data stored in clinical repositories for the development of decision support applications.

The literature [Baranov A.A, 2016] describes data analysis environments that are considered for the use of ontological data and knowledge. These systems are intended to demonstrate the means by which it is possible to use ontologies in conjunction with specialized data analysis programs.

By the way, this can reduce the requirements for the resources required to develop CDS diagnostic software. In the references this environment is called: "ontological system of diagnostic modeling" (ODMS) [Baranov A.A, 2016] (Fig. 2). Considering what we said in our study should take into account the most successful decisions regarding the use of medical ontological data on the disease and the construction of data warehouse (EDW).

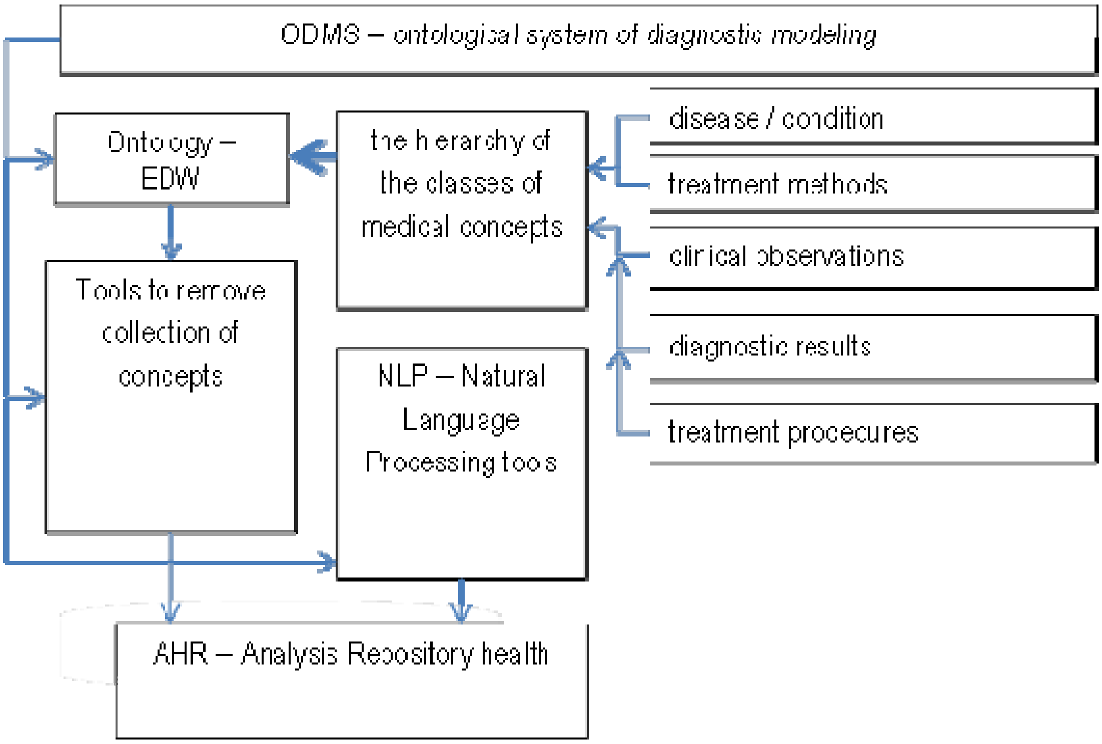

ODMS – ontological system of diagnostic modeling
Ontology – EDW
the hierarchy of the classes of medical concepts
disease / condition
treatment methods
clinical observations
diagnostic results
treatment procedures
Tools to remove collection of concepts
NLP – Natural Language Processing tools
AHR – Analysis Repository health

Figure 2 Ontological system of diagnostic modeling

Automation of the search for signs of illness

At present, the idea of automating the search for signs of a disease for symptoms has become relevant and have beneficial results [Baranov A.A, 2016]. However, when considering the problem of identifying symptoms, attention should be paid to the existence of uncertainty, subjectivity in assessing the patient's condition. Necessary understanding of the problem of identifying symptoms in contact with the patient's doctor. Some aspects of communication with the patient when setting up a preliminary survey are reflected on (Fig. 3).

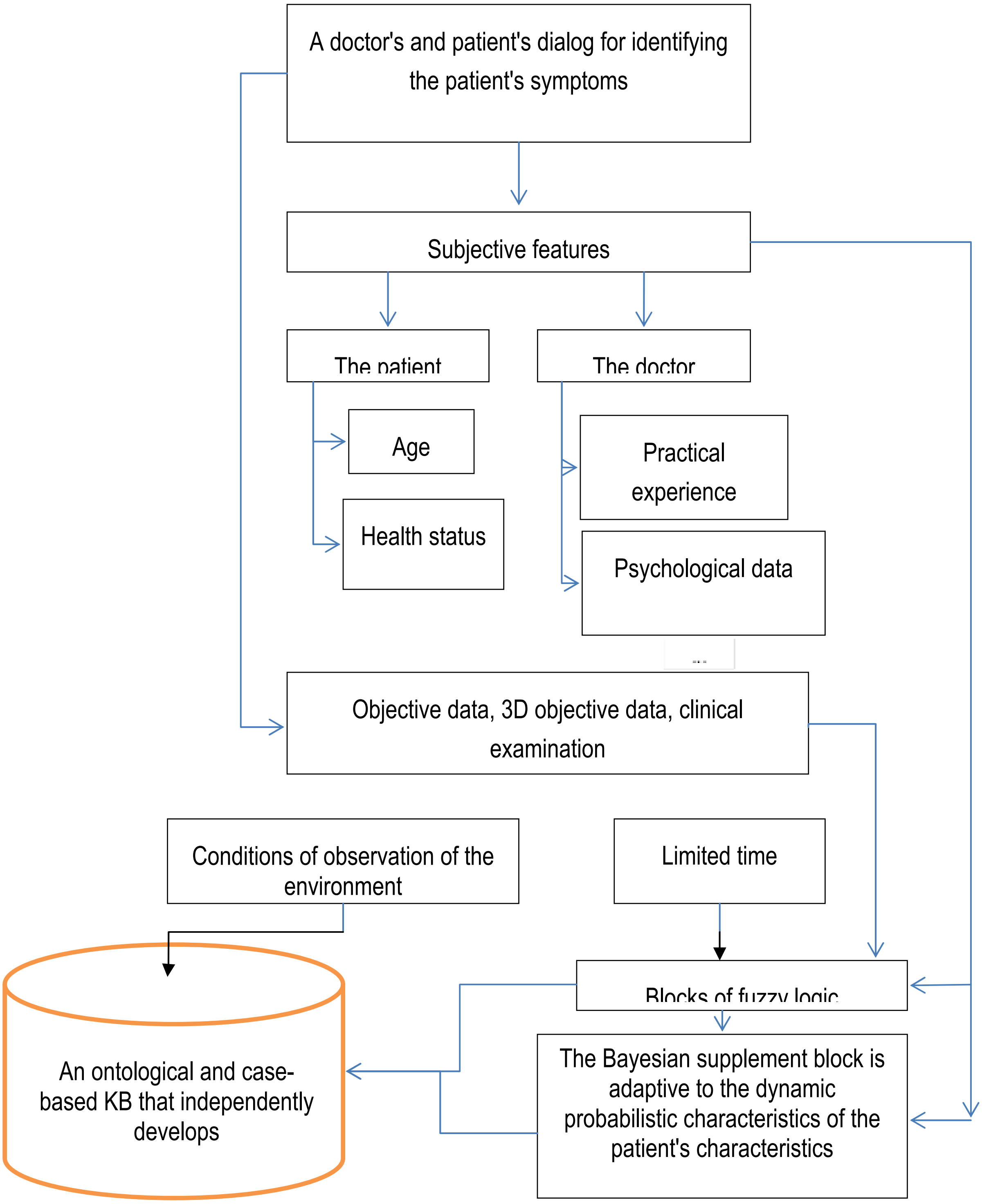

A doctor's and patient's dialog for identifying the patient's symptoms
Subjective features
The patient
The doctor
Age
Health status
Practical experience
Psychological data
Objective data, 3D objective data, clinical examination
Conditions of observation of the environment
Limited time
Blocks of fuzzy logic
An ontological and case-based KB that independently develops
The Bayesian supplement block is adaptive to the dynamic probabilistic characteristics of the patient's characteristics

Figure 3 - Formation of the basis of symptoms in the expert system of questioning and preliminary diagnosis

Conclusion

On the basis of the intellectual analysis of medical data and proposed typical directions of application, primary tasks were identified in developing an intellectual system for diagnosing diseases. Most significant of them:

1) Search for structured data upon request.

2) Search of hidden logic and statistical regularities in given sets of medical data.

3) Classification and prediction of signs of patients on the basis of revealed patterns for solving generalized problems of diagnosis and prognostication.

4) Structured Data Grouping.

5) Work with super massive data.

6) Linguistic analysis of text documents.

7) Search for similar text medical documents in the natural language.

Bibliography

[Baranov A.A, 2016] Технологии комплексного интеллектуального анализа клинических данных.// Баранов А.А., Намазова-Баранова Л.С., Смирнов И.В., Девяткин Д.А., Шелманов А.О., Вишнёва Е.А., Антонова Е.В., Смирнов В.И./ *Вестник РАМН.* 2016;71(2):160–171. doi: 10.15690/vramn663)

[Kuzomin and Vasylenko, 2014] Kuzomin, O.Ya., Vasylenko, O., Obespechenie bezopasnosti ispolzovaniia baz dannyh v usloviiah chrezvychainyh situazii. International Journal "Information Technologies Knowledge", Vol. 8, Num. 2. 2014. pp. 173-187.

[Kuzomin and Vasylenko, 2010] Kuzomin, O.Ya., Vasylenko, O., Analiz estestvenno iazykovyh obiektov I predstavlenie znanii. Vostochno-Evropeiskii zhurnal peredovyh technologii, Vol. 6/2(48). 2010.

Authors' Information

Oleksii Vasylenko *– Aspirant of Kharkiv National University of Radioelectronics; Kharkiv, Ukraine;*

e-mail: ichbierste@gmail.com tel.: +380 63 841 66 23

Major Fields of Scientific Research: General theoretical information research, Knowledge Discovery and Engineering, Business Informatics.

Prof. Dr.-hab. Oleksandr Kuzomin *– Informatics chair of Kharkiv National University of Radio Electronics; Kharkiv, Ukraine Ukraine;*

e-mail: kuzy@daad-alumni.de tel.: +38(057)7021515

Major Fields of Scientific Research: General theoretical information research, Decision Making, Emergency Prevention, Data Mining, Business Informatics.

Vladislav Shvez*– Master student in Information and Communication technologies of Kharkiv National University of Radio Electronics; Kharkiv, Ukraine Ukraine;*

email: Vladyslav.shvets@nure.ua

Major Fields of Scientific Research: Big Data, Data Mining, Data Analyses.

AUTOMATED TESTS FOR ERRORS IN COMPUTER SYSTEM ‘ENVIRONMENT’

Oleksii Vasylenko, Oleksandr Kuzomin

***Abstract**: This paper describes technology of automated testing for complex computer systems like ‘smart house’, etc. The basic idea to abstract from natural environment to cyber analogue.*

Existing predicates contained in the semantic network to the new analysis may serve as benchmarks. Thus, the proposed recognition model of natural language objects can be faster and more efficiently than the existing ones.

***Keywords**: knowledge acquisition, SAP, DoD, Automated Testing, knowledge processing, automated knowledge management system, Internet, Intranet, logical testing, Smart testing, Complex Computer System (CCS), Selenium –Bamboo.*

Introduction

Recently, one often hears about creating smart homes and automation of everyday life. Many of today tend to make "smart" every detail of daily life. But few of us think about what a great information systems can make mistakes. And the more the system - the more expensive the cost of failure or malfunction. In order to monitor the performance of the whole agglomeration area connections of information systems need a comprehensive and versatile solution [Kuzomin and Vasylenko, 2014].

Imagine the future. One ‘smart house’ is connected to another one. Each ‘smart house’ – independent smart complex computer system. We have the network, which based on other, smaller networks. Lets call them ‘basic element’ or element level 1. The next level of this system is communication of these elements from level 1. I will call it “elements level 2”. Finally, we have correlation of elements level 2, which create special “environment”.

On the other hand, everybody knows what is it the ‘state of Emergency’. Where is it possible to happen? Nature, Factories, etc. What is it? The hurricane, earthquake, floods, landslides, etc. When they going to happen? Everytime. No one know date and time. Everybody afraid and tries to make the prognosis for the next Emergency.

We live in 21st Century so why we think only about the common Environment? Why we postpose the one which is going to stay the most important for ages now. I will call it ‘Cyber Environment’.

Computers, networks, complicated systems – everything is part of this new Environment. Lets take a look here from the point of Earth (Natural) Environment investigation. We will find all things common. Why we don't investigate the emergencies which are in this 'CyberInvironment'?

I purpose the Test Solution, based on Atlassian Bamboo, running by Selenium server. Selenium is still actively supported (mostly in maintenance mode) and provides some features that may not be available in Selenium 2 for a while, including support for several languages (Java, Javas Script, Ruby, PHP, Python, Perl and C#) and support for almost every browser out there.

First, we will describe how the components of Selenium RC operate and the role each plays in running your test scripts. e-mail for questions: *ichbinerste@gmail.com*

The essence of work

Task formulation:

1) Develop intellectual language to express, statistical tracking the errors, data mining, based on CCX automated logical test results .

2) Develop model of complex result/dependence matrix (model: test – error tracked - reason – feature request)

3) Develop general model of Emergency Forecasting (EF) for basic CCS

4) Develop logical intellectual language for automated decision making based EF for basic CCS

5) Develop complex decision making matrix for velocity & quality increasing for EF and trouble shooting in CCS

6) Create the feature request system logical language based on EF

Scientific innovation: The practical novelty is in the development of the intelligent decision-making language based on EF for CCS. Basic principle is completely new and based on systematical automated finding and tracking the errors (Emergences in CCS environment), creating the tracking and data mining model to find and keep logical depended data for Emergence practice in CCS environment. This data will be a row input for automated EF and Decision Making for Emergency prevention in CCS.

Goals and objectives of my work: The aim is to develop a model of knowledge extraction by running 'smart' tests in huge computer systems. The aim is to create system with automated testing and reporting, trouble shooting and prognosis stages set up. The main aim for project is to make life easier by giving the hard and boring job for machine and results for us – people (Fig.1)).

There are basic stages of my work and several 'pre-aims' of the whole project:

1) Create test logic level 1. (deep investigation for test-ready fields of computer system environment)

2) Create test logic level 2. (look for test-connection ready stages)

3) Create test logic level 3. (whole self-dependant smart test architecture network projecting)
4) Final test logic stage. Automated smart-unitTest builds creating. Creating whole smart testing network for each possible error.
5) Coding and realising the 'road-map' for the smart unitTest build.
6) Creating the test-analysis algorithm and automatisation for the whole project

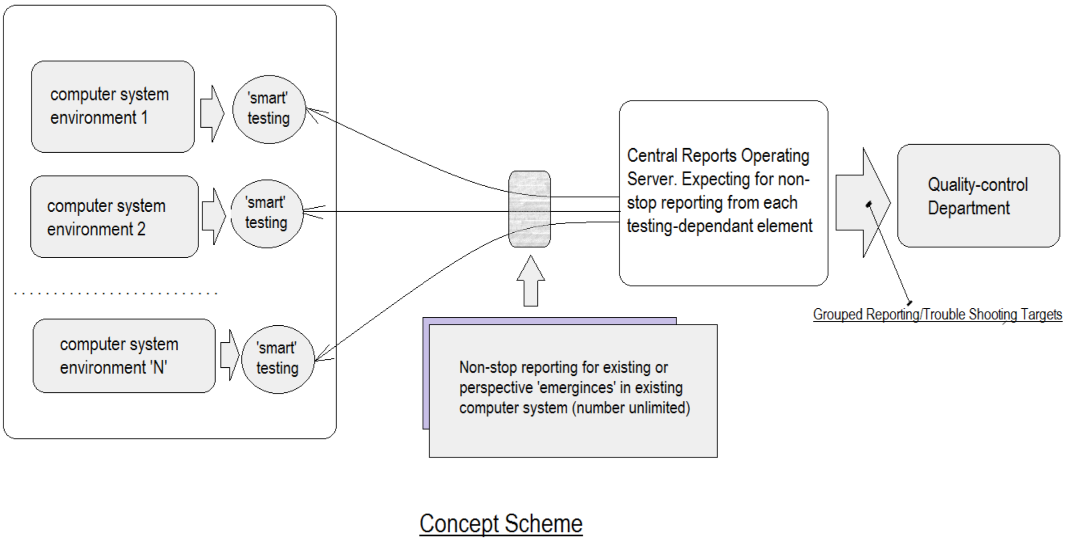

computer system environment 1
'smart' testing
computer system environment 2
'smart' testing
computer system environment 'N'
'smart' testing
Central Reports Operating Server. Expecting for non-stop reporting from each testing-dependant element
Quality-control Department
Grouped Reporting/Trouble Shooting Targets
Non-stop reporting for existing or perspective 'emergineces' in existing computer system (number unlimited)
Concept Scheme

Figure 1 – Concept Scheme

Sphere of application:

1. Information intelligence. Data mining for emergency predicting environment. 'Smart' prognosis for computer systems for life support, etc.
2. Bamboo is the central management server which schedules and coordinates all work.
3. Bamboo itself has interfaces and plugins for lots of types of work.

Generating Idec load

In order to build up load in a structured, guided ways from SAP there are a few ideas of what can be done. My initial hopes, were to push IDocs from a load generator to the message broker. This would be the easiest way in which to control the flow of data towards the broker and thus the easiest way to make sure we are fully in control of how busy the broker is. Alas, when talking to the guys behind the broker interfaces it turned out that this method would not work for the setup used. The only way the broker would actually do something with the IDocs was if it could pull them from the SAP RFC port, pushing to the broker would not work, since the RFC receiving end of the broker is not listening, it is pulling.

Alternatively sending data off into the message queue would fill up the MQ, but not help with getting the messages pushed through the Broker, again, due to the specific setup of the Enterprise Service Bus which contains the broker interfaces.

So alternatives needed to be found. One obvious alternative is setup a transaction in SAP which generates a boat-load of IDocs and sends the to the RFC port in one big bulk. This would generate a spike, such as shown in this image, rather than a guided load. In other words, this is not what we want for this test either. It might be a useful step for during a spike test, however the first tests to be completed are the normal, expected load tests.

The search for altneratives continued. At my customer, not a lot of automation tools were available, especially not for SAP systems. One tool however has been purchased a while ago and apparently is actively used: Win Shuttle

Win shuttle seems to be able to generate the required load, based on Excel input, the main issue with Win shuttle however, was the lack of available licenses. There are two licenses available and both are single use licenses. This meant I would have to find a way to hijack one of the PC's it was installed on, script my way through it and run the tests in a very time boxed manner. In other words, not really a solution to the problem. The resulting load ramp up was a linear ramp up of IDocs being generated and sent to the SAP RFC port, where they were picked up by the Message Broker and subsequently transformed and sent back to the Locus system, where the load turned out to be quite on the high side.

All in all this was a fun excercise in automating SAP to do something it is absolutely not meant to do with a tool not built nor designed to do what it did.

Definition of Done

In the majority of the DoD's I have seen, one of the items is something referring to "tests automated". The thing I have thus far not seen however, is the team adding as much value to the automation code as they do to the production code. Quite a lot of DoD's refer to certain coding standards, however these standards often seem to not apply to functional test automation. Isn't your functional automation code also just code? If so, why then should this not be covered in code reviews, be written according to some useful guidelines and standards and hopefully use a framework to make the code sustainable (Fig.2)?

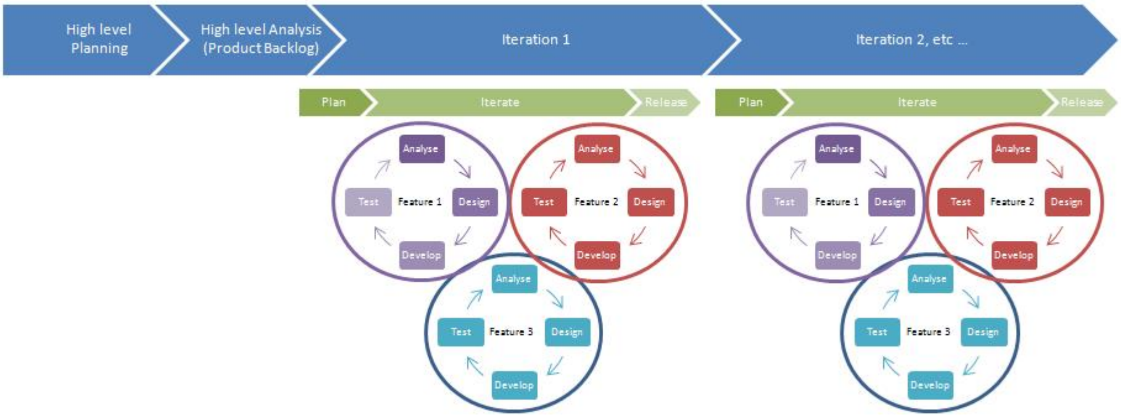


Figure 2 – Agile Development Cycle diagram for logical testing

There aren't really any distinct stages during the development. Instead, each feature is taken from start to finish within an iteration, with the software being released at the end of each iteration, or if appropriate even during an iteration.

An iteration is simply a fixed, short period of time that the team chooses to work within. Typically for agile teams, an iteration is between 1 week and 30 days. Strictly speaking, the Scrum agile development methodology advocates 30 days, but I've encountered very few teams that actually do this. 2 or 3 weeks seems to be more common. The Extreme Programming agile methodology advocates 1 week. This is very short and in my experience requires quite a lot of maturity in the team and its processes to achieve, because getting to a stable release every week can be difficult.

Either way, the principles are the same. The idea is to stick to short, fixed-length iterations and complete all stages of the development cycle for each feature in turn within an iteration. Basically, idea is to SWOPE teams of people by teams of hardware. The same cycle, the same responsibility stages & reporting.

Scientific novelty: The model extraction knowledge by using the automated smart unit testing environment is pretty much new. The main logic for this – make the technical support stage much more easier than it is now. Smart test will be able to run all scheduled time, give the report and make each part of the system tested by time, make it ready and non-forgiven for the possible mistakes, that are able to destroy the whole component or system at all. Aim for this project is automation for system support part. As huger computer system is becoming – as harder and more expensive support level becoming as well. That is why we have this big challenge to develop new strategic for keeping system's support 'eyes' clear and ready to react.

Practical value: Bamboo is a continuous integration (CI) server that can be used to automate the release management for a software application, creating a continuous delivery pipeline.

CI is a software development methodology in which a build, unit tests and integration tests are performed, or triggered, whenever code is committed to the repository, to ensure that new changes integrate well into the existing code base. Integration builds provide early 'fail fast' feedback on the quality of new changes.

Release management describes the steps that are typically performed to release a software application, including building and functional testing, tagging releases, assigning versions, and deploying and activating the new version in production.

Selenium Usage: Selenium Server receives Selenium commands from your test program, interprets them, and reports back to your program the results of running those tests.

The RC server bundles Selenium Core and automatically injects it into the browser. This occurs when your test program opens the browser (using a client library API function). Selenium-Core is a JavaScript program, actually a set of JavaScript functions which interprets and executes Selenium commands using the browser’s built-in JavaScript interpreter.

The Server receives the Selenium commands from your test program using simple HTTP GET/POST requests. This means you can use any programming language that can send HTTP requests to automate Selenium tests on the browser.

Client Libraries

The client libraries provide the programming support that allows you to run Selenium commands from a program of your own design. There is a different client library for each supported language. A Selenium client library provides a programming interface (API), i.e., a set of functions, which run Selenium commands from your own program. Within each interface, there is a programming function that supports each Selenium command.

The client library takes a Selenium command and passes it to the Selenium Server for processing a specific action or test against the application under test (AUT). The client library also receives the result of that command and passes it back to your program. Your program can receive the result and store it into a program variable and report it as a success or failure, or possibly take corrective action if it was an unexpected error.

So to create a test program, you simply write a program that runs a set of Selenium commands using a client library API. And, optionally, if you already have a Selenium test script created in the Selenium-IDE, you can generate the Selenium RC code. The Selenium-IDE can translate (using its Export menu item) its Selenium commands into a client-driver's API function calls. See the Selenium-IDE chapter for specifics on exporting RC code from Selenium-IDE.

Logical Testing Organizing:

1. The client/driver establishes a connection with the selenium-RC server.
2. Selenium RC server launches a browser (or reuses an old one) with a URL that injects Selenium-Core's JavaScript into the browser-loaded web page.
3. The client-driver passes a Selenium command to the server.
4. The Server interprets the command and then triggers the corresponding JavaScript execution to execute that command within the browser. Selenium-Core instructs the browser to act on that first instruction, typically opening a page of the AUT.
5. The browser receives the open request and asks for the website's content from the Selenium RC server (set as the HTTP proxy for the browser to use).
6. Selenium RC server communicates with the Web server asking for the page and once it receives it, it sends the page to the browser masking the origin to look like the page comes from the same server as Selenium-Core (this allows Selenium-Core to comply with the Same Origin Policy).
7. The browser receives the web page and renders it in the frame/window reserved for it.

Many applications switch from using HTTP to HTTPS when they need to send encrypted information such as passwords or credit card information. This is common with many of today's web applications. Selenium RC supports this.

To ensure the HTTPS site is genuine, the browser will need a security certificate. Otherwise, when the browser accesses the AUT using HTTPS, it will assume that application is not 'trusted'. When this occurs the browser displays security popups, and these popups cannot be closed using Selenium RC.

When dealing with HTTPS in a Selenium RC test, you must use a run mode that supports this and handles the security certificate for you. You specify the run mode when your test program initializes Selenium. In Selenium RC 1.0 beta 2 and later use *firefox or *iexplore for the run mode. In earlier versions, including Selenium RC 1.0 beta 1, use *chrome or *iehta, for the run mode. Using these run modes, you will not need to install any special security certificates; Selenium RC will handle it for you.

In version 1.0 the run modes *firefox or *iexplore are recommended. However, there are additional run modes of *iexploreproxy and *firefoxproxy. These are provided for backwards compatibility only, and should not be used unless required by legacy test programs. Their use will present limitations with security certificate handling and with the running of multiple windows if your application opens additional browser windows.

In earlier versions of Selenium RC, *chrome or *iehta were the run modes that supported HTTPS and the handling of security popups. These were considered â€˜experimental modes although they became quite stable and many people used them. If you are using Selenium 1.0 you do not need, and should not use, these older run modes (Fig.2).

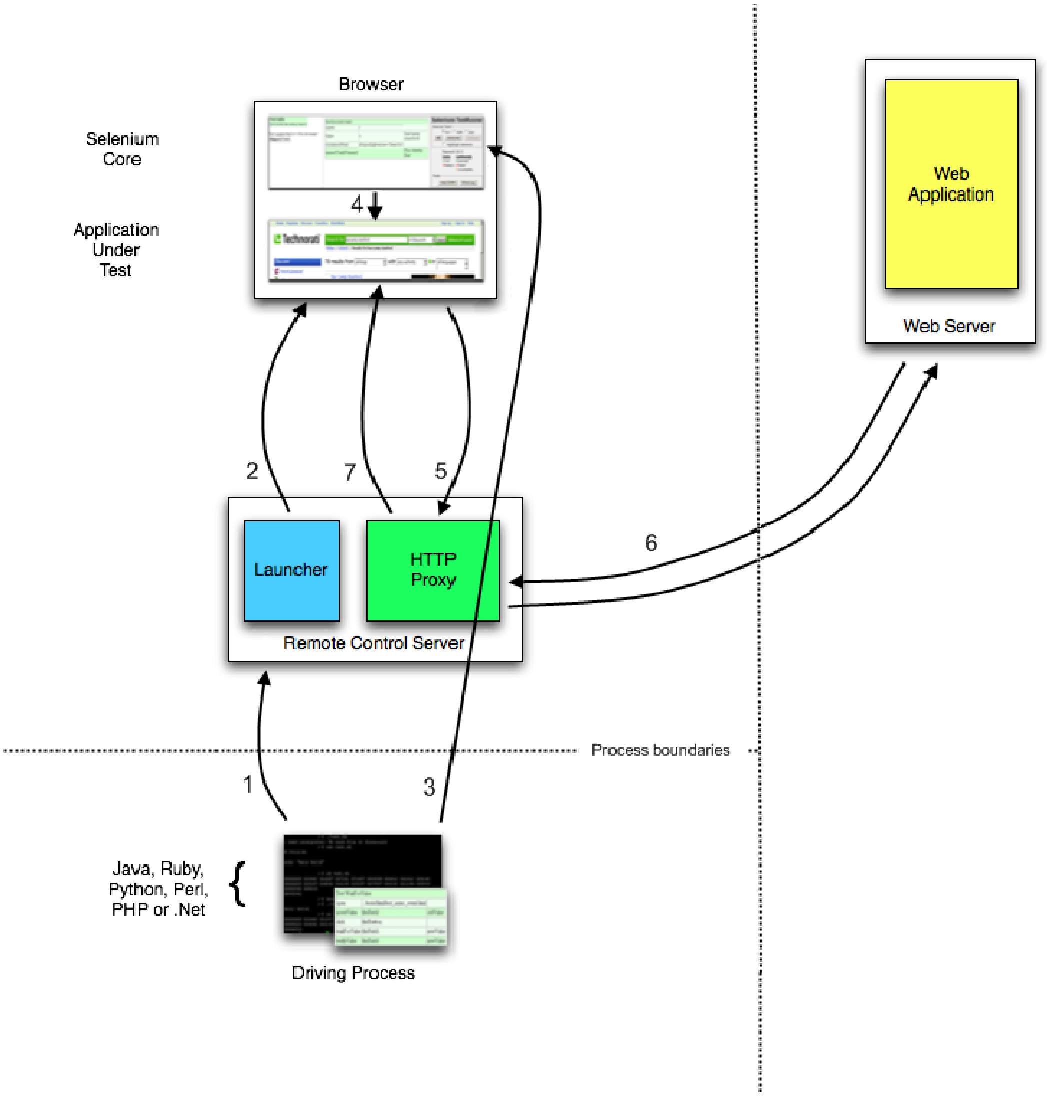


Figure 2 - Security popups

Work Flow organization

Bamboo uses the concept of a 'plan' with 'jobs' and 'tasks' to configure and order the actions in the workflow.

Project Has one, or more, plans.
Provides reporting (using the wallboard, for example) across all plans in the project.
Provides links to other applications.

Plan Has a single stage, by default, but can be used to group jobs into multiple stages.
Processes a series of one or more stages that are run sequentially using
the same repository.
Specifies the default repository.
Specifies how the build is triggered, and the triggering dependencies between the plan and other plans in the project.
Specifies notifications of build results.
Specifies who has permission to view and configure the plan and its jobs.
Provides for the definition of plan variables.

Stage Has a single job, by default, but can be used to group multiple jobs.
Processes its jobs in parallel, on multiple agents (where available).
Must successfully complete all its jobs before the next stage in the plan can be processed.
May produce artifacts that can be made available for use by a subsequent stage.

Job Processes a series of one or more tasks that are run sequentially on the same agent.
Controls the order in which tasks are performed.
Collects the requirements of individual tasks in the job, so that these requirements can be matched with agent capabilities.
Defines the artifacts that the build will produce.
Can only use artifacts produced in a previous stage.
Specifies any labels with which the build result or build artifacts will be tagged.

Task Is a small discrete unit of work, such as source code checkout, executing a Maven goal, running a script, or parsing test results.
Is run sequentially within a job on a Bamboo working directory.

The proposed method: After carefully considering all the existing methods, capabilities and operating time, We can interpretive our hike to understand the meaning of natural language by computer.

The concepts can replace each other on the principle of consistency of meaning. It is above all that are essential to accelerate the process of text recognition as well - reducing the load on the knowledge base. (Kuzemin A., thesis work). That is – replaceability of the concepts significantly reduces the volume of the knowledge base by reducing the number of own concepts as its components. Create the new theorem replace ability of the concepts.

Theorem 1. If the concept Po_2 inherits the concept $\mathrm{Po}_1 - \mathrm{G}(\mathrm{Po}_2, \mathrm{Po}_1)$ in a certain category of concepts C, the notion of Po_2 can substitute for a mikrosituations concept Po_1 without losing the semantic load and informative of this mikrosituation [Kuzomin and Vasylenko, 2010].

Proof. In accordance with the structure and strategy categories identify the concept to identify the concept Po_2 must affirmatively respond to the decision rules that relevant concepts $\mathrm{Po}_1, \mathrm{Po}_{k_1}, \ldots, \mathrm{Po}_{k_n}, \mathrm{Po}_2$ This means that identified the problematic concept as we have passed these decision rules, including p_1 which refers to the notion Po_1 . Hence, the notion Po_2 may be perceived as a concept Po_1, possessing its characteristic features.

It is possible to obtain the final simplified formula

To begin to define a mathematical model for this area, that is, a preliminary algorithm of the program domain.

Thus, we set the field, which we consider to find items that need to be put in the text as a priority. Items that are behind them before the end of the line, will be the ones most desired predicates that become the nodes of a semantic network specification.

So, try to express this simple mathematical formula:

Consider the above described concept $t_{i_m j}$ - the word in a sentence t_j determinants in appliance has a_i and τ_i:

a_{i_m}, τ_{i_m} when m=1 (definite value) <=> a_{i_1}, τ_{i_1} - given, = const

(The above mentioned node predicates, which is searched, that is - nodes ontology similarity Product RCO).

If the nodes of ontologies are not specified clearly (for thematic units characterized by peculiar and persistent concept), then to search for meaning in a sentence erants, first highlight the main predicate, then place it in a network node.

Consider the 1-st case with known predicates:

Suppose there is $t_{i_m j}$, that

$$(a_{i_1}, \text{т}_{i_1} \in t_{i_m j}, \{t_{i_m j_1}, t_{i_m j_2}, t_{i_m j_3}, \ldots, t_{i_m j_n}\}) \in t_j =>$$

$$\Rightarrow \quad [\Pi_n] = \{t_{i_m j_2}, \ldots\ldots t_{i_m j_n}\},$$

Where Π_n - predicate found meeting the criteria a_{i_1} and т_{i_1}

Consider Case 2, where the proposal is a set of elements without an explicit predicate. In this case, I have proposed for the allocation of nodal predicates analyze the proposal for the parts of speech, then find the subject and the predicate method for counting the frequency of occurrence of noun and a verb. Mathematically, it is possible to express this:

We have a set of words

$$\{t_{i_m j_1}, t_{i_m j_2}, t_{i_m j_3}, \ldots, t_{i_m j_n}\} \in t_j,$$

Let each of them has an additional parameter *α* - the frequency for each element, as well as the parameter responsible for the definition of the speech clearly designed combinations of endings – p (look application "A", list of terminals), where (1…n) ∈ p, we have a structure:

$$\{ {t_{i_m j_{p\,(1\ldots n)}}}^{\alpha}, {t_{i_m j_{p\,(2\ldots n)}}}^{\alpha}, {t_{i_m j_{p\,(3\ldots n)}}}^{\alpha}, \ldots, {t_{i_m j_{p(n)}}}^{\alpha}\} \in t_j,$$

Moreover, if found by the predicate coincides with the already existing sites - it is not analyzed in the future, and put in its place. If the predicate is unique - it is analyzed as part of speech and related items - similar to the ongoing analysis of combinations.

After receiving the new value Π_n, determine its meaning - it is entered in the knowledge base, which compares to the value of existing concepts $\mathrm{Po}_{1\ldots n}$. Further processing of meaning is on a similar principle, but with reference to a specific predicate.

Conclusion

In this work the method of analysis of natural language objects, which accelerates the work with large volumes of the analyzed verbal text. By itself, the semantic network is a self-learning, as accumulating predicates, new to them in their meaning. Existing predicates contained in the semantic network to the new analysis may serve as benchmarks. Thus, the proposed recognition model of natural language objects can be faster and more efficiently than the existing ones.

Bibliography

[Kuzomin and Vasylenko, 2014] Obespechenie bezopasnosti ispolzovaniia baz dannyh v usloviiah chrezvychainyh situazii.// Kuzomin, O.Ya., Vasylenko, O., International Journal "Information Technologies Knowledge", Vol. 8, Num. 2. 2014. pp. 173-187.

[Kuzomin and Vasylenko, 2010] Analiz estestvenno iazykovyh obiektov I predstavlenie znanii // Kuzomin, O.Ya., Vasylenko, O. / Vostochno-Evropeiskii zhurnal peredovyh technologii, Vol. 6/2(48). 2010. PP. 60-64.

Authors' Information

Oleksii Vasylenko – *Aspirant of Kharkiv National University of Radioelectronics; Kharkiv, Ukraine;*

e-mail: *ichbierste@gmail.com tel.: +380 63 841 66 23*

Major Fields of Scientific Research*: General theoretical information research, Knowledge Discovery and Engineering, Business Informatics.*

Prof. Dr.-hab. Oleksandr Kuzomin – *Informatics chair of Kharkiv National University of Radio Electronics; Kharkiv, Ukraine Ukraine;*

e-mail: *kuzy@daad-alumni.de tel.: +38(057)7021515*

Major Fields of Scientific Research*: General theoretical information research, Decision Making, Emergency Prevention, Data Mining, Business Informatics.*

DEVELOPING METHODS BASED ON TEXT MINING TECHNOLOGY TO IMPROVE THE QUALITY AND SPEED OF AUTOMATIC CLUSTERING OF DOCUMENTS

Oleksii Vasilenko, Oleksandr Kuzomin, Artem Mertsalov

Abstract: *The goal is to develop methods based on the use of Text Mining technology, which allows to improve the quality and speed of automatic clustering of documents.The object of the research is the intellectual analysis of the text array.The methods of numerical simulation and analytical substantiation are used. The research of methods of intellectual analysis of the text was conducted; Methods of preprocessing of text, selection of keywords and classification of documents are considered.As a result of implemented software implementation of the system analysis of the text array.*

The aim of the work is to study of ranking algorithms on data from Web Archives. A developed software use web archives and extracted data to create a graph of relations between web pages. A HITS algorithm applied on the graph, allows to find meaningful hubs among the pages. The hubs allow calculating authority of each page and ranking them by using the value.

Link pairs from German Web Archive Data are used as input data for graph (incoming and outgoing links), and German Wikipedia articles as search topics to evaluate results.

The results evaluated using survey with results of HITS algorithm in comparison with results of Bing search engine and competing algorithm PageRank.

The work use Java programming language for implemented algorithms and support software, the destination graph stored in-memory by using Redis. The data extracted from Web Archives by using Hadoop framework and stored in Hive database.

Keywords: *web archives, web search, ranking, hyperlink induced topic search, page rank.*

ITHEA Keywords: *NNP - proper noun, AJ - general adjective, DT - general determiner, NLP - natural language processing, E.2 Data Storage Representations, H.3.1 Content Analysis and Indexing.*

Introduction

Automated exclusion of knowledge from the text is one of the main tasks of artificial intelligence and is directly related to the understanding of texts in the natural language. Since the mid-50s of the last century, considerable efforts of scientists have been directed at the development of mathematical algorithms and computer programs for the processing of texts in the natural language [Bansal, 2014]. To automate the analysis and synthesis of texts, various patterns of text processing were created, as well as corresponding algorithms and data representation structures. Traditionally, the analysis of natural language texts was presented as a sequence of processes - morphological analysis, syntactic analysis, semantic analysis. For each of these steps, appropriate models and algorithms were created. For semantics of the text - the classical semantic networks and the framed models of Minsk, for the syntax of the sentence - the chronological grammar, the systemic grammar of the Hollywood, the trees of subordination and the system of components of Gladky, the expansion of the network of transitions; For morphological analysis, many different models have been developed that focus on specific language groups.

The task of automated analytical processing of text information is trying to solve many foreign and domestic scientists. In particular, in 1979 Kuzin N.T. [Ridings, 2002] described the methods of frequency subtraction of textual information, which were subsequently improved in the works of A. Broder and D.V. Lande [Bansal, 2014] In his manual A.A. Barseghyan and MS Kupriyanov summarized data on modern methods of automatic analysis of Data Mining and Text Mining. However, none of the described methods does not provide extraction from text information knowledge. In studies AI Vavilenkova provided a description of the main methods of Data Mining, highlighting their advantages and disadvantages, emphasizing that none of the described methods is capable of removing knowledge from the information. In this work the researcher demonstrated the work of the Robinson resolution method for comparing two simple sentences; the algorithm of comparison of logical-linguistic models of text information on the content is proposed.

Thus, one can distinguish the main problems associated with the need to optimize the modeling and development of methods for analyzing textual information:

- The rapid growth in the amount of information contained on the Internet is the cause of increasing and growing difficulties in finding the necessary documents and organizing them in structured, structured, content-based repositories;

- most technologies of work with text documents focus on the organization of convenient work with information for a person, but virtually no opportunity to convey the semantic content of the text, that is, there is no semantic indexing;

- To effectively address the problem of search, it is necessary to broaden the concept of a traditional document: the document must link the knowledge that allows to interpret and process the data that is retained in this document;

- Unstructured information is a significant part of modern electronic text documents.Web Archives contains web pages from the past years and save new pages for future researchers, historians, and the public. They are very important for learning how interned is developing. They allow to use knowledge from the past and apply it today to extract new knowledge and very popular for hundreds of research tasks as playground.

Search is the most demanded tool in today's web. There is a trillions of pages stored somewhere in the web and search engines are like road signs before the navigator was developed. By typing simple search term, they navigate to destination pages. A web search engine is a software system that is designed to search for information on the World Wide Web. The common approach for web search engines is to analyze content of the pages in the web and create index for them. Then, using that index and different retrieval methods, they select all pages that match a required search term and to return relevant results they apply ranking algorithms to order results. All the same is true for web archives, but in addition, they also contain some specific attributes, which include crawling information and a history of the page. That can be used to improve search results quality.The algorithms are improving with years and modern search engines use very advanced technology with Artificial Intelligence (AI) and machine learning. But unfortunately, those high-end algorithms only available for lead companies that specified on search engines.

Therefore, the goal of the work is to apply available ranking algorithms on Web Archives and investigate potential ability to improve search results by including new attributes to the algorithm.

Motivation and problem statement

Improving search results is interesting and demanded task, which can be applied on Web Archives for research. In the work, one of the popular algorithms for ranking search results is applied on German Web Archives and learned how archived data can change search results. Despite the fact that for the web search simple algorithms are no longer enough, nevertheless they are increasingly used for research, personal and low-middle level commercial purposes. They should have good enough results and lower maintenance costs.

As a problem statement we have got web archives that contains web pages (articles, news, etc.), which stored as natural language text (unstructured text). With the purpose of building an efficient search results, we are confronted with following problems:

1. Different formats of URLs and aliases.
3. Using diacritic characters and language specific characters in URLs.

2. A huge amount of data, which should be processed in real-time.
4. Slow processing time for Web Archives.

Topic of research

Recent development of the Internet and computing technologies makes the amount of information increasing rapidly. That is why it is necessary to retrieve the best of the web pages that are more relevant in terms of information for the query entered by the user in search engine. In recent years, semantic search for relevant documents on web has been an important topic of research. Many semantic web search engines have been developed that helps in searching meaningful documents presented on semantic web. To relate entities, texts and documents having same meaning, semantic similarity approach is used based on matching of the keywords, which are extracted from the documents. For example, groups of authors presented a new web ranking system by using Semantic Similarity and HITS algorithm along with AI technique [Bansal, 2014]. In this paper, author proposed Intelligent Search Method (ISM) - a ranking system with improved HITS and Semantic Similarity techniques. It is used to rates the web pages and also known as Hubs and Authorities. A good hub represented a page that pointed to many other pages and a good authority represented a page that was linked by many different hubs. Therefore, its authority value, which estimates the value of the content of **the page, and its hub value, which estimates the value of its links to other pages.**

Author developed new method to index the web pages using an intelligent search strategy in which meaning of the search query is interpreted and then indexed the web pages based on the interpretation. Comparison of HITS Algorithm, Semantic Similarity Algorithm and ISM method is shown in (Tab. 1).

Table 1 – Comparison of Techniques

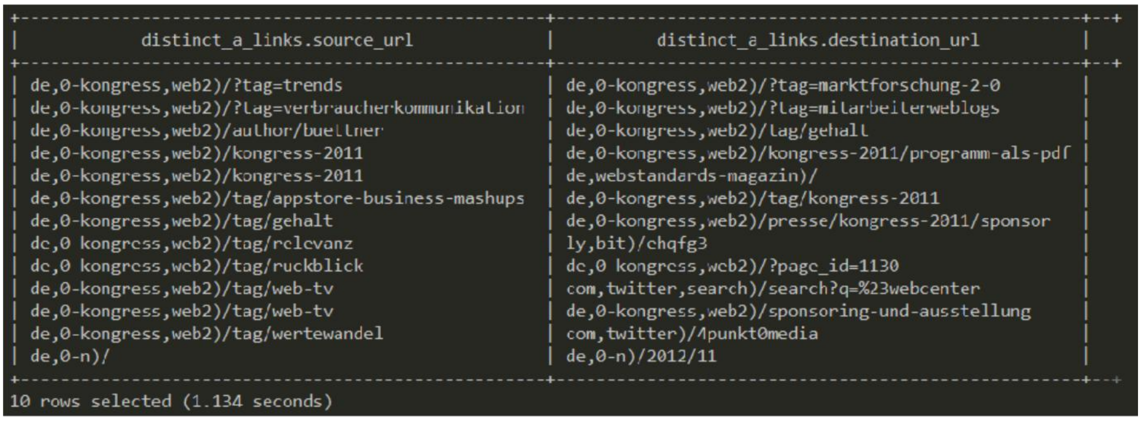
Parameter / Technique	HITS Algorithm	Semantic Similarity Algorithm	Proposed System
Time Efficiency	72%	87%	91%
Accuracy	79%	91%	95%
User specific Page Generation	No	No	Yes
Relevance Ratio	90%	92%	96%
High Relevance Ratio	30%	41%	51%

New ISM method can be integrated with any of the Page Ranking Algorithms to produce better and relevant search results.

Preparing dataset

On the input of the task, we have Hive table with two columns: source_url and desctination_url. Each row contains one edge of future graph (see Figure 1 for example of DB row).

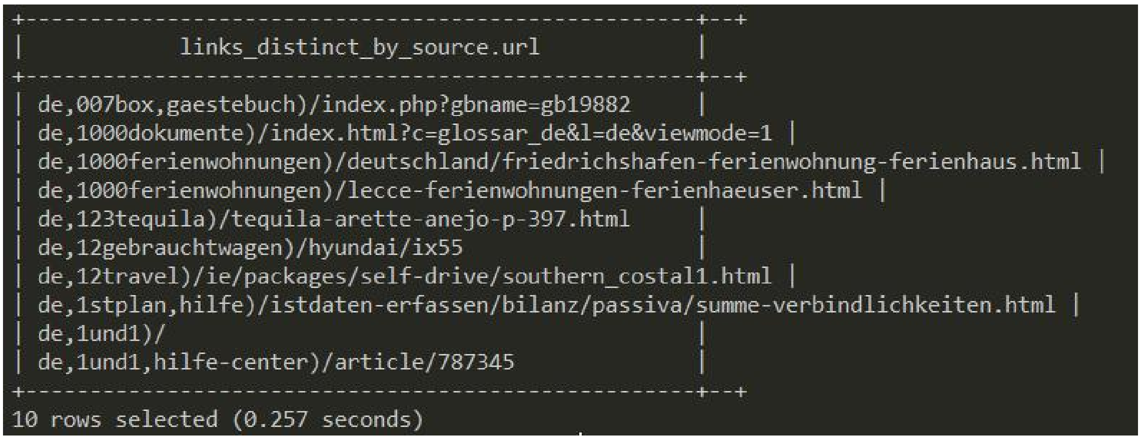
As the size of the graph is very big (more than 130 billion edges, about 10 terabyte size), first we need to optimize it. The first step was to extract all unique URLs from the table and replace them with IDs, then we will have much smaller graph where edges will be long to long instead of string to string.



```
+--------------------------------------------------+--------------------------------------------------+--+
|           distinct_a_links.source_url            |         distinct_a_links.destination_url         |
+--------------------------------------------------+--------------------------------------------------+--+
| de,0-kongress,web2)/?tag=trends                  | de,0-kongress,web2)/?tag=marktforschung-2-0      |
| de,0-kongress,web2)/?tag=verbraucherkommunikation | de,0-kongress,web2)/?tag=mitarbeiterweblogs     |
| de,0-kongress,web2)/author/buettner              | de,0-kongress,web2)/tag/gehalt                   |
| de,0-kongress,web2)/kongress-2011                | de,0-kongress,web2)/kongress-2011/programm-als-pdf |
| de,0-kongress,web2)/kongress-2011                | de,webstandards-magazin)/                        |
| de,0-kongress,web2)/tag/appstore-business-mashups | de,0-kongress,web2)/tag/kongress-2011           |
| de,0-kongress,web2)/tag/gehalt                   | de,0-kongress,web2)/presse/kongress-2011/sponsor |
| de,0-kongress,web2)/tag/relevanz                 | ly,bit)/chqfg3                                   |
| de,0-kongress,web2)/tag/ruckblick                | de,0-kongress,web2)/?page_id=1130                |
| de,0-kongress,web2)/tag/web-tv                   | com,twitter,search)/search?q=%23webcenter        |
| de,0-kongress,web2)/tag/web-tv                   | de,0-kongress,web2)/sponsoring-und-ausstellung   |
| de,0-kongress,web2)/tag/wertewandel              | com,twitter)/1punkt0media                        |
| de,0-n)/                                         | de,0-n)/2012/11                                  |
+--------------------------------------------------+--------------------------------------------------+--+
10 rows selected (1.134 seconds)
```

Figure 1. Sample results from distinct_a_links table

We extracted about 6 billion unique links (see Figure 2 for sample data), to rewrite out initial table from string to string to use short IDs of the pages we need very fast access to database which contains those IDs, therefore we decided to use in-memory databases.
Decoding URLs from SURT format.



```
+---------------------------------------------------+--+
|            links_distinct_by_source.url           |
+---------------------------------------------------+--+
| de,007box,gaestebuch)/index.php?gbname=gb19882    |
| de,1000dokumente)/index.html?c=glossar_de&l=de&viewmode=1 |
| de,1000ferienwohnungen)/deutschland/friedrichshafen-ferienwohnung-ferienhaus.html |
| de,1000ferienwohnungen)/lecce-ferienwohnungen-ferienhaeuser.html |
| de,123tequila)/tequila-arette-anejo-p-397.html    |
| de,12gebrauchtwagen)/hyundai/ix55                 |
| de,12travel)/ie/packages/self-drive/southern_costal1.html |
| de,1stplan,hilfe)/istdaten-erfassen/bilanz/passiva/summe-verbindlichkeiten.html |
| de,1und1)/                                        |
| de,1und1,hilfe-center)/article/787345             |
+---------------------------------------------------+--+
10 rows selected (0.257 seconds)
```

Figure 2. Sample results from distinct_links_all table

A However, 6 billion URLs still was a lot to fit in-memory (about 4 terabyte), we decided to store SHA-1 hash instead of URLs as key and ID long value as value for our Redis database.

During that work, we noticed that some of URLs has different formats, for example, some of them use http and some use https, some use www prefix and some not. We decided to remove such difference and added pre-processing of URLs which include:

1. Removing all unsafe ASCII characters, if they appears in domain names we replace them into Punycode domains and if such characters appears in path then replace them by using "%" followed by two hexadecimal digits.
2. Removing protocol prefixes, like: http, https.
3. Removing www prefixes.
4. Removing port numbers, like: 80, 443.

After replacing URLs to hashes, it has 21 153 collisions on our dataset. The hashes was extracted for investigation.

As we can see in table above, that is false negative results. In the middle column you can see URL as it stored in database, right column as it was normalized and left column is hash of the URL. As we can see, after normalization, we have same strings and as result same hashes. Also original URLs from database are not valid SURT format.

Tble 2. Hash collisions over links dataset

Hasah value	Original URL	Normalized URL
qEb3qCpvWQthRNLdkKTOiHInVmg=	de,merkspruch)/	merkspruch.de/
qEb3qCpvWQthRNLdkKTOiHInVmg=	de,merkspruch,)/	merkspruch.de/
ZLPnZaYBNeerff/5PH5ip3XXq40=	de,wuhletal,kirche)/	kirche.wuhletal.de/
ZLPnZaYBNeerff/5PH5ip3XXq40=	de,wuhletal,http://kirche)/	kirche.wuhletal.de/

Now when assigned short ID (long value) to each URL we need to update our whole dataset of pairs. That should significantly reduce size of it. Also, to create a graph we will need to go over all the pairs again. To decrease number of operations, we created module that read pairs, normalize URL and retrieve its ID from Redis database from previous task.

To create graph, we need to know incoming and outgoing links from each link. Unfortunately, hash table, which we used for previous task can’t store multiple values per single key. Therefore we used two new databases, one which store linkID and list of incoming linkIDs and other one with linkID and list of outgoing linkIDs.

The data stored in Hive database is unsorted, but to decrease number of operation on Redis server we need to order the database by source_url or destination_url depending of which table we want to fill. If the pairs are ordered for example by source_url, then during going through them we can collect all neighbors for same source_url (just compare if previous value is equal current) and merge destination_url values, put them to Redis in single operation.

As we already can access to incoming and outgoing links for any page, we can calculate hubs and authorities, which requires for HITS algorithm [Miller, 2001].

HITS ranking results

To evaluate HITS algorithm we use search results from Bing search engine. We have pre-saved results for all German Wikipedia articles. We selected most popular 3000 pages and used their title as search term. For each search term, we have about 100 search results from Bing.

For our HITS algorithm we use 100 pages from Bing as root set. Then we use base set of pages and all pages which linked or links to them as base set. For that base set we calculate authority and hubs values for thee steps. Then order pages from root set by authority and compare results with original Bing results. We also implemented PageRank algorithm for better evaluation of HITS results. It will add additional set of results to compare. The implementation of PR algorithm is very simple, we use 10 as default score for each page and split the score between all outgoing pages.

There are some limitations associated with Archived Data, for example the actives mainly contains German Internet pages (in .de domain zone), when Bing provides results regardless of the domain zone. But, we still have such pages in results because we have German pages that have outgoing links to different domain zones and we can calculate authority value for them.

The example results of applying HITS algorithm on search term “Kassel” you can see in the (Tab. 3), the comparison with PageRank score displays in (Tab. 4).

As we can see in (Tab. 4), the website of Kassel’s football team has lower PageRank score. That can be explained that in total the website has smaller number of incoming links, but the links is from better sources.

Table 3. Authority and hub values of HITS for search term “Kassel”

Link	Authority	Hub
Kassel Marketing | Tourismus-Informationen für Kassel kassel-marketing.de/	58929	148127497
Wetter Kassel - aktuelle Wettervorhersage wetteronline.de/wetter/kassel	46581	114631041
KSV Hessen Kassel e.V. - Die offizielle Homepage dasbesteausnordhessen.de/	45712	125635663
Kassel: Information für Kassel bei meinestadt.de home.meinestadt.de/kassel-documenta-stadt	45666	125741104

Table 4. HITS Authority and PageRank score

Link	HITS Authority	PageRank Score
Kassel Marketing | Tourismus-Informationen für Kassel kassel-marketing.de/	58929	0.09894597
Wetter Kassel - aktuelle Wettervorhersage wetteronline.de/wetter/kassel	46581	0.027334956
KSV Hessen Kassel e.V. - Die offizielle Homepage dasbesteausnordhessen.de/	45712	0.023083081
Stadtportal - Startseite www.kassel.de kassel.de/	45666	125741104
Kassel: Information für Kassel bei meinestadt.de home.meinestadt.de/kassel-documenta-stadt	45666	0.025142923
Stadtportal - Startseite www.kassel.de kassel.de/	45666	0.025142923

Some other results that illustrate the difference between HITS and PR are displayed in (Tab. 5). Also the results compared to Bing results in (Tab. 6).

Table 5. Results of HITS and PR for search term “Volkswagen AG”

Link	HITS Authority	PageRank Score
Volkswagen AG - Home - SSI SCHÄFER https://www.ssi-schaefer.com/de-de	184900	9.485999 (1)
VOLKSWAGEN AKTIEN News | 766403 Nachrichten… http://www.finanznachrichten.de/nachrichten-aktien/vo...	7462	0.10863955 (5)
Volkswagen Aktie | Aktienkurs | Chart | 766400 wallstreet-online.de/aktien/volkswagen-aktie	6456	0.025147859 (11)
Volkswagen Konzern Startseite volkswagenag.com/	2002	0.82785743 (2)
Volkswagen Personal volkswagen-karriere.de/de.html	1898	0.30051792 (3)

Table 6. Results of HITS and PR for search term “Volkswagen AG”

HITS results	Bing results
Volkswagen AG - Home - SSI SCHÄFER https://www.ssi-schaefer.com/de-de	Volkswagen Konzern Startseite volkswagenag.com/
VOLKSWAGEN AKTIEN News | 766403 Na… http://www.finanznachrichten.de/nachrichten-a...	Wie gut klingt das denn. volkswagen.de/de.html
Volkswagen Aktie | Aktienkurs | Chart | 766400 wallstreet-online.de/aktien/volkswagen-aktie	Volkswagen AG – Wikipedia de.wikipedia.org/wiki/Volkswagen_AG
Volkswagen Konzern Startseite volkswagenag.com/	Volkswagen Group Homepage volkswagenag.com/content/vwcorp/con…
Volkswagen Personal volkswagen-karriere.de/de.html	Volkswagen International de.volkswagen.com/de.html

Evaluating results

To evaluate results we created survey page, which contains ten results from Bing, and ten reordered results by using authority value of HITS algorithm. Also ten results of HITS with ten results of PageRank algorithm. The survey asks users to compare results which is more relative, as they think and make decision by clicking on one of the submit buttons on the bottom. Survey is available online for everyone, we asked some students to participate in and 31 people accept proposal. In average one-person answers for 50 topics, and 1541 in total. The results approximate expectations and Bing search engine provides better results than re-ranked results by using HITS algorithm. The results of that survey illustrated on (Fig. 3).

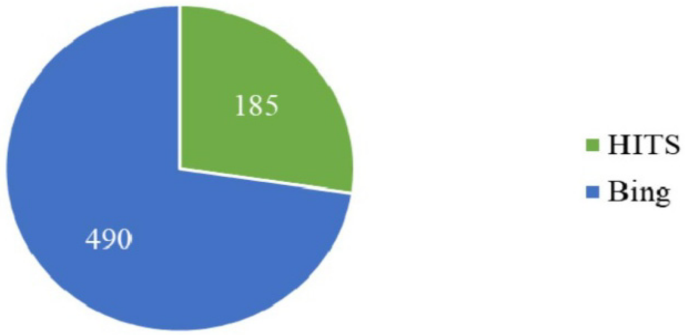


Figure 3. The results of survey HITS vs Bing results

Comparing results of HITS algorithm and PageRank algorithm (see Figure 4) give a little more points in favor of HITS algorithm.

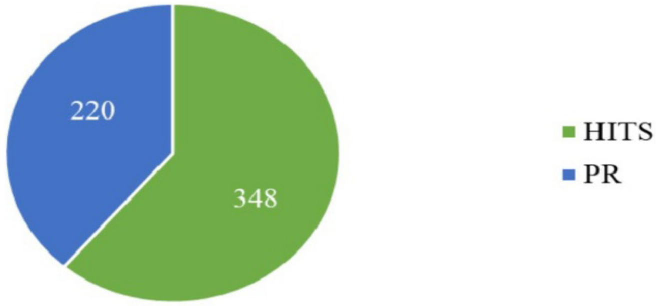


Figure 4. The results of survey HITS vs PR results

Comparing results of PageRank vs Bing (see Figure 5), gives most of the points to Bing results. It should be noted that in comparison with Bing results, HITS results take a bit more points than PageRank results.

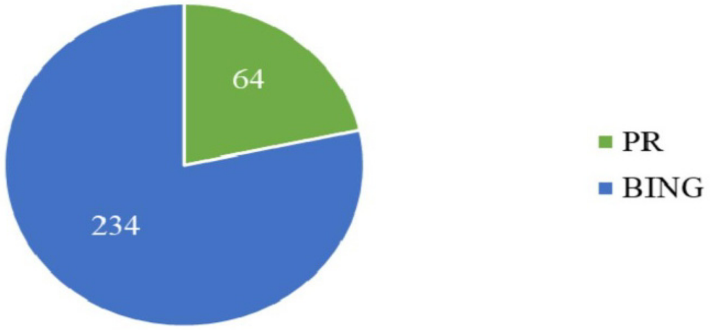


Figure 5. The results of survey PR vs Bing results

As we can see on table 6 the results of HITS and PageRank results worst results than Bing, six of ten top results contains pages with information about company shares. That pages appears on top due to specific content of the Web Archives that was used. For some other research purposes that archives contains a lot of pages with information about trades and shares.

In addition to that, some other results also contains very specific pages only for used Web Archives. Using a simple survey among unprepared users was not the best way to evaluate the quality of the results.

Conclusion

The aim of the work is to begin research in the direction and show some first results.

During the work was implemented two algorithms for ranking web results, the initial HITS algorithm was compared with PageRank algorithm.

Our results confirmed that modern search engines use very sophisticated technologies that include not only ranking algorithms, they also use AI and machine learning techniques to improve our daily Internet search experience.

Nevertheless, HITS algorithm that was developed slightly later than PageRank and using more depth scanning gives relatively better results. And that also gives us motivation to continue our work, we have plans to improve the results.

The amount of data that we have on input is very huge, and several first tries ware failed. We were need to experiment with different techniques and technologies to work with given data. Moreover, even now, when we have prepared relations graph, each iteration in the program must be justified, otherwise, everything works very slowly.

The search in Web Archives is not for everyday use and we do not expected that results will completely satisfy us. The key idea the work is research of additional attributes of web archives, unfortunately, we do not have enough time to present them in the work.

In the work, we finished only first part of our goal. Right now, we implemented simple HITS and PageRank algorithms, they allow us to make some small researches over retrieved data.

The next step will be to include first crawl date and last crawl date into HITS algorithm. Potentially we can find some hubs that existed before, but no longer exists today. By using those properties, we also can know age of the pages and we do not know how it change results.

Acknowledgement

The research was done with the help of L3S Research Center, Hannover, Germany.

Bibliography

[Bansal, 2014] N. Bansal, S. Paramjeet. Improved Web Page Ranking Algorithm Using Semantic Similarity and HITS Algorithm. International Journal of Emerging Trends & Technology in Computer Science (IJETTCS), 2014. pp. 346-348. http://www.ijettcs.org/Volume3Issue4/IJETTCS-2014-08-26-146.pdf

[Sharnagat, 2014] R. Sharnagat. Named Entity Recognition: A Literature Survey, 2014. 27 p. https://pdfs.semanticscholar.org/83fd/67f0c9e8e909dc7b90025e64bde0385a9a3a.pdf

[Ridings, 2002] C. Ridings, M. Shishigin. Pagerank Uncovered, Technical report, 2002. 56 p. http://www.voelspriet2.nl/PageRank.pdf

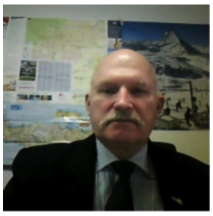
[Miller, 2001] J. C. Miller, G. Rae, F. Schaefer, L.A. Ward, T. LoFaro, & A. Farahat. Modifications of Kleinberg's HITS algorithm using matrix exponentiation and web log records. Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval, 2001. pp. 444-445. https://dl.acm.org/citation.cfm?id=384086

Authors' Information

Oleksii Vasylenko *– Aspirant of Kharkiv National University of Radioelectronics; Kharkiv, Ukraine;*

e-mail*: ichbierste@gmail.com tel.: +380 63 841 66 23*

Major Fields of Scientific Research*: General theoretical information research, Knowledge Discovery and Engineering, Business Informatics.*



Prof. Dr.-hab. Oleksandr Kuzomin *– Informatics chair of Kharkiv National University of Radio Electronics; Kharkiv, Ukraine Ukraine;*

e-mail*: kuzy@daad-alumni.de tel.: +38(057)7021515*

Major Fields of Scientific Research*: General theoretical information research, Decision Making, Emergency Prevention, Data Mining, Business Informatics.*

Artem Mertsalov *– Master student in Information and Communication technologies of Kharkiv National University of Radio Electronics; Kharkiv, Ukraine*

e-mail*: khripushinka@gmail.com*

Major Fields of Scientific Research*: General theoretical information research, Knowledge Discovery and Engineering, Business Informatics.*

TABLE OF CONTENTS OF IJ IMA VOL.7, NUMBER 1

TABLE OF CONTENTS OF IJ IMA VOL.7, NUMBER 2

TABLE OF CONTENTS OF IJ IMA VOL.7, NUMBER 3

TABLE OF CONTENTS OF IJ IMA VOL.7, NUMBER 4