



Речевые

ТЕХНОЛОГИИ

3/2012

Главный редактор Харламов А.А., доктор технических наук

Состав редколлегии:

Потапова Р.К., доктор филологических наук, профессор,
заместитель главного редактора

Ронжин А.Л., доктор технических наук, доцент

Женило В.Р., доктор технических наук, профессор

Жигулёвцев Ю.Н., кандидат технических наук, доцент

Кривнова О.Ф., доктор филологических наук, профессор

Кушнир А.М., кандидат психологических наук

Лобанов Б.М., доктор технических наук (Беларусь)

Максимов Е.М., доктор технических наук

Голенков В.В., доктор технических наук, профессор (Беларусь)

Ромашкин Ю.Н., кандидат технических наук

Петровский А.А., доктор технических наук, профессор (Беларусь)

Хитров М.В., кандидат технических наук

Чучупал В.Я., кандидат физико-математических наук

Шелепов В.Ю., доктор физико-математических наук (Украина)

Пилипенко В.В., старший научный сотрудник (Украина)

Сажок Н.Н., кандидат технических наук (Украина)

Кушнир Д.А., кандидат технических наук, ответственный секретарь

Содержание

Кушнир Д.А., Ромашкин Ю.Н.

**Автоматическое реферирование
речевых сообщений 3**

Голенков В.В., Гулякина Н.А.

**Графодинамические модели параллельной
асинхронной обработки знаний 13**

*Петровский А.А., Азаров И.С., Лихачёв Д.С., Ромашкин Ю.Н.,
Жигулёвцев Ю.Н., Харламов А.А.*

**Фильтрация речи на фоне полигармонических
и стохастических помех 45**

<i>Маковкин К.А.</i> Гибридные модели – Скрытые марковские модели/Многослойный персептрон – и их применение в системах распознавания речи. Обзор	58
<i>Шынкевич М.</i> Философия в эпоху компьютеров. Несколько комментариев на тему философского измерения информационных технологий	84
<i>Корсакова Н.С., Засыпкина К.А.</i> Методика тестирования систем автоматического синтеза и распознавания речи в целях определения коммерческой целесообразности их использования	92

Редакция:

Редактор — Елена Долматова
Выпускающий редактор — Анастасия Чипенко
Корректор — Татьяна Денисьева
Дизайн — Анна Ладанюк
Вёрстка — Александр Перевозов

Адрес редакции: 109341, Москва, ул. Люблинская, д. 157, корп. 2
Тел.: 8 (495) 979-54-27

Подписано в печать 29.11.2012. Формат 60x90/8. Бумага офсетная. Печать офсетная
Печ. л. 12,0. Тираж 1000 экз. Заказ № . Издательский дом «Народное образование»
Отпечатано в ООО «Чебоксарская типография № 1», 428019, г. Чебоксары, пр. И. Яковлева, 15

© «Народное образование»

Автоматическое реферирование речевых сообщений

Кушнир Д.А., кандидат технических наук,



Ромашкин Ю.Н., кандидат технических наук



В статье представлен подход к решению задачи автоматического реферирования речевых сообщений на основе кластерного анализа и n-граммных языковых моделей, разработаны критерии и предложена методика оценки качества автоматических рефератов. Проведены экспериментальные исследования с использованием разных мер информационной значимости термов, сформулированы основные выводы о качестве автоматических рефератов в зависимости от количества ошибок распознавания, длительности реферируемых речевых сообщений и коэффициента сжатия рефератов.

• *распознавание речи* • *автоматическое реферирование речевых сообщений*
• *извлечение полезной информации* • *n-граммные модели языка* • *кластеризация текстовых документов.*

The paper presents an approach to solving the problem of automatic summarization of voice messages based on cluster analysis and n-gram language models, and developed criteria for assessing the quality of the technique of automatic summaries. Experimental studies using different measures of information value terms, sets out the basic conclusions about the quality of automatic abstracts according to the number of recognition errors, length of peer voice and the compression ratio of abstracts.

• *speech recognition* • *automatic speech summarization* • *Information retrieval* • *n-gram language modeling* • *clusterization of text documents.*

Введение

Задача автоматического реферирования в целом заключается в определении тематики документа, выделении семантически наиболее значимых слов, фраз и предложений или синтезе связного текста, отражающего основное содержание документа при уменьшении его объёма. Связный текст — не единственный конечный результат реферирования.

Это может быть список ключевых слов, набор семантических отношений, заполненные поля некоторой структуры данных и др.

Автоматическое реферирование речевых сообщений сводится к реферированию текстов, полученных в результате автоматического распознавания этих сообщений. Оно относится к классу нерешённых в настоящее время задач в связи с отсутствием надёжных систем распознавания слитной речи. Дополнительно возможно привлечение просодической информации в речевом сигнале, что в некоторых случаях позволяет выделять акцентированные фрагменты речи, которые предположительно являются семантически более значимыми.

В данной статье предлагается подход к автоматическому реферированию речевых сообщений, учитывающий ошибки распознавания речи, а также отсутствие автоматической сегментации полученного текста на семантические единицы (предложения и фразы).

Чтобы уменьшить влияние ошибок распознавания, полученный в результате распознавания текст пропускается через n -граммную языковую модель, которая служит своеобразным фильтром текста от слов и словосочетаний, не связанных с общим содержанием. При этом слова и словосочетания, не распознанные в рамках языковой модели, не исключаются из последующего анализа, а отмечаются как малоинформативные. Напротив, распознанные n -граммы слов служат ядром формирования семантических единиц выходного текста реферата.

Реализация такой процедуры предполагает использование большого корпуса текстовых документов, который предварительно подвергается процедуре кластеризации. Для каждого кластера документов строится своя n -граммная модель, а реферируемый текст на первом этапе обработки при помощи кластерного анализа относится к ближайшему кластеру и его n -граммной модели.

Обучение системы реферирования

Структурная схема обучения системы автоматического реферирования речевых сообщений показана на рис. 1.

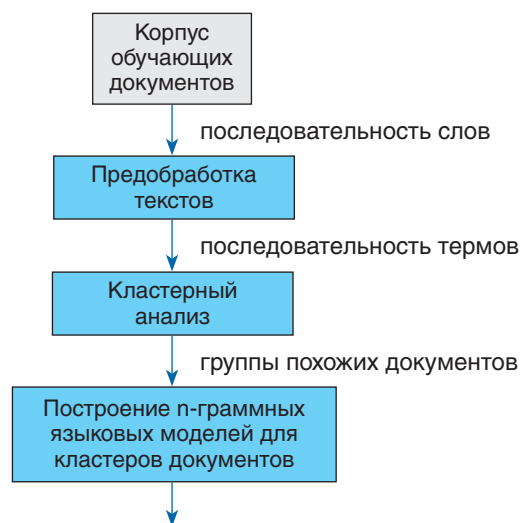


Рис. 1. Структурная схема обучения системы реферирования

Модуль предобработки текстов используется как на этапе обучения, так и на этапе реферирования речевых сообщений. Основная задача этого модуля заключается в преобразовании всех словоформ входного текста в нормальную форму и удалении малозначимых слов и словосочетаний. Лингвистическое обеспечение системы состоит из морфологического словаря, стеммера, словарей стоп-слов и синонимов. Морфологический словарь представлен различными формами слов русского языка со ссылками на их нормальную форму. Он необходим при работе с русским языком со сложной системой флексий. В данной работе словарь насчитывал более 3 млн словоформ. Если слово отсутствует в словаре, то оно поступает на вход стеммера, который осуществляет формальное выделение основы слова. Словарь стоп-слов состоял из 1200 малоинформативных слов и словосочетаний.

Сбор обучающего множества текстов проводился с помощью информационно-поисковой системы. В качестве источников были выбраны интернет-сайты (более 50, включая архивы), среди них www.rian.ru, www.rbc.ru, www.kp.ru, www.aif.ru и др. Общий объем обучающей выборки составил порядка 200 тыс. документов.

Для выполнения кластерного анализа использовалась векторно-пространственная модель представления текстов. В рамках данной модели каждому терму документа сопоставляется неотрицательный вес, и образ документа представляется в виде многомерного вектора. В качестве весов терма использовалась мера TF-IDF [1]. Элементы многомерного вектора нормировались для учёта размера документа путём деления на квадратный корень из суммы весов всех термов. Алгоритм k -средних разбивал множество текстов, представленных в векторном пространстве, на заранее известное число кластеров. В результате было получено 110 кластеров, каждый из которых включал в себя от 1000 до 3000 документов.

Языковые модели, реализованные в данной работе, представляют собой иерархические структуры из динамических ассоциативных запоминающих устройств (ИС ДАЗУ) [2]. Иерархическая структура ДАЗУ — многоуровневая однородная сеть нейроэлементов (НЭ). На первом уровне множество НЭ соответствует множеству отдельных слов в нормальной форме, во втором — биграммам слов, на третьем — триграммам слов и т.д. Таким образом, каждый НЭ соответствует n -грамме слов и однозначно идентифицируется своими координатами (уровнем и номером).

Кроме того, он содержит в себе поля, необходимые для выполнения аналитики: частоту встречаемости каждого НЭ в обучающих примерах, ссылку на НЭ-вершину, индекс текстового документа и признак того, что НЭ является вершинным, т.е. включает в себя целое предложение. Частота встречаемости показывает, насколько часто соответствующая последовательность слов встречалась в обучающих данных. Ссылка на НЭ-вершину содержит в себе механизм доступа ко всем элементам-вершинам, частью которых является данный НЭ. При такой реализации НЭ-вершина соответствует одному предложению текста. Благодаря такой ссылке в ИС ДАЗУ есть возможность быстро восстановить все предложения, в которые входит анализируемый НЭ. Индекс текстового документа используется для организации доступа к оригиналу с целью извлечения из него исходных последовательностей слов. Данная процедура необходима для того, чтобы иметь возможность получить список реализаций НЭ в исходных текстах. Признак вершины используется для выявления n -грамм целых предложений.

Процедура обучения ИС ДАЗУ сводится к запоминанию входных последовательностей термов, поступающих из обучающих текстов, после их предварительной обработки. В результате обучения n -граммная языковая модель позволяет распознавать произвольные последовательности термов в обрабатываемом тексте.

Алгоритм автоматического реферирования

Структурная схема разработанного алгоритма автоматического реферирования речевых сообщений (РС) показана на рис. 2.

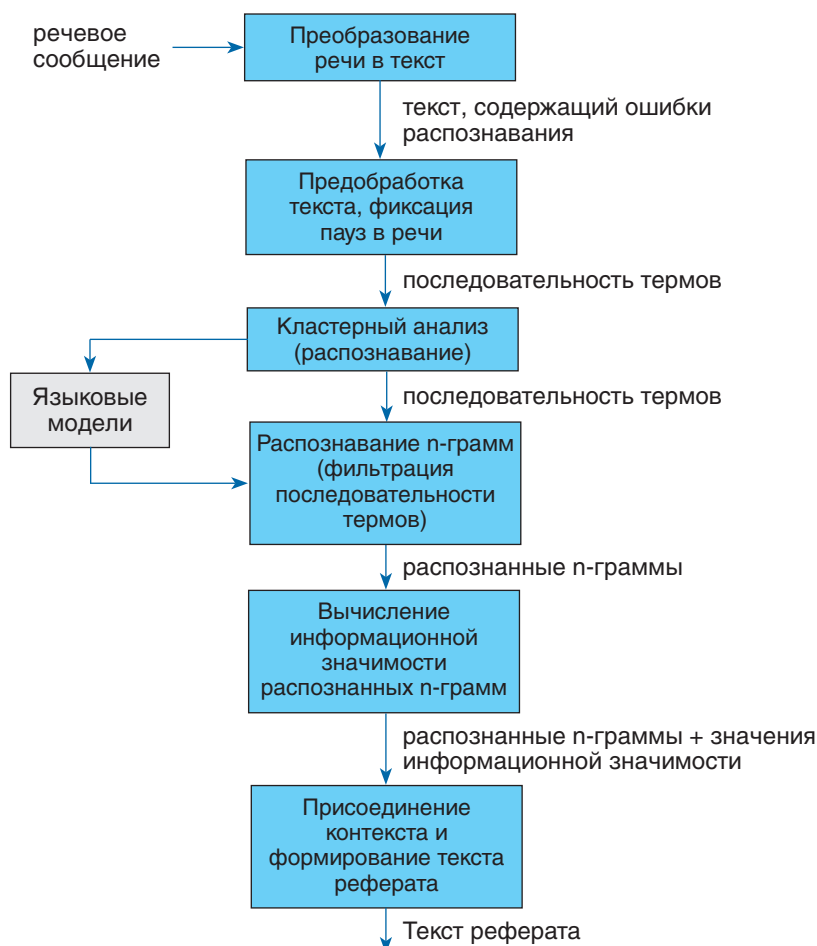


Рис. 2. Схема алгоритма автоматического реферирования РС

Результат автоматического преобразования речи в текст поступает на вход модуля, определяющего идентификатор кластера, которому больше всего соответствует входной текст. По распознанному кластеру загружается соответствующая n-граммная модель. Необходимость данного этапа реферирования обусловлена тем, что результат автоматического распознавания слитной речи содержит много ошибок. Их можно частично устранить при помощи фильтров, настроенных на заданную предметную область реферируемого текста. Роль такого фильтра выполняет ИС ДАЗУ, построенная на множестве текстов, принадлежащих распознанному кластеру. Фильтрация реализуется за счёт распознавания n-грамм в ИС ДАЗУ. Предполагается, что если языковая модель действительно соответствует входному речевому сообщению, то будут распознаны только те слова и словосочетания, которые уместны в рамках данной модели, остальные будут не распознаны и отмечены как неинформативные на этом этапе.

Одна из особенностей задачи реферирования речевых сообщений заключается в том, что в распознанном тексте недостаточно информации для его точной сегментации на такие семантические единицы, как фразы и предложения. Обучение ИС ДАЗУ происходит на предложениях. По этой причине распознанные n-граммы, как правило, являются частью одного предложения.

Для каждой распознанной n -граммы вычисляется значение информационной важности, которое складывается из значений информационной значимости входящих в неё отдельных термов. В исследованиях проверялись три способа вычисления информационной значимости термов.

Первый способ соответствовал выражению из [3, 4]:

$$I_i^{(1)} = f_i \cdot \log \frac{F_\Sigma}{F_i},$$

где f_i и F_i — частоты встречаемости определённого слова во входном тексте и в корпусе обучающих текстов соответственно, F_Σ — общее количество слов в корпусе текстов.

При втором способе вычислялось произведение

$$I_i^{(2)} = TF_i \cdot IDF_i = \frac{n_i}{\sum_k n_k} \cdot \log \frac{|D|}{|d_i \supset t_i|},$$

где TF — отношение числа вхождений i -го терма к общему количеству слов в тексте, IDF — обратная частота документа, определяемая как логарифм отношения общего количества документов в кластере к количеству документов кластера, в которых встречается терм t_i .

Третий способ (TF-IDF-LEN) основан на мере TF-IDF с добавлением длины терма в символах (LEN) [5]:

$$I_i^{(3)} = TF_i \cdot IDF_i \cdot LEN.$$

Идея его базируется на предположении, что часто встречающиеся слова стремятся быть краткими, т.е. являются стоп-словами.

Многие термы реферируемого текста не несут существенной информационной нагрузки, но они необходимы в качестве контекста к информационно значимым n -граммам для повышения связности изложения. Однако определить автоматически, где начинается и заканчивается нужный контекст информационно значимой n -граммы, задача нетривиальная.

Для её решения был разработан критерий *равномерного распределения информационной важности* по реферату. Идея предлагаемого подхода состоит в следующем: значение информационной значимости n -граммы пропорционально «силе притяжения» контекста, т.е. чем важнее распознанная n -грамма, тем более широким контекстом целесообразно её снабдить при составлении текста реферата. При этом в процессе расширения контекста происходит постоянный пересчёт общей информационной важности формируемой семантической единицы. Как только она достигает некоторого среднего значения, наращивание контекста текущей n -граммы прекращается и происходит переход к следующей. Все n -граммы обрабатываются в порядке уменьшения их информационной важности.

Процедуру выравнивания информационной важности (путём присоединения контекста к информационно значимым n -граммам) можно представить в такой последовательности:

- 1) все распознанные n -граммы ранжируются в порядке уменьшения их информационной значимости (ИЗ);
- 2) по заданному объёму реферата определяется количество n -грамм, которые должны попасть в реферат из ранжированного списка, начиная с самой важной;



- 3) для полученного в п.2 подсписка n-грамм вычисляется среднее значение (\bar{I}) ИЗ;
- 4) задаётся желаемое значение ИЗ фрагментов исходного текста, из которых будет складываться реферат: $I_0 = \bar{I} / k$, где коэффициент k определяет значимость контекста и варьируется от 1 до 2;
- 5) к каждой n-грамме присоединяются их левые и правые контексты в порядке уменьшения ИЗ до тех пор, пока для полученного фрагмента текста ИЗ не достигнет желаемого значения I_0 .

Методики оценки качества

В настоящее время применительно к обработке текстовых документов разработан ряд методик оценки качества алгоритмов их автоматического реферирования [6–10]. Однако к задаче реферирования речевых сообщений, когда отдельные слова распознаются неверно и информация о границах предложений отсутствует, они не подходят.

Качество реферирования речевых сообщений зависит от точности работы модуля автоматического распознавания речи и эффективности модуля автоматического реферирования результата распознавания. Поскольку в данной работе функционирование этих модулей осуществлялось независимо друг от друга, желательно провести оценку качества как для системы в целом, так и для модуля реферирования отдельно. При этом целесообразно использовать два способа оценки качества алгоритмов реферирования:

- 1) объективная оценка на основе вычисления некоторой меры подобия рефератов, подготовленных экспертами и полученных автоматически;
- 2) субъективная оценка с помощью метода экспертных оценок качества автоматических рефератов.

Учитывая указанную выше специфику реферируемого текста, подобие рефератов характеризовалось размером области пересечения множеств лексических единиц (n-грамм слов) рефератов, составленного экспертом и полученного автоматически.

Процедура получения объективной оценки качества реферирования состоит из пяти шагов:

1. Отбирается заданное количество тестовых речевых сообщений с представительным разбросом по точности автоматического преобразования в текст (значения ошибки правильного распознавания слов от 0.1 до 0.5).
2. Рефераты составляются экспертами путём анализа результата автоматического распознавания речи и последующего выбора n-грамм слов, которые наиболее точно характеризуют семантику каждого речевого сообщения (для заданных коэффициентов сжатия текста в 2–10 раз с шагом 2).
3. Автоматическое формирование рефератов в соответствии с разработанным алгоритмом и с заданными коэффициентами сжатия.
4. Сравнение составленных рефератов по полноте и точности в соответствии со следующими выражениями:

$$\text{Полнота} = R^{(n)} = \frac{\{N_{\text{Э}}^{(n)}\} \cap \{N_A^{(n)}\}}{N_{\text{Э}}^{(n)}}$$

$$\text{Точность} = P^{(n)} = \frac{\{N_{\text{Э}}^{(n)}\} \cap \{N_A^{(n)}\}}{N_A^{(n)}}$$



где $N_{Э}^{(n)}$ — количество n -грамм слов в реферате, подготовленном экспертом;
 $N_A^{(n)}$ — количество n -грамм слов в реферате, полученном автоматически.

Т.е. полнота равна отношению числа n -грамм автоматического реферата, совпавших с n -граммами реферата эксперта, к общему количеству n -грамм в реферате эксперта. Точность равна отношению числа n -грамм автоматического реферата, совпавших с n -граммами реферата эксперта, к общему количеству n -грамм в автоматическом реферате. Для проверки совпадения выбирались униграммы и биграммы слов (т.е. $n = 1, 2$).

Для получения одной оценки качества автоматического реферата использовалась F-мера — гармоническое среднее полноты и точности:

$$F = \frac{1}{\alpha \cdot \frac{1}{P} + (1 - \alpha) \cdot \frac{1}{R}}, \text{ где } \alpha = 0,5$$

5. Усреднение полученных оценок качества по всем тестовым речевым сообщениям (отдельно по точности автоматического распознавания речи и заданному коэффициенту сжатия текста).

При субъективной оценке качества автоматических рефератов группа из пяти экспертов выставляла баллы в соответствии со шкалой качества (таблица 1). Затем по среднему значению проставленных баллов определялся класс качества реферата для заданного коэффициента сжатия текста.

Таблица 1

Класс качества	Характеристика класса	Балльная оценка
Высший	В реферате отражены все основные факты и упоминаются все личности из речевого сообщения (при данном объеме реферата)	$\geq 4,5$
I	По реферату можно определить суть речевого сообщения. Он содержит несколько (более одного) важных фактов, включая основной, или упоминание о ключевой личности	от 3,5 до 4,4
II	По реферату можно определить общий смысл речевого сообщения. Реферат содержит один важный, не обязательно основной, факт или упоминание об одной, не обязательно ключевой, личности	от 2,5 до 3,4
III	По реферату понятна только тема речевого сообщения. Суть сообщения определить невозможно. Реферат не содержит ни одного важного факта или упоминания о какой-либо личности	от 1,5 до 2,4
Срыв реферирования	По реферату невозможно определить даже тему речевого сообщения	$< 1,5$

Результаты экспериментов

В результате работы группы из пяти экспертов была получена база рефератов со следующими характеристиками:

- общее количество обработанных речевых сообщений на русском языке из новостных передач радиовещательных станций = 176;
- количество рефератов на каждое сообщение = 5 (с коэффициентами сжатия текста 2, 4, 6, 8 и 10);
- общее количество рефератов на одного эксперта: = 880;
- ошибка алгоритма автоматического распознавания слов (WER) в слитной речи (дикторской или спонтанной) от 0,1 до 0,5;
- распределение числа реферируемых текстов в зависимости от WER приведено в таблице 2.

Таблица 2

WER	0.1	0.2	0.3	0.4	0.5
Количество реферируемых текстов	6	27	46	59	38

С целью определения лучшего способа вычисления информационной значимости термов и значения коэффициента значимости контекста были получены объективные оценки качества рефератов для различных комбинаций этих параметров (для каждого варианта создавалось 880 автоматических рефератов). Анализ этих оценок показал, что наилучшим способом вычисления информационной значимости n-грамм слов, включаемых в автоматический реферат, является TF-IDF-LEN с коэффициентом значимости контекста равным 2.

Сбор экспертных оценок качества автоматического реферирования проводился с помощью специальной программы, вид главного окна которой показан на рис. 3. Справа в окне отображается шкала допустимых балльных оценок (шаг оценивания составляет 0,1) и качественное описание каждого из 5 возможных классов. В левой верхней части окна отображается полный текст речевого сообщения (без ошибок распознавания), а ниже — автоматически сформированный реферат для этого сообщения с заданным коэффициентом сжатия.

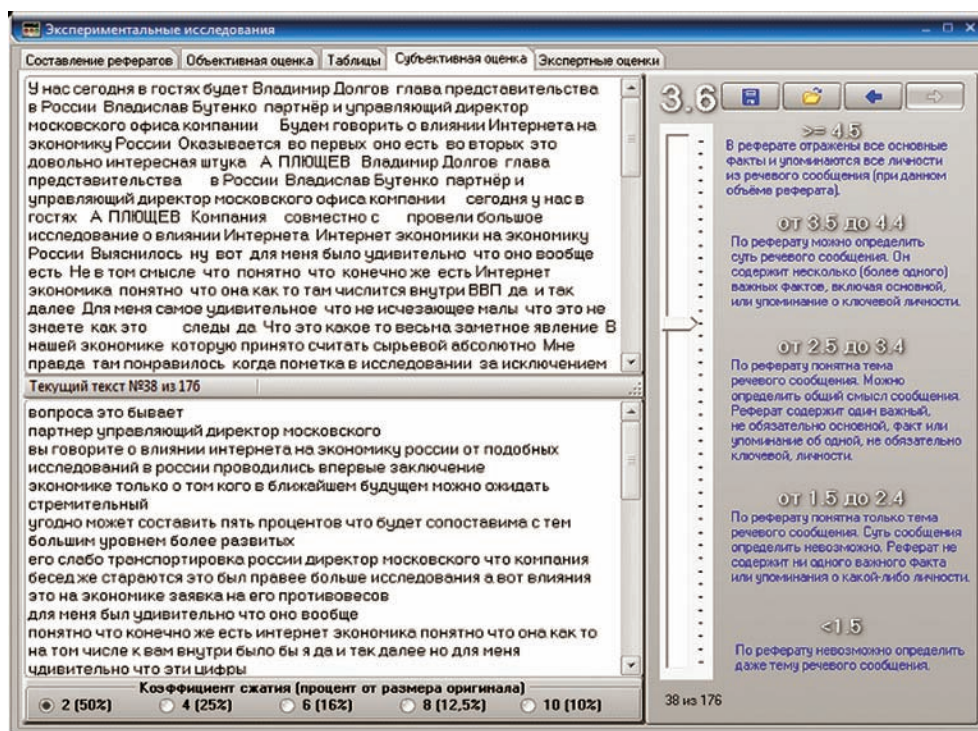


Рис. 3. Внешний вид интерфейса программы для экспертных оценок класса качества реферирования

Средние значения балльных оценок, выставленные всеми экспертами, и соответствующие им номера классов качества автоматического реферирования в зависимости от величины ошибки распознавания речи и коэффициента сжатия текста приведены в таблице 3, а в зависимости от длительности (Т) реферируемых речевых сообщений — в таблице 4.

Они позволяют сделать выводы:

- качество рефератов слабо зависит от вероятности ошибок распознавания слов в слитной речи (в пределах от 0.1 до 0.5), оставаясь преимущественно в рамках одного класса;
- качество рефератов монотонно возрастает с увеличением длительности речевого сообщения. В основном, это связано с используемой мерой информационной значимости, в которой частота встречаемости термина пропорциональна его значимости;
- для коротких речевых сообщений (длительностью менее 1 мин.) при коэффициенте сжатия 10 наблюдается срыв реферирования.

Таблица 3

WER	Балльные оценки / классы качества автоматического реферирования				
	Коэффициент сжатия				
	2	4	6	8	10
0,1	3,3/ II	2,6/ II	2,0/ III	1,8/ III	1,7/ III
0,2	3,6/ I	3,0/ II	2,3/ III	2,1/ III	1,8/ III
0,3	3,9/ I	3,3/ II	2,7/ II	2,5/ III	2,2/ III
0,4	3,5/ II	3,0/ II	2,5/ II	2,2/ III	2,0/ III
0,5	3,2/ II	2,7/ II	2,3/ III	2,1/ III	1,8/ III

Таблица 4

T, с	Балльные оценки / классы качества автоматического реферирования				
	Коэффициент сжатия				
	2	4	6	8	10
< 45	3,1/ II	2,4/ III	1,8/ III	1,6/ III	1,1/ C
45–60	3,2/ II	2,5/ II	1,8/ III	1,6/ III	1,3/ C
60–120	3,3/ II	2,7/ II	2,0/ III	1,8/ III	1,6/ III
121–180	3,4/ II	2,8/ II	2,3/ III	2,0/ III	1,7/ III
181–240	3,5/ I	2,9/ II	2,4/ III	2,2/ III	2,0/ III
> 240	3,6/ I	3,1/ II	2,7/ II	2,4/ III	2,1/ III

Заключение

Разработан метод автоматического реферирования речевых сообщений на русском языке, обеспечивающий максимальное согласование экспертных и автоматических рефератов по выбранным мерам (полнота и точность для униграмм и биграмм). Предложены методики оценки качества автоматического реферирования, использующие расчётные (объективные) и экспертные (субъективные) количественные показатели.

С использованием тестовых записей речевых сообщений новостных программ радиовещания на русском языке получены оценки качества их автоматического реферирования для различных длительностей (T) сообщений и коэффициентов (K) сжатия текста. Экспериментально установлено, что при T=1–3 мин. разработанный алгоритм способен обеспечивать I–III классы качества автоматического реферирования. При T<1 мин. и K=10 происходит срыв автоматического реферирования.

С увеличением ошибки алгоритма распознавания слов в речи от 0.1 до 0.5 класс качества рефератов преимущественно не изменяется.



Список литературы

1. Debole F. and Sebastiani F. Supervised term weighting for automated text categorization. In the Proceedings of SAC-03, 18th ACM Symposium on Applied Computing, Melbourne, US: ACM Press, New York, US, 2003. Pp. 784–788.
2. Харламов А.А. Нейроподобные элементы с временной суммацией входного сигнала и блоки ассоциативной памяти на основе этих элементов // Вопросы кибернетики. Устройства и системы. М.: МИРЭА, 1983. С. 57–68.
3. Kikuchi T., Furui S., Hori C. Automatic Speech Summarization Based on Sentence Extraction and Compaction // ICASSP 2003. Pp. 384–387.
4. Furui S. Recent Advances in Automatic Speech Summarization // Conference RIAO2007, Pittsburgh PA, U.S.A. May 30-June 1, 2007.
5. Luhn H. The automatic creation of literature abstracts. In IBM Journal of Research and Development, Vol. 2(2), 1958. Pp. 159–165.
6. Salton G. et al. Automatic Text Structuring and Summarization // Information Processing & Management, Vol. 33, No. 2, 1997. Pp. 193–207.
7. Radev D. and Tam D. Single-document and multidocument summary evaluation via relative utility // Poster session, CIKM'03, 2003.
8. Lin C.-Y. ROUGE: A Package for Automatic Evaluation of Summaries. Information Sciences Institute. University of Southern California 2004.
9. Lin C.-Y. ROUGE: A package for automatic evaluation of summaries // Proc. of the Workshop on Text Summarization Branches Out (ACL'2004). Barcelona, Spain, 2004. Pp. 74–81.
10. Тарасов С.Д. Исследование и оптимизация параметров алгоритма Manifold Ranking на основе метрики автоматической оценки качества обзорного реферирования ROUGE-RUS // Труды 11-й Всерос. науч. конф. «Электронные библиотеки: перспективные методы и технологии, электронные коллекции» RCDL'2009. Петрозаводск, Россия, 2009. С. 86–93.

Сведения об авторах

Кушнир Д.А. —

кандидат технических наук, ведущий научный сотрудник
ЗАО «НТЦ «Поиск-ИТ».
kushnir@speechtechnology.ru

Ромашкин Ю.Н. —

кандидат технических наук.



Графодинамические модели параллельной асинхронной обработки знаний (продолжение)

Голенков В.В., доктор технических наук, профессор,

Гулякина Н.А., кандидат физико-математических наук, доцент

В статье рассматриваются принципы построения технологии проектирования интеллектуальных систем, ориентированных на семантическое представление знаний, расширение контингента разработчиков и сокращение сроков проектирования.

• графодинамическая модель обработки информации • графодинамическая память • многоагентная система над графодинамической памятью • графовый язык программирования • семантическая сеть • бинарная семантическая сеть • язык визуализации семантических сетей • унификация семантических сетей • интеграция семантических сетей • семантическая технология проектирования интеллектуальных систем • семантическая технология проектирования баз знаний • семантическая технология проектирования решателей задач • семантическая технология проектирования пользовательских интерфейсов • библиотека семантически совместимых многократно используемых компонентов интеллектуальных систем.

The article presents the principles of construction technology of designing intelligent systems which are oriented on semantic representation of knowledge, expansion of the number of developers and shortening time of design are considered.

• graph-dynamic information processing model • graph-dynamic memory multiagent system over graph-dynamic memory • graph programming language • semantic network • binary semantic network • semantic network visual language, unification of semantic network • semantic network integration • semantic network merging • semantic network fusion • semantic intelligent system design technology • semantic knowledge base design technology • semantic problem solver design technology • semantic user interface design technology • library of reusable semantically compatible components for intelligent systems.

Принцип 5. Унификация семантического представления знаний

Определение структуры унифицированных семантических сетей, обеспечивающих представление различных видов знаний, предполагает разработку соответствующего стандарта, выделяющего из всего многообразия абстрактных языков семантических сетей определённый базовый универсальный язык, который мы назвали SC-кодом (Semantic Computer code).

Для того чтобы перейти от семантических сетей произвольного вида к текстам SC-кода, уточним критерии качества SC-кода, к которым относится, во-первых, переход от семантических сетей, имеющих унарные и многокомпонентные (многоместные) связки, к семантическим сетям, имеющим только бинарные связки. Такие сети будем называть бинарными семантическими сетями; во-вторых, минимизация алфавита, т.е. минимизация числа меток, используемых в семантических сетях; в-третьих, универсальность разрабатываемого языка, т.е. возможность представления любых знаний в виде текстов этого языка.

Семантическую сеть G_b , задаваемую пятёркой $\langle V_b, C_b, I_b, M_b, K_b \rangle$, будем называть **бинарной семантической сетью** только в том случае, если:

- все связки, входящие во множество C_b , являются бинарными (каждая связка имеет только два компонента: это могут быть элементы множеств V_b, C_b и K_b);
- $I_b = \{i_1, i_2\}$, где i_1 — бинарное ориентированное отношение инцидентности, связывающее бинарную связку с её компонентом; i_2 — бинарное ориентированное отношение инцидентности, связывающее бинарную ориентированную связку с её вторым компонентом. Очевидно, что $i_1 \subset i_2$.

Для любой семантической сети G , задаваемой пятёркой $\langle V, C, I, M, K \rangle$, можно построить семантически эквивалентную ей бинарную семантическую сеть G_b , задаваемую пятёркой $\langle V_b, C_b, I_b, M_b, K_b \rangle$, следующим образом:

- во множество V_b включаются все вершины (V) семантической сети G и все небинарные связки (C_n) семантической сети G ($C_n \subset G$);
- $I_b = \{i_1, i_2\}$ — это отношения инцидентности, связывающие бинарные ориентированные связки (из множества C) с их компонентами;
- $K_b = K \cup (I_b)$. Т.е. знаки отношений инцидентности, связывающих (в рамках G) небинарные связки с их компонентами, становятся в рамках G_b ключевыми узлами;
- $M_b = M$. При этом, если в алфавит M не входила метка отношения принадлежности (re), то она дополнительно вводится в алфавит M_b ;
- во множество C_b включаются все бинарные связки (C_b) семантической сети G ($C_b \subset C$); все ориентированные пары всех отношений инцидентности (I), связывающих небинарные ориентированные связки (из множества C) с их компонентами; все пары принадлежности, которые явно связывают ключевые узлы семантической сети G_b , соответствующие различным отношениям инцидентности (I) семантической сети G , с парами принадлежности, которые явно связывают небинарные связки с их компонентами и которые в рамках этих связок выполняют роли, обозначаемые соответствующими отношениями инцидентности.

Нетрудно заметить, что приведение семантической сети к бинарному виду приводит к минимизации числа отношений инцидентности. Все пары инцидентности, связывающие небинарные связки семантической сети G с их компонентами, в бинарной семантической сети C_b «превращаются» в бинарные ориентированные связки, которые принадлежат отношению принадлежности и, следовательно, должны быть помечены знаком отношения принадлежности re ($re \in M_b$).

Таким образом, каждая небинарная связка семантической сети G в семантической сети G_b трактуется как множество связываемых им элементов семантической сети, связь которого с его элементами представляется в виде дополнительно вводимых связок принадлежности. Эти связки принадлежности, в свою очередь, могут быть вторыми компонентами других связок

принадлежности, связывающих указанные связки с ключевыми узлами, обозначающими различные роли компонентов небинарных ориентированных связок исходной семантической сети.

Бинарные связки легко изображать графически (в виде линий, каждая из которых соединяет графические изображения двух связываемых элементов семантической сети). Использование только бинарных связок существенно упрощает машинное кодирование семантических сетей и, в частности, упрощает разработку специальной памяти для их хранения сетей.

В основе **минимизации алфавита** элементов лежит следующее свойство семантических сетей. Метки, входящие в состав алфавита элементов семантической сети, и ключевые узлы этой сети семантически эквивалентны, т.е. являются синтаксически различными способами выделения (задания) различных классов элементов семантической сети. При этом, в отличие от алфавита символов линейного текста, все элементы алфавита (все метки) семантической сети, как и все её ключевые узлы, имеют семантическую интерпретацию на описываемой предметной области — все они являются знаками различных классов элементов этой предметной области. Таким образом, метки элементов семантической сети без какого-либо изменения её семантики можно преобразовывать в ключевые узлы. При этом семантическая интерпретация каждого ключевого узла будет совпадать с семантикой соответствующей преобразованной метки.

Если мы имеем дело с корректно (правильно) построенной семантической сетью и записанной на некотором языке семантических сетей, то её перевод на любой другой язык семантических сетей не требует больших усилий, так как синтаксическое и семантическое устройства всех языков семантических сетей сходны. Фактически эти языки отличаются алфавитами, элементы которых легко преобразуются в ключевые узлы семантических сетей.

Итак, число меток семантической сети можно уменьшать ценой расширения множества её ключевых узлов. Вопрос в том, до какого предела это можно делать и как выглядит минимальный алфавит универсального языка семантических сетей. Универсальный язык семантических сетей — такое множество семантических сетей, элементами которых являются семантические сети, представляющие любую информацию о любой описываемой предметной области.

Из вышесказанного следует, что в состав минимального алфавита универсального языка семантических сетей, по крайней мере, должна входить метка *re*, обозначающая **отношение принадлежности**. Без этой метки невозможно описать связи ключевых узлов семантической сети с элементами обозначаемых ими классов объектов, а также невозможно осуществить переход к бинарным семантическим сетям.

Текст, принадлежащий SC-коду, т.е. **sc-текст** (sc-структура, sc-конструкция), является семантической сетью частного вида, имеющей следующие особенности:

— все связки sc-текстов — бинарные связки — **sc-коннекторы**. Неориентированные sc-коннекторы — **sc-ребра**, ориентированные — **sc-дуги**.

Множество меток элементов sc-текстов (алфавит sc-элементов, алфавит SC-кода) включает в себя метки:

- **sc-узлов** (вершин sc-текстов);
- **sc-ребер**;
- **sc-дуг общего вида**;
- **sc-дуг принадлежности**;
- **sc-дуг основного вида**.

Множество отношений инцидентности элементов sc-текстов состоит из двух следующих отношений:

- **быть компонентом sc-коннектора** (sc-ребра или sc-дуги);

— **быть вторым компонентом sc-дуги;**

— множество ключевых узлов sc-кода (ключевых узлов sc-текстов) вместе с метками sc-элементов задаёт базовую семантическую типологию sc-элементов, т.е. базовую онтологию SC-кода.

К числу ключевых узлов SC-кода, определяющих разбиение множества sc-элементов по признаку константности, относятся:

— ключевой sc-узел, обозначающий множество всевозможных константных sc-элементов (**sc-констант**). Каждая sc-константа — обозначение некоторого конкретного фиксированного объекта;

— ключевой sc-узел, обозначающий множество всевозможных переменных sc-элементов (**sc-переменных**), каждая из которых обозначает некоторый произвольный, нефиксированный объект из некоторого дополнительно уточняемого множества объектов. Используются sc-переменные в логических формулах (в т.ч. в высказываниях), в программах (в обобщённых описаниях способов решения различных классов задач), в формулировках вопросов.

К числу ключевых узлов SC-кода, определяющих разбиение множества sc-элементов по структурному признаку, относятся:

— ключевой sc-узел, обозначающий множество всевозможных **sc-коннекторов**. Более детальное разбиение множества sc-коннекторов по структурному признаку осуществляется с помощью меток sc-элементов (на **sc-ребра**, **sc-дуги общего вида**, **sc-дуги принадлежности**, **sc-дуги основного вида**);

— ключевой sc-узел, обозначающий множество всевозможных sc-узлов, каждый из которых обозначает некоторое связующее множество sc-элементов. Указанные sc-элементы — **связующие sc-узлы**. К числу ключевых узлов SC-кода, определяющих более детальную структурную типологию связующих sc-узлов, относятся:

— ключевой sc-узел, обозначающий множество всевозможных sc-узлов, каждый из которых обозначает некоторую неатомарную связь между sc-элементами, т.е. связь, не являющуюся sc-коннектором. Такие sc-узлы будем называть **неатомарными sc-связками**. Более детальная структурная типология неатомарных sc-связок задаётся такими ключевыми узлами SC-кода, как быть **унарной sc-связкой**, быть **бинарной неатомарной sc-связкой**, быть **многокомпонентной sc-связкой**, быть **ориентированной неатомарной sc-связкой**, быть **неориентированной неатомарной sc-связкой**;

— ключевой sc-узел, обозначающий множество всевозможных sc-узлов, каждый из которых обозначает некоторую структуру из sc-элементов — **sc-структуру**;

— ключевой sc-узел, обозначающий множество всевозможных sc-узлов, каждый из которых обозначает некоторый класс sc-элементов — **sc-понятия**. Более детальная структурная типология sc-понятий задаётся следующими ключевыми узлами SC-кода: быть **отношением** (классом однотипных связей), быть **бинарным отношением**, быть **унарным отношением**, быть **многоместным отношением**, быть **ориентированным отношением**, быть **неориентированным отношением**, быть **ролевым отношением** (т.е. отношением, которое является подмножеством отношения принадлежности), быть **классом структур**, быть **классом терминальных sc-узлов** (первичных sc-узлов, которые не являются обозначениями множеств sc-элементов);

— ключевой sc-узел, обозначающий множество всевозможных sc-узлов, каждый из которых обозначает некоторый объект, который не является множеством sc-элементов. Такие sc-узлы будем называть **терминальными sc-**

узлами (первичными sc-узлами). Более детальная структурная типология терминальных sc-узлов задаётся следующими ключевыми узлами sc-кода:

- быть **предметным sc-узлом**, каждый из которых обозначает некоторый реальный (материальный, физический) или вымышленный внешний объект некоторой предметной области;
- быть **sc-ссылкой**, каждая из которых обозначает либо определённый файл, который можно просматривать или в котором закодирована в определённом формате некоторая внешняя, инородная для sc-кода информационная конструкция, либо некоторую компьютерную систему, с которой можно взаимодействовать;
- быть **терминальным элементом шкалы или шаблона** (sc-элементы, для которых трудно установить обозначаемые ими объекты, поскольку эти sc-узлы просто являются терминальными элементами каких-либо шкал, шаблонов, типовых структур, с которыми устанавливаются соответствия, сравниваются, сопоставляются различные объекты и структуры).

К числу ключевых узлов SC-кода, уточняющих семантику sc-дуг принадлежности, относятся:

- **ключевые узлы SC-кода, определяющие разбиение множества sc-дуг принадлежности по признаку позитивности:**
 - быть **sc-дугой позитивной принадлежности**;
 - быть **sc-дугой негативной принадлежности**;
 - быть **sc-дугой нечёткой принадлежности** (т.е. sc-дугой позитивность или негативность которой в текущий момент не установлена);
- **ключевые узлы SC-кода, определяющие разбиение множества sc-дуг принадлежности по признаку стационарности:**
 - быть **sc-дугой стационарной принадлежности**, семантический тип которой является постоянным, не изменяющимся во времени;
 - быть **sc-дугой нестационарной принадлежности**, семантический тип которой изменяется во времени.

Заметим, что **sc-дуги основного вида**, которые выделяются с помощью соответствующей метки, семантически трактуются как **sc-дуги позитивной стационарной принадлежности**.

Особенности и достоинства SC-кода

Унифицированные семантические сети (sc-тексты) — абстрактная семантическая модель знаний — *инвариант* различных способов представления и кодирования этих же знаний (в т.ч. и самих семантических сетей). Наличие такого инварианта необходимо для решения проблемы интеграции различных видов знаний. На основе унифицированных семантических сетей можно строить семантические модели различных компьютерных систем и решить проблему их интеграции.

SC-код — *абстрактный язык*: способ изображения (материализации) его текстов не уточняется. Следовательно, можно разрабатывать различные графические уточнения SC-кода (например, SCg-код), различные варианты изображения sc-текстов в виде строк символов (например, SCs-код) и машинного представления sc-конструкций в адресной памяти традиционных компьютеров, а также в специальной структурно перестраиваемой ассоциативной памяти будущих компьютеров, ориентированных на обработку баз знаний.

Все sc-элементы, кроме терминальных sc-узлов, являются обозначениями множеств, состоящих из sc-элементов — **вторичных sc-элементов**. Из этого следует то, что SC-код имеет базовую теоретико-множественную семантическую интерпретацию.

SC-код — достаточно простой компьютерный код семантических сетей, который является не «инородным» представлением семантических сетей, а их представлением в виде се-

мантических сетей, но максимально простого вида — с минимальным алфавитом и бинарными связками.

SC-код ориентирован на представление информации в компьютерной памяти и может рассматриваться как основа модели структурно перестраиваемой ассоциативной памяти будущих компьютеров, ориентированных на обработку семантических сетей. Т.е. SC-код можно рассматривать как универсальную основу машинного кодирования знаний в памяти будущих компьютеров, ориентированных на обработку семантических сетей. В такой памяти биты и байты «уступят место» sc-дугам, sc-ребрам и sc-узлам.

На базе SC-кода можно создавать целое семейство совместимых специализированных языков, ориентированных на представление разных видов знаний (логических формул и высказываний, программ, вопросов, поведенческих целей, различных видов моделей динамических систем и т.п.) таким образом, чтобы тексты всех этих специализированных языков полностью соответствовали SC-коду, т.е. были sc-текстами. Такие специализированные языки называют **sc-языками**. Каждый sc-язык определяется своим расширением множества ключевых узлов SC-кода.

SC-код — ядро **универсального открытого языка семантических сетей**, являющегося результатом интеграции всевозможных языков семантических сетей, построенных на основе SC-кода, и задаваемого:
— фиксированным алфавитом (алфавитом SC-кода);
— постоянно расширяемым (открытым) семейством ключевых узлов, в состав которого входят все ключевые узлы всех интегрируемых языков.

SC-код — **единство языка и метаязыка**. Так, например, в виде sc-конструкций можно описать синтаксис, семантику и онтологию SC-кода. С формальной точки зрения, SC-код можно трактовать как метаязык базовой семантической спецификации sc-элементов с помощью специального набора ключевых узлов SC-кода.

Единство языка и метаязыка в SC-коде проявляется и на самом низком уровне — на уровне sc-дуг принадлежности $\langle si, ei \rangle$, в которых сам sc-элемент ei , а не обозначаемый им объект, является элементом множества, обозначаемого sc-узлом si .

SC-код позволяет описать структуру любой информационной конструкции, не принадлежащей SC-коду, на любом уровне (на любом этапе синтаксического и семантического анализа). В частности, первичную синтаксическую структуру любой информационной конструкции можно представить в виде изоморфной sc-конструкции. Следовательно, SC-код может быть использован в качестве метаязыка для описания любого внешнего языка, т.е. языка, тексты которого не являются sc-конструкциями.

SC-код хорошо приспособлен к использованию в условиях **не-факторов** — нестационарности, неточности, противоречивости, неактуальности знаний, а также неполноты знаний (нечёткости, несформированности множеств, несформированности внешних информационных конструкций) [49].

Информационные конструкции SC-кода (sc-конструкции) легко **визуализируются**.

Принцип 6. Унификация структуризации баз знаний

Трактовка семантической структуры **базы знаний** интеллектуальной системы как отражения иерархической системы взаимосвязанных друг с другом

предметных областей, представляемых в базе знаний, предполагает, во-первых, уточнение понятия предметной области; во-вторых, разработку языковых средств описания структуры предметных областей с помощью унифицированных семантических сетей; в-третьих, разработку языковых средств описания типологии предметных областей и различных видов связей между ними.

Структуризация базы знаний, выделение в ней различных связанных между собой подструктур необходимы по целому ряду причин, в частности, для дидактических целей (человеку, усваивающему некоторые знания, желательно иметь своего рода оглавление или «карту» этих знаний, что позволяет планировать их усвоение и рассматривать их с различной степенью детализации), а также для организации распределения работ по проектированию баз знаний (разным исполнителям поручается разработка разных фрагментов базы знаний, имеющих достаточно чёткие границы).

Таким образом, база знаний рассматривается как система взаимосвязанных между собой интегрируемых структур — **фрагментов базы знаний**. Связи между ними могут быть самыми различными. Каждый фрагмент и вся база знаний в целом могут иметь несколько вариантов декомпозиции на подфрагменты (частные фрагменты).

По структурно-семантическому принципу можно выделить следующие типы фрагментов баз знаний:

- база фактов некоторой предметной области, которая представляет собой результат интеграции всех известных в текущий момент фактографических высказываний, являющихся истинными для указанной предметной области;
- **иерархическая система нескольких предметных областей**, которые нецелесообразно объединять (интегрировать) в одну предметную область;
- **семантическая окрестность заданного объекта**;
- **связная семантическая окрестность конечного множества заданных объектов**, представляющая сравнение (сравнительный анализ) и связи объектов из заданного конечного множества, т.е. описывающая сходства (аналогии), отличия заданных объектов, а также «близкие» связи между ними.

При структуризации базы знаний некоторым её фрагментам приписывается статус **разделов базы знаний**, которые именуются, нумеруются и входят в её состав (оглавления).

Рассмотрим формальное уточнение понятия «**предметная область**» с помощью SC-кода. Если в рамках **sc-модели базы знаний** явно вводится некоторая предметная область, то она трактуется как некоторая sc-структура, для которой в базе знаний явно вводится обозначающий её sc-узел, который, в свою очередь, связывается входящей в него sc-дугой основного вида с sc-узлом, обозначающим класс sc-структур, являющихся предметными областями. После этого в указанной sc-структуре необходимо явно задать роли некоторых *ключевых узлов*, входящих в состав этой sc-структуры. К числу таких ролей относятся:

- быть **максимальным классом исследуемых объектов**, т.е. множеством всех исследуемых объектов и только их. В каждой предметной области существует только один ключевой узел, выполняющий такую роль;
- быть **классом исследуемых объектов**. Каждая предметная область может иметь любое число таких классов;
- быть **классом вторичных объектов, построенных на основе исследуемых**;
- быть **классом вспомогательных объектов**, через связи с которыми описываются некоторые характеристики исследуемых объектов;
- быть **отношением, каждая связка которого связывает только исследуемые объекты** или вторичные объекты, построенные на основе исследуемых;
- быть **отношением, каждая связка которого связывает исследуемые объекты со вспомогательными**.

Такое явное указание ролей ключевых элементов предметных областей есть не что иное, как их семантическая спецификация, уточняющая то, *какие объекты* описываются (иссле-

дуются) в данной предметной области, и о *каких характеристиках*, связях исследуемых объектов в данной предметной области идёт речь.

Можно говорить о достаточно богатой типологии предметных областей. В частности, можно выделить следующие классы предметных областей:

- предметная область, описывающая теоретико-множественные характеристики связи заданного семейства объектов. Такие предметные области, в частности, могут быть онтологиями других предметных областей;
- терминологическая сеть заданного фрагмента базы знаний;
- текст формальной теории, описывающей свойства и закономерности заданной предметной области. Классами объектов исследования такой предметной метаобласти являются класс логических формул и, в частности, высказываний интерпретируемых на заданной предметной области; класс элементов заданной предметной области, используемых в качестве констант в указанных логических формулах; класс переменных, используемых в указанных логических формулах, возможными значениями которых являются соответствующие элементы заданной предметной области;
- логическая система понятий, описываемых в заданной формальной теории. Эта предметная метаобласть выделяет класс понятий, неопределяемых в заданной формальной теории, и связывает каждое определяемое понятие с теми понятиями, на основе которых оно определяется;
- логическая система утверждений заданной формальной теории. Эта предметная метаобласть выделяет класс аксиом для заданной формальной теории, каждой теореме ставит в соответствие одно из её доказательств и связывает каждую теорему со всеми утверждениями и определениями, которые используются в основном доказательстве этой теоремы.

Заметим, что некоторым из перечисленных классов предметных областей может соответствовать унифицированный набор используемых в них ключевых понятий. Унификация (стандартизация) таких наборов понятий — важнейшее средство более глубокой семантической совместимости (интегрируемости) различных фрагментов базы знаний. Результат такой унификации — разработка средств SC-кода, ориентированных на представление предметных областей соответствующего класса. Такие языковые средства будем называть специализированным **sc-языком**.

К числу таких специализированных sc-языков можно отнести:

- **теоретико-множественный sc-язык**, обеспечивающий описание теоретико-множественных характеристик и связей заданного семейства объектов. С помощью такого языка, в частности, могут быть представлены предметные области, являющиеся теоретико-множественными отношениями других предметных областей;
- **терминологический sc-язык**, обеспечивающий построение терминологических сетей;
- **логический sc-язык**, обеспечивающий построение sc-текстов формальных теорий.

Таким образом, **SC-код** — ядро целого семейства различных **sc-языков**, ориентированных на описание различных классов предметных областей, в каждый из которых входят предметные области с разными множествами исследуемых объектов, но с одинаковыми предметами исследования.

Каждому такому специализированному sc-языку ставится в соответствие **множество ключевых узлов**, обозначающих различные классы исследуемых объектов, различные отношения и алгебраические операции, заданные на множестве исследуемых объектов.

SC-язык, являющийся объединением всевозможных специализированных sc-языков, будем называть **языком SCK** (Semantic Code Knowledge). Этот язык рассматривается, как интегрированный язык представления знаний. Язык SCK — открытый (расширяемый), поскольку его всегда можно пополнить новым sc-языком, описывающим структуры нового вида предметных областей.

Построение **семантической структуры базы знаний** интеллектуальной системы требует не только явного представления спецификаций каждой описываемой предметной области в виде sc-текста, но и явного описания всевозможных связей между этими предметными областями.

Переходя к рассмотрению отношений, заданных на множестве предметных областей, мы фактически переходим к некоторой предметной *метаобласти*, объектами исследования которой являются всевозможные предметные области (в т.ч. и сама эта предметная метаобласть).

Обобщая понятия «гомоморфизм» и «изоморфизм» алгебраических систем, можно говорить о гомоморфизме и изоморфизме предметных областей. Это даёт основу для выявления глубоких нетривиальных аналогий между предметными областями.

Элементы одной предметной области также могут быть элементами другой. При этом возможны различные варианты такого пересечения: строгое пересечение, строгое включение. Общие элементы пересекающихся предметных областей в рамках этих областей могут играть как одинаковые, так и разные роли. Так, например, первичные элементы одной предметной области могут входить в состав другой предметной области в качестве вторичных или ключевых элементов. Объекты исследования одной предметной области могут входить в состав другой предметной области и в качестве вспомогательных элементов.

В качестве примеров отношений, заданных на множестве предметных областей, рассмотрим несколько вариантов выделения частных предметных областей:

— на основе выделения подмножества из максимального класса исследуемых объектов. Таким способом из предметной области геометрии, объектами исследования которой являются геометрические точки, фигуры и семейства фигур, можно выделить предметную область планиметрии, изучающую планарные фигуры и планарные семейства фигур (т. е. семейства фигур, лежащих в одной плоскости) и предметную область стереометрии, которая изучает непланарные фигуры и непланарные семейства фигур, которые могут состоять как из непланарных, так и из планарных фигур;

— выделение частной предметной области, на основе выделения подмножества из семейства классов исследуемых объектов. Таким способом из предметной области, изучающей треугольники, можно выделить предметную область, изучающую остроугольные, тупоугольные и прямоугольные треугольники, а также предметную область, изучающую равнобедренные, разносторонние и равнобедренные треугольники;

— выделение частной предметной области, на основе выделения подмножества из семейства отношений, заданных на исследуемых объектах. Таким способом из геометрической предметной области можно выделить предметную область, объектами исследования которой являются геометрические фигуры, а предметом исследования — их числовые характеристики, и предметную область, объектами исследования которой являются геометрические фигуры, а предметом исследования — различные виды их конгруэнтности (движений).

Предметная область позволяет рассматривать исследуемые объекты на разных уровнях детализации, как в рамках исходной (заданной) предметной области, расширяя эту предметную область в соответствующих направлениях, так и в системе самостоятельных, но связанных между собой предметных областей.

Первый и важнейший этап проектирования базы знаний — уточнение структуры описываемой предметной области или нескольких взаимосвязанных предметных областей. Уточ-

нение такой структуры — это прежде всего уточнение класса исследуемых объектов, предмета исследования, всего семейства ключевых узлов семантической сети, представляющей предметную область. При этом для заданного класса исследуемых объектов и заданного предмета исследования можно построить более и менее качественные предметные области.

Рассмотрим ещё один тип фрагментов баз знаний — **семантические окрестности**. В общем случае семантическая окрестность заданного объекта — описание некоторых примет, числовых характеристик, свойств и связей заданного объекта. Частными видами семантических окрестностей являются:

- описание характеристик, свойств или связей заданного объекта, *однозначно* определяющих (устанавливающих) заданный объект (для понятий — это определение или любое другое высказывание определяющего типа);
- полная (интегрированная) семантическая окрестность заданного объекта, содержащая все известные в *текущий момент* сведения об этом объекте в рамках заданного раздела базы знаний или в рамках всего текущего состояния базы знаний;
- описание числовых характеристик (параметров, признаков) заданного объекта;
- описание свойств заданного объекта (истинные, не фактографические высказывания, в которых знак этого объекта используется в качестве константы);
- описание теоретико-множественных связей заданного объекта с другими объектами;
- описание разноязычных терминов, иероглифов, пиктограмм, используемых для внешней идентификации заданного объекта (в т. ч. и описание происхождения этих идентификаторов);
- перечень отношений, соответствующих заданному объекту (отношений, в области определения которых заданный объект входит в качестве элемента, подмножества, надмножества, отношений, область определения каждого из которых строго пересекается с заданным объектом).

В **sc-модели базы знаний** каждая семантическая окрестность представляется в виде соответствующего sc-текста. Для явного введения (задания) этой семантической окрестности в рамках sc-модели базы знаний необходимо:

- 1) явно ввести sc-узел, обозначающий эту семантическую окрестность;
- 2) явно связать введённый sc-узел sc-дугами основного вида со всеми элементами обозначаемого им sc-текста;
- 3) явно указать центральный элемент семантической окрестности с помощью ролевого отношения **быть центром семантической окрестности**;
- 4) явно связать введённый sc-узел sc-дугой основного вида со специальным sc-узлом, обозначающим **класс sc-структур, являющихся семантическими окрестностями**.

После этого можно описывать различные характеристики (в частности, типологию) явно введённой семантической окрестности, а также её различные связи с другими фрагментами базы знаний.

Принцип 7. ГРАФОВЫЕ ЯЗЫКИ ПРОГРАММИРОВАНИЯ

Графовые языки программирования используются для описания способов решения задач и поведения агентов над общей графодинамической памятью. Они ориентированы на обработку унифицированных семантических сетей. Их программы — унифицированные семантические сети.

Если все используемые в интеллектуальной системе графовые языки программирования привести к общему унифицированному стандарту — SC-коду, то можно достаточно эффективно решать проблему формализации семантической совместимости программ, написанных не только на одном, но и на разных языках программирования.

В традиционных языках программирования синтаксическая структура и семантика хранящихся в памяти обрабатываемых данных отдаются на откуп программисту. Поэтому о семантической совместимости программ не может идти и речи.

Наряду с применением в интеллектуальных системах целого множества sc-языков различного назначения, востребованным является использование целого семейства *совместимых* sc-языков программирования, которые могут иметь разный уровень, быть последовательными, процедурными и декларативными.

Важнейшая особенность всех этих языков — использование ассоциативного доступа к обрабатываемым фрагментам хранимого в графодинамической памяти sc-текста.

Операционная семантика каждого такого графового языка программирования (точнее, sc-языка программирования) задаётся коллективом агентов над общей графодинамической памятью, обеспечивающих интерпретацию любой программы указанного языка программирования, хранящейся вместе с обрабатываемой информацией в указанной графодинамической памяти.

Программы, представленные в виде семантической сети и описывающие обработку семантических сетей, а также соответствующие им языки программирования, фактически, открывают новую страницу в теории программирования, которую можно назвать семантической теорией программ и языков программирования. Основным лейтмотивом такой теории должно быть обеспечение семантической совместимости программ и языков программирования.

Принцип 8. Унификация формального описания агентов, работающих над семантической памятью

Выделение базового sc-языка программирования предназначено для унификации формального описания поведения агентов, работающих над общей графодинамической памятью. Такой базовый sc-язык программирования будем называть **языком SCP** (Semantic Code Programming), а написанные на нём программы — **scp-программы**.

Принцип 9. Унификация семантических моделей обработки знаний

На основе унифицированных семантических сетей (sc-текстов) уточнить понятие «**Унифицированная модель обработки информации**», а также понятие «Унифицированная модель решения задач».

Все указанные абстрактные модели будем называть **sc-моделями обработки знаний** или **sc-машинами**, поскольку в их основе лежит использование SC-кода. Каждая такая модель (sc-машина) представляет собой многоагентную систему, состоящую из **sc-памяти** — графодинамической памяти, в которой хранятся и обрабатываются тексты SC-кода, и **sc-агентов** — коллектива агентов, работающих над общей для них sc-памятью и взаимодействующих между собой только через неё.

Очевидно, что sc-модели обработки знаний являются частным, унифицированным видом графодинамических моделей параллельной асинхронной обработки информации.

Каждый sc-агент реагирует на соответствующий ему класс ситуаций и/или событий, происходящих в sc-памяти, и осуществляет определённое преобразование sc-текста, находящегося в семантической окрестности обрабатываемой ситуации и/или события. Типология sc-агентов достаточно богата:

- sc-агенты, обеспечивающие интерпретацию программ различных sc-языков программирования высокого уровня;
- sc-агенты информационного поиска;
- sc-агенты, обеспечивающие реализацию правил логического вывода, соответствующих различным логическим исчислениям;
- sc-агенты сведения задач к подзадачам;
- sc-агенты анализа качества хранимой базы знаний, в частности, её корректности, полноты;
- sc-агенты обнаружения автоматического склеивания синонимичных sc-элементов;
- sc-агенты автоматического устранения некоторых ошибок в базе знаний;
- sc-агенты удаления информационного мусора (в частности, удаления фрагментов базы знаний, которые редко востребованы и могут быть достаточно легко восстановлены в случае их отсутствия);
- sc-агенты, обеспечивающие трансляцию вводимой информации с различных внешних языков в SC-код;
- sc-агенты, обеспечивающие трансляцию sc-текстов, вводимых пользователю на различные внешние языки;
- рецепторные sc-агенты;
- эффекторные sc-агенты.

В понятии sc-машины набор агентов не задаётся, т.е. могут существовать разные sc-машины с разным набором sc-агентов. Несколько разных sc-машин можно **интегрировать**. С формальной точки зрения это сделать несложно. Во-первых, интегрировать sc-текст, описывающий текущее состояние взаимодействия sc-агентов одной sc-машины, с аналогичным sc-текстом другой sc-машины. Во-вторых, полученный интегрированный sc-текст поместить в sc-память интегрированной sc-машины. В-третьих, включить в интегрированную sc-машину все sc-агенты первой и второй интегрируемых sc-машин.

Более того, одна sc-машина может **интерпретировать** другую. Для этого необходима разработка целого семейства sc-языков программирования различного уровня. Тексты (программы) всех этих языков должны храниться в sc-памяти, т.е. должны быть семантическими сетями, представленными в SC-коде. Операционная семантика (интерпретация) каждого из этих языков задаётся определённым набором sc-агентов, процедура выполнения (поведения) каждой из которых описывается программой, написанной на языке *более низкого уровня*.

В абстрактных sc-машинах можно выделить следующие языки программирования.

- Семейство sc-языков программирования высокого и сверхвысокого уровня (как процедурные, так и непроцедурные). Тексты программ этих языков хранятся в базе знаний интеллектуальной системы и описывают способы решения различных классов задач в соответствующих предметных областях.
- Базовый sc-язык программирования (язык SCP), на котором описываются sc-агенты и интерпретации sc-языков программирования высокого и сверхвысокого уровня, а также sc-операции, обеспечивающие интерпретацию различных логических исчислений, различных моделей интеллектуального решения задач.
- SC-язык программирования, на котором описываются sc-агенты интерпретации базового sc-языка программирования. Фактически, это первый язык микропрограмм для **sc-компьютера**, обеспечивающего аппаратную интерпретацию языка SCP.

— При необходимости можно ввести второй язык микропрограммирования, описывающий интерпретацию первого и т.д.

На основе понятия абстрактной sc-машины можно уточнить понятие «Унифицированная логико-семантическая модель интеллектуальной системы». Такую унифицированную модель интеллектуальной системы будем называть абстрактной **sc-моделью** интеллектуальной системы или, сокращённо, **sc-системой**. Абстрактная sc-система включает в себя **sc-модель базы знаний** — интегрированную базу *всех* знаний, которые необходимы для функционирования интеллектуальной системы и представлены в виде интегрированного sc-текста, и абстрактную **sc-машину**, в памяти которой хранится sc-текст базы знаний абстрактной системы.

Нетрудно заметить, что sc-текст базы знаний интеллектуальной системы является формальным и унифицированным уточнением того, что должна знать проектируемая интеллектуальная система. А sc-машина интеллектуальной системы и, в первую очередь, набор входящих в её состав sc-агентов, является формальным и унифицированным уточнением того, что должна эта система уметь делать со своими знаниями.

Подчёркнём, что чёткое выделение абстрактного семантического уровня интеллектуальной системы позволяет не только обеспечивать их семантическую совместимость, но и сформировать критерии сравнения интеллектуальных систем по уровню их возможностей. Очевидно, что уровень возможностей интеллектуальной системы определяется качеством (корректностью, полнотой, многообразием) её *знаний* и эффективностью её *умений* (т.е. эффективностью используемых её моделей решения задач).

Заметим, что абстрактную логико-семантическую модель, в принципе, можно построить для любой компьютерной системы, обеспечивая их семантическую совместимость на абстрактном логико-семантическом уровне.

Принцип 10. Унификация семантических моделей информационного поиска

На основе унифицированных семантических сетей (т.е. на основе SC-кода) обеспечить построение **унифицированных семантических моделей информационного поиска** — унифицированных семантических моделей ассоциативного доступа к информации, хранимой в семантической памяти.

Ассоциативный доступ — это доступ, основанный не на знании того, где находится искомая информация (в частности, на знании адреса или имени соответствующей области памяти), а на знании того, как искомая информация связана с известной информацией, хранимой в памяти.

Эффективность организации информационного поиска в базе знаний интеллектуальной системы во многом определяет эффективность самой интеллектуальной системы. Это обусловлено тем, что время, затрачиваемое интеллектуальной системой на поиск нужных в текущий момент знаний и навыков, занимает, мягко говоря, не меньше половины времени, затрачиваемого на решение задачи в целом.

Унифицированная семантическая модель информационного поиска (**sc-модель информационного поиска**) включает в себя, во-первых, **SC-язык вопросов**, с помощью которого в виде sc-текстов осуществляется описание (спецификация) запрашиваемых (искомых) фрагментов всего интегрированного sc-текста, хранимого в текущий момент в sc-памяти (т.е. sc-текста, который является sc-моделью базы знаний). Тексты, принадлежащие SC-языку вопросов, будем называть **sc-вопросами**. Во-вторых, **SC-язык оформления ответов**, с помощью которого осуществляется явное выделение sc-текстов, являющихся ответами, и явное описание их связи с явно выделенными sc-текстами, которые представляют вопросы, соответствующие указанным ответам. В-третьих, семейство информационно-поисковых sc-агентов, каждый из которых реагирует на соответствующий ему

тип инициированного sc-вопроса и выполняет соответствующую поисковую процедуру в sc-памяти.

Семантическая типология вопросов — предмет отдельного рассмотрения. Приведем фрагмент такой типологии, чтобы проиллюстрировать семантическую мощьность sc-языка вопросов.

Прежде всего, по аналогии с логическими формулами множество вопросов разбивается на **атомарные вопросы** и **неатомарные вопросы**, каждый из которых представляет собой конечное множество вопросов.

Компонентами неатомарного вопроса могут быть как атомарные, так и неатомарные вопросы. При этом, если построить оргграф, вершинами которого будут знаки всех вопросов, входящих в состав заданного неатомарного вопроса, а дуги будут связывать знаки неатомарных вопросов, входящих в состав заданного неатомарного вопроса, с их компонентами, то этот оргграф будет деревом, все конечные вершины которого являются знаками атомарных вопросов. Частным видом неатомарного вопроса является конъюнктивный вопрос, ответом на который является конъюнкция (интеграция) ответов на все вопросы, являющиеся компонентами этого конъюнктивного вопроса.

Поскольку в общем случае вопросу может соответствовать *несколько* правильных ответов (т.е. ответов, удовлетворяющих, релевантных, соответствующих заданному вопросу), множество вопросов разбивается на вопросы, запрашивающие:

- все правильные ответы;
- один (или, по крайней мере, один) правильный ответ;
- вопросы, запрашивающие несколько разнообразных правильных ответов;
- вопросы, запрашивающие точно указанное число правильных ответов.

Специальным видом неатомарных вопросов являются **сколько-вопросы**, запрашивающие не сами правильные ответы некоторого вопроса (который может быть атомарным и единственным компонентом сколько-вопроса), а количество правильных ответов.

Типы атомарных вопросов

Какой-вопрос атомарного вида. Каждый такой вопрос запрашивает фрагменты базы знаний, *изоморфные* заданному **образцу**, который может иметь *произвольный* размер, произвольную конфигурацию и может быть представлен не только *логической формулой существования*, в которой квантор существования действует на конъюнкцию *атомарных логических формул*, но также и логической формулой существования, в которой квантор существования действует на логическую формулу *произвольного вида*. Суть атомарного *какой-вопроса* заключается в поиске знаков таких объектов, которые заданным образом связаны с другими известными и неизвестными (искомыми) объектами, т.е. в поиске знаков таких объектов, которые удовлетворяют заданным требованиям. На основе *какой-вопросов* атомарного вида строится важный класс неатомарных вопросов, ключевыми компонентами которых являются атомарные *какой-вопросы*, а остальными компонентами — вопросы любого вида, в формулировках которых используются переменные, входящие в состав соответствующих ключевых *какой-вопросов*.

Запрос всех элементов заданного конечного множества (чаще всего это множество из элементов некоторой структуры);

Запрос внешней информационной конструкции, представленной некоторым файлом в том или ином формате.

Запрос полного текста заданного высказывания.

Ли-вопрос, запрашивающий факт истинности или ложности заданного высказывания в рамках заданной формальной теории.

Вопрос выбора альтернатив, запрашивающий одно или несколько истинных высказываний из заданного множества высказываний.

Почему-вопрос, запрашивающий обоснование (доказательство) истинности заданного высказывания.

Что-это-вопрос, запрашивающий основные сведения об указываемом объекте. Фактически, речь идёт о выделении из базы знаний семантической окрестности, «центром» которой является знак указываемого объекта. Таким объектом может быть все, что угодно — понятие, предметная область, формальная теория, высказывание, любая структура, материальный объект.

Запрос общих свойств объектов, принадлежащих заданному классу.

Запрос идентифицирующих признаков заданного объекта. Здесь запрашиваются фрагменты базы знаний, каждый из которых *однозначно* определяет (устанавливает, идентифицирует) заданный объект. Если заданным объектом является понятие, то таким идентифицирующим признаком является либо определение этого понятия, либо соответствующая теорема о необходимости и достаточности.

Запрос связей между заданными объектами.

Запрос сравнительного анализа заданных объектов.

Запрос сходств заданных объектов (сходства, аналогии — частный вид связей между объектами).

Запрос отличий заданных объектов (отличия объектов — частный вид связей между ними).

Запрос плана решения заданной конкретной задачи, т.е. плана достижения заданной цели в заданных конкретных условиях.

Запрос обобщённого способа решения любой задачи из заданного класса задач. Таким обобщённым способом может быть алгоритм, декларативная (непроцедурная) программа, нестрогое предписание (рекомендация).

Зачем-вопрос, запрашивающий то, какой надцели соответствует заданная цель, которая может быть сформулирована как в декларативной, так и в процедурной форме.

Список типов атомарных вопросов можно продолжить, но почти все они будут подтипами (подмножествами) перечисленных типов вопросов. В основе sc-языка вопросов лежит построение онтологии вопросов, в рамках которой чётко прописываются все теоретико-множественные (в первую очередь, родо-видовые) связи между *всеми* выделенными типами и подтипами вопросов. При этом в формулировке *каждого* конкретного sc-вопроса *явным образом* отражаются иерархия *всех* типов вопросов, которым принадлежит данный конкретный sc-вопрос. Для этого каждому типу вопросов ставится в соответствие ключевой sc-узел, обозначающий этот тип вопросов.

В заключение заметим, что в **SC-языке оформления ответов** кроме отношения релевантности, связывающего вопрос с правильными на него ответами, используются языковые средства, описывающие качество, полноту ответов. Это вызвано тем, что некоторые типы вопросов предполагают наличие целого множества правильных ответов, но разного качества.

Принцип 11. Унификация семантических моделей интеграции знаний и семантических моделей интеграции целых интеллектуальных систем

На основе унифицированных семантических сетей обеспечить построение **унифицированных семантических моделей интеграции знаний** (понимания знаний) и использовать эти модели, во-первых, как основу процесса приобретения интеллектуальной системой новых знаний как со стороны конечных пользователей, так и со стороны разработчиков; во-вторых, как основу интеграции программ и различных семантических моделей расширения задач; в-третьих, как основу интеграции абстрактных логико-семантических моделей интеллектуальных систем.

Главное свойство интеллектуальной системы — не те интеллектуальные знания и навыки, способности, которые она имеет в текущий момент, а *мета-способность* приобретать *любые* необходимые ей *новые* знания и навыки. А для этого интеллектуальная система, как минимум, должна уметь интегрировать эти приобретаемые знания и навыки. Следовательно, проблема формализации интеграции знаний и навыков — центральная для деятельности интеллектуальных систем.

Принципиальные отличия интеграции двух интеллектуальных систем заключается в следующем: пусть имеются две интеллектуальных системы $s1$ и $s2$, первая из которых способна решать задачи из множества $q1$, вторая — $q2$. В результате простого соединения этих систем получаем систему, которая способна решать задачи из множества $(q1 \cup q2)$. Тогда как в результате интеграции мы получаем систему, которая способна решать значительно большее число задач, чем $(q1 \cup q2)$. Таким образом, при интеграции интеллектуальных систем происходит приобретение нового качества «на стыке» интегрируемых систем, когда для решения некоторых задач одна часть необходимых знаний и/или умений находится в одной интегрируемой системе, а другая часть — в другой системе.

Процесс интеграции двух семантических сетей рассмотрим как систему следующих взаимодействующих подпроцессов, некоторые из которых могут выполняться параллельно:

- приведение интегрируемых семантических сетей к унифицированному виду, т.е. представление (запись) их в SC-коде;
- согласование ключевых узлов и онтологий, используемых в интегрируемых sc-текстах. Очевидно, что полностью автоматизировать такое согласование невозможно, поэтому разработчикам интегрируемых фрагментов баз знаний и целых баз знаний необходимо *уметь договариваться* друг с другом;
- выделение в интегрируемых sc-текстах таких sc-элементов, которые имеют глобальные (уникальные) идентификаторы (внешние имена);
- выделение в интегрируемых sc-текстах sc-элементов, имеющих локальные идентификаторы вместе с областью действия каждого такого идентификатора. Область действия локального идентификатора — такой фрагмент базы знаний, в рамках которого разные sc-элементы считаются синонимичными;
- склеивание sc-элементов, имеющих одинаковые глобальные идентификаторы;
- склеивание sc-элементов, имеющих одинаковые локальные идентификаторы, если каждый из этих sc-элементов принадлежит области действия своего и другого локальных идентификаторов;
- склеивание sc-элементов на основании однозначности используемых алгебраических операций;

- склеивание sc-элементов на основании логических высказываний о существовании единственности;
- склеивание кратных связей, принадлежащих отношениям:
- не имеющих кратных связей;
- имеющих кратные связи, но не для заданных типов компонентов (например, кратные связи принадлежности не могут выходить из знаков канторовских множеств).

Таким образом, интеграция семантических сетей, т.е. процесс погружения (понимания) одной семантической сети в другую — нетривиальный процесс рассуждений, направленный на выявление пар синонимичных элементов семантической сети на основе определённых знаний, имеющихся в базе знаний интеллектуальной системы.

От унифицированной семантической модели интеграции знаний (точнее, sc-текстов) можно достаточно легко перейти к интеграции sc-моделей интеллектуальных систем, поскольку после интеграции sc-моделей баз знаний интегрируемых интеллектуальных систем интеграция соответствующих им наборов sc-агентов сводится к простому теоретико-множественному объединению указанных множеств sc-агентов.

Принцип 12. Унификация и интеграция различных семантических моделей решения задач

Обеспечить в рамках проектируемой интеллектуальной системы использование не только различных видов знаний, но и **различных моделей и стратегий решения задач**.

Для этого необходимо акцентировать внимание не столько на разработке новых моделей решения задач, сколько на унификации и интеграции в рамках проектируемых интеллектуальных систем уже разработанных и хорошо зарекомендовавших себя моделей (дедуктивных, индуктивных, абдуктивных, чётких, нечётких, универсальных, специализированных). Подчеркнём то, что в разных проектируемых интеллектуальных системах могут быть востребованы самые разные сочетания известных моделей и стратегий решения задач. Подавляющее число моделей представления знаний и решения задач не являются альтернативными и *дополняют друг друга*. Не составляют исключение и такие классы моделей, как фреймовые, логические, продукционные.

Рассмотренное выше понятие **вопроса** и его формализация являются основой не только для информационно-поисковых моделей, но и для различных моделей решения задач. С точки зрения решателя задач, вопрос — **непроцедурная формулировка информационной цели**, т.е. декларативная формулировка некоторой информационной цели, которая описывает спецификацию (свойства) той информации, которую *требуется* либо найти, если она уже присутствует в текущем состоянии базы знаний, либо построить (сгенерировать, вывести), если она отсутствует в текущем состоянии памяти. Таким образом, вопрос можно считать описанием целевого (требуемого) состояния обрабатываемой базы знаний (а точнее, определённого фрагмента этой базы знаний). Вопрос также можно считать одним из видов *метазнаний*, описывающих (специфицирующих) наше *незнание*, т.е. наше знание о том, что мы не знаем, но хотели бы знать.

Вопросы могут инициироваться (задаваться) как пользователями, так и самой системой. Это означает, что в процессе обработки информации интеллектуальная система сама себе может задавать вопросы. Если инициирован некоторый sc-вопрос, то сначала активизируются соответствующие агенты информационного поиска в «надежде» на то, что запрашиваемый ответ (или ответы) на указанный sc-вопрос уже присутствует в текущем состоянии базы знаний. И только после того, как информационно-поисковые sc-агенты обнаружат отсутствие ответа в текущем состоянии базы знаний, начинается работа решателя задач, направленная на генерацию (построение, порождение, вывод) требуемого ответа.

Кроме вопроса используется также и **процедурная формулировка информационной цели** — описание (спецификация) некоторого действия, которое требуется выполнить

и которое направлено на преобразование базы знаний, хранимой в некоторой памяти. Указанное действие, выполняемое одним или несколькими sc-агентами, порождает определённое событие (изменение состояния sc-памяти).

Для унификации различных моделей решения задач необходимо уточнить не только понятие **информационной цели**, но и понятие **информационной задачи**. Информационная задача задаётся, во-первых, формулировкой информационной цели (*что требуется*), и, во-вторых, той хранимой в памяти информацией, которая семантически связана с заданной информационной целью (*что дано*). В пределе, контекстом информационной цели можно считать текущее состояние всей хранимой базы знаний.

Формальное рассмотрение контекстов различных информационных задач требует разработки специальных языковых средств, предназначенных для описания *текущего состояния хранимой* базы знаний, а точнее, для описания «границ» между тем, что в текущем состоянии базы знаний известно, и тем, что неизвестно. К числу таких языковых средств, в частности, относятся следующие ключевые узлы, являющиеся знаками нестационарных множеств (т.е. множеств, которые в разные моменты времени могут иметь разные элементы):

- быть sc-дугой нечёткой принадлежности (такая sc-дуга связывает sc-узел, обозначающий некоторое множество, с sc-элементом, о котором в текущий момент времени неизвестно, принадлежит он указанному множеству или нет);
- быть построенным конечным множеством (у каждого такого множества в текущем состоянии базы знаний известны и явно указаны *все* его элементы);
- быть построенным высказыванием (для каждого такого высказывания в текущем состоянии базы знаний представлен не только его знак, но и полный текст);
- быть построенной внешней информационной конструкцией (файлом);
- быть аксиоматизированной формальной теорией;
- быть построенным рассуждением (обоснованием, доказательством, решением);
- быть построенной программой.

Более подробно унифицированные семантические модели решения задач и технология их проектирования рассмотрены в работе [26].

Принцип 13. Унификация визуализации семантических сетей

Использовать язык унифицированного визуального представления абстрактных унифицированных семантических сетей в виде, близком к изоморфному в качестве основы организации графического пользовательского интерфейса.

Указанный язык графического изображения sc-текстов назван SCg-кодом (Semantic Code graphical). Подчеркнём, что следует чётко отличать язык абстрактных унифицированных семантических сетей (SC-код), который абстрагируется от того, как должны быть физически представлены узлы и коннекторы текстов этого языка (sc-текстов), и язык графического изображения таких семантических сетей. Т.е. абстрактная семантическая сеть и её рисунок — принципиально разные вещи.

С помощью SCg-кода осуществляется отображение на экране не только пользовательских сообщений, адресуемых системе, и сообщений, адресуемых пользователю, но и всей остальной информации, необходимой для организации работы пользователя (прежде всего, элементы управления интерфейсом). Такая унификация отображаемой пользователю информации даёт возможность организовать взаимодействие пользователя с help-системой точно так же, как и его взаимодействие с основной (предметной) системой.

Трактовка элементов управления пользовательским интерфейсом как элементов отображаемого на экране SCg-текста позволяет унифицировать представления *любой* информации, отображаемой на экране, и способы инициирования различных вопросов, касающихся *любой* отображаемой на экране информации (в том числе, и элементов управления).

Для того, чтобы чётко отделить те средства SCg-кода, которые обусловлены самим SC-кодом, от тех средств, которые обусловлены стремлением повысить наглядность SCg-текстов, введём ядро SCg-кода (или просто SCg-ядро), алфавит которого взаимно однозначно соответствует алфавиту SC-кода и, соответственно, тексты которого *изоморфны* семантически эквивалентным текстам SC-кода.

Переход от SCg-ядра к SCg-коду заключается в ослаблении требований, предъявляемых к изображениям семантических сетей, в целях обеспечения удобства для человеческого восприятия. Такое ослабление осуществляется в следующих направлениях: вводится приписывание идентификаторов изображаемых sc-элементов, расширяется алфавит графических примитивов, допускается уникальное изображение некоторых sc-узлов, допускается синонимия sc.g-элементов, но при этом синонимичным элементам должны быть приписаны одинаковые идентификаторы, вводятся специальные графические средства, направленные на повышение наглядности (шинные линии, контуры).

Заметим также, что кроме SCg-кода для внешнего представления абстрактных унифицированных семантических сетей используются также и другие языки:

- SCs-код, обеспечивающий представление унифицированных абстрактных семантических сетей (sc-текстов) в виде, близком к традиционным текстам;
- SCn-код, обеспечивающий гипертекстовое представление абстрактных sc-текстов, предназначенное для оформления исходных текстов баз знаний.

Более подробно различные языки внешнего представления абстрактных sc-текстов вместе с большим количеством примеров рассмотрены в работах [12 — 14].

Принцип 14. УНИФИКАЦИЯ СЕМАНТИЧЕСКИХ МОДЕЛЕЙ РАЗЛИЧНЫХ ПОЛЬЗОВАТЕЛЬСКИХ ИНТЕРФЕЙСОВ

Пользовательский интерфейс интеллектуальной системы, построенной на основе предлагаемой технологии, рассматривать как **специализированную интеллектуальную систему**, предназначенную для **трансляции адресуемых пользователю сообщений** с внутреннего абстрактного семантического языка представления знаний (SC-кода) на тот или иной внешний язык, тексты которого отображаются пользователю в удобном для него виде, а также для **трансляции пользовательских сообщений** с внешнего языка на внутренний семантический язык интеллектуальной системы (SC-код).

Трактовка пользовательских интерфейсов как интеллектуальных систем и унификация семантических моделей таких систем дают возможность, во-первых, унифицировать проектирование пользовательских интерфейсов. Во-вторых, легко наращивать возможности пользовательских интерфейсов. В-третьих, легко интегрировать пользовательские интерфейсы с предметными (основными) интеллектуальными системами. И наконец, неограниченно использовать базу знаний предметных интеллектуальных

систем для семантического анализа и понимания вводимой пользователем информации (в частности, естественно-языковых текстов).

Более подробно унифицированные семантические модели пользовательских интерфейсов и технология их проектирования рассмотрены в работе [34].

Принцип 15. Библиотека типовых семантически совместимых компонентов интеллектуальных систем и методика модульного проектирования интеллектуальных систем

Создать общую библиотеку многократно используемых семантически совместимых компонентов интеллектуальных систем, на основе которой разработать методику модульного (компонентного, сборочного) проектирования интеллектуальных систем, в целях ускорения процесса проектирования интеллектуальных систем.

В указанной библиотеке можно выделить следующие разделы (частные библиотеки):

- библиотека многократно используемых компонентов баз знаний. Прежде всего, в эту библиотеку входят самые различные по содержанию, но семантически совместимые онтологии. Кроме того, сюда попадают различные «джентльменские наборы» знаний, которыми должны владеть «образованные» интеллектуальные системы. К таким знаниям, в частности, относятся базовые знания по арифметике, теории множеств, теории отношений, логике и многие другие знания, востребованность которых может быть самой разной;
- библиотека компонентов семантических моделей информационного поиска. Сюда, прежде всего, попадают различные информационно-поисковые агенты;
- библиотека компонентов семантических моделей интеграции знаний и машин обработки знаний;
- библиотека интерпретаторов программ, соответствующих различным языкам программирования;
- библиотека различных стратегий решения задач, различных моделей решения задач и агентов, входящих в состав таких моделей;
- библиотека компонентов пользовательских интерфейсов.

Все компоненты, включаемые в состав общей библиотеки компонентов интеллектуальных систем, оформляются как компоненты интеллектуальной собственности (intellectual property), поэтому будем их также называть ip-компонентами.

Особо подчеркнём, что это модульное проектирование интеллектуальных систем возможно только в том случае, если отбор компонентов, включаемых в состав рассмотренной библиотеки, будет осуществляться на основе тщательного анализа качества этих компонентов. Одним из важнейших критериев такого анализа является семантическая совместимость анализируемых компонентов со всеми компонентами, имеющимися в текущей версии библиотеки.

Для обеспечения семантической совместимости таких компонентов интеллектуальных систем, которые являются унифицированными семантическими моделями (sc-моделями знаний, sc-моделями машин обработки знаний, sc-агентов, sc-моделями интеллектуальных подсистем), необходимо согласовать семантику (смысл) всех используемых ключевых узлов и глобальные идентификаторы ключевых узлов, используемых в разных компонентах.

После этого интеграция всех компонентов, входящих в состав библиотеки, и в любых комбинациях осуществляется автоматически, без вмешательства разработчика.

Принцип 16. ПЛАТФОРМЕННО-НЕЗАВИСИМЫЙ ХАРАКТЕР ПРОЕКТИРОВАНИЯ ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ

Для максимальной платформенной независимости технологии обеспечить чёткое разделение процесса проектирования формального описания логико-семантической модели разрабатываемой интеллектуальной системы от процесса реализации (интерпретации) этой модели на той или иной платформе.

Подчеркнём при этом следующее. Если каждой интеллектуальной системе соответствует своя уникальная логико-семантическая модель, то каждый интерпретатор абстрактных логико-семантических моделей интеллектуальных систем должен обеспечивать интерпретацию целого класса таких моделей, а в идеале — интерпретацию любой такой модели. Следовательно, разработка указанных интерпретаторов может осуществляться абсолютно независимо от разработки логико-семантических моделей конкретных интеллектуальных систем.

Таким образом, SC-код, обеспечивающий унификацию семантического представления любых знаний, вместе с языком SCP, обеспечивающим унификацию формального описания агентов, работающих над семантической памятью, являясь средством унификации логико-семантических моделей интеллектуальных систем, выполняют в рамках предлагаемой технологии роль, аналогичную той, которую выполняет язык VHDL в современных микропроцессорных технологиях. В лице SC-кода и языка SCP мы имеем *стандарт полного* формального описания логико-семантических моделей интеллектуальных систем, обеспечивающий независимость проектирования абстрактных логико-семантических моделей конкретных интеллектуальных систем от разработки различных вариантов реализации (различных вариантов их интерпретации на различных платформах). Такой стандарт является своего рода «водоразделом» между полным платформенно-независимым описанием интеллектуальной системы (абстрактной логико-семантической моделью) и платформенно-зависимой реализацией (интерпретацией) этой абстрактной модели.

Полностью построенная абстрактная логико-семантическая модель проектируемой интеллектуальной системы:

- является открытой, поскольку её можно легко пополнять новыми знаниями и навыками, интегрируя их в текущую версию модели;
- концентрирует внимание на семантических аспектах функционирования интеллектуальной системы и не содержит никаких лишних деталей, обусловленных способами её технической реализации (интерпретации);
- является абстрактным инвариантом целого множества различных способов её технологической реализации (в том числе, и с помощью принципиально новых компьютеров).

Разработка прототипа интеллектуальной системы завершается разработкой полной sc-модели этой системы, которая записывается в виде исходного текста с использованием таких языковых средств, как SCg-код, SCs-код, SCn-код. После этого разработчик выбирает один из *универсальных* вариантов интерпретации (реализации) sc-моделей, загружает разработанные им исходные тексты в выбранный интерпретатор и получает прототип, пригодный для опытной эксплуатации и последующего совершенствования.

Если же после этого разработчик интеллектуальной системы что-то не устраивает в выбранном варианте интерпретации sc-моделей (в частности, производительность), должна существовать достаточно продуманная методика совершенствования выбран-

ного варианта интерпретатора sc-моделей интеллектуальных систем. Очевидно, что для каждого такого варианта указанная методика будет иметь свои особенности.

Следовательно, нижние уровни детализации проектируемых интеллектуальных систем, в отличие от верхнего (логико-семантического), являются платформенно-зависимыми. Следовательно, можно говорить о различных модификациях технологии проектирования интеллектуальных систем, соответствующих разным платформам. Напомним при этом, что основная трудоёмкость проектирования интеллектуальных систем, полностью определяющая уровень её возможностей (уровень знаний и навыков), концентрируется именно на первом этапе проектирования — на разработке её абстрактной логико-семантической модели.

Таким образом, проектирование интеллектуальной системы можно организовать как два следующих самостоятельных процесса, выполняемых одновременно и независимо друг от друга:

- 1) процесс разработки абстрактной унифицированной логико-семантической модели проектируемой интеллектуальной системы;
- 2) процесс совершенствования выбранного интерпретатора абстрактных унифицированных логико-семантических моделей интеллектуальных систем.

Заметим, что сама идея обеспечения кросс-платформенной разработки компьютерных систем путём внедрения формального языка, обеспечивающего описание абстрактных (логических) моделей этих систем, не нова. Существует целый ряд кросс-платформенных технологий. Вопрос в том, о каком классе разрабатываемых компьютерных систем идёт речь; какими свойствами обладают используемые абстрактные модели компьютерных систем; какими достоинствами обладает технология разработки самих этих абстрактных моделей.

Вопросы программной реализации и, в частности, web-ориентированной реализации унифицированных логико-семантических моделей интеллектуальных систем рассмотрены в работе [32].

Принцип 17. СЕМАНТИЧЕСКИЙ АССОЦИАТИВНЫЙ ПАРАЛЛЕЛЬНЫЙ КОМПЬЮТЕР

Обеспечить возможность реализации унифицированных логико-семантических моделей интеллектуальных систем на **семантических ассоциативных компьютерах**, специально ориентированных на аппаратную реализацию таких моделей.

Очевидно, что для указанных компьютеров базовый графовый язык программирования (язык SCP) является их ассемблером, т. е. аппаратно интерпретируемым языком программирования.

В связи с проблемой создания компьютеров, ориентированных на обработку знаний, необходимо отметить следующее:

- 1) в таких компьютерах принципиально важна поддержка именно параллельной обработки знаний;
- 2) опыт использования параллельных компьютеров показывает, что эффективное их использование, предполагающее разработку качественных параллельных программ, требует особой профессиональной подготовки и высокой квалификации. Мир параллельного программирования требует особой культуры, стиля мышления. Ещё более серьёзная профессиональ-

ная подготовка требуется для разработки параллельных программ, ориентированных на обработку знаний и использующих ассоциативный доступ к обрабатываемой информации;

3) уровень развития микроэлектронных технологий в настоящее время позволяет достаточно быстро реализовывать самые смелые компьютерные архитектуры и модели обработки информации;

4) созданию параллельных компьютеров для обработки знаний должно предшествовать создание **технологии** проектирования интеллектуальных систем, в основе которой лежат те модели параллельной обработки знаний, которые будут аппаратно поддерживаться в указанных компьютерах. Иначе мы получим «грудю» талантливо сделанного «железа», эффективность использования которого будет весьма низкой;

5) предлагаемая технология проектирования интеллектуальных систем как раз и предполагает последовательное выполнение следующих этапов:

- разработка технологии проектирования абстрактных унифицированных логико-семантических моделей интеллектуальных систем;
- разработка нескольких вариантов программной реализации абстрактных унифицированных логико-семантических моделей интеллектуальных систем, выполненных на современных компьютерах;
- разработка и эксплуатация достаточно большого количества прикладных интеллектуальных систем, совершенствование технологии проектирования интеллектуальных систем на основе приобретённого опыта;
- разработка семантического ассоциативного компьютера, появление которого не отменит ранее созданного. Просто появится ещё один, но уже аппаратный вариант реализации абстрактных унифицированных логико-семантических моделей интеллектуальных систем. Его применение для разработанных абстрактных унифицированных логико-семантических моделей различных прикладных систем для их конечных пользователей абсолютно ничего не изменит, кроме существенного повышения быстродействия.

Рассматривая абстрактную sc-машину обработки знаний на самом верхнем уровне, не уточняем (не детализируем) «внутреннее устройство» sc-агентов обработки знаний. Разработав язык SCP, мы получили возможность формально описывать (детализировать) поведение sc-агентов обработки знаний. Если трактовать язык SCP как ассемблер семантического ассоциативного компьютера, то проектирование этого компьютера можно рассматривать как формальный переход к sc-машинам более низкого уровня, обеспечивающим *интерпретацию sc-машин* более высокого уровня. Существенным здесь является то, что при этом мы не выходим за пределы класса абстрактных sc-машин. Просто вводится последовательность sc-языков программирования всё более низкого уровня, каждый из которых обеспечивает формальное описание sc-агентов, входящих в состав sc-машины, интерпретирующей программы непосредственно предшествующего ему sc-языка программирования более высокого уровня (см. Принцип 9). При этом таких уровней должно быть столько, сколько необходимо для доведения формального описания sc-машин до такого уровня детализации, который позволяет перейти от соответствующего абстрактного языка микропрограммирования к формальному описанию цифровой аппаратуры на языке VHDL.

Аппаратная интерпретация абстрактных sc-машин предполагает создание реконфигурируемой памяти с распределёнными в ней процессорными элементами. Такую интеграцию памяти и процессора будем называть **процессоро-памятью**. Реконфигурируемость (структурная перестраиваемость) памяти может быть обеспечена коммутационной средой для процессорных элементов. Можно рассматривать целый ряд подходов к реализации реконфигурируемой семантической ассоциативной процессоро-памяти. В частности, процессорным элементам можно ставить в соответствие узлы обрабатываемых унифицированных семантических сетей, а коммутируемым каналам связи между процессорными элементами — коннекторы этой семантической сети. В этом случае теку-

щее состояние конфигурации коммутируемых каналов связи будет полностью соответствовать текущему состоянию конфигурации обрабатываемой семантической сети. Следовательно, память «превращается» из пассивного хранилища байтов в коммутационную среду между процессорными элементами.

Принцип 18. Встроенные подсистемы интеллектуальных систем, обеспечивающие их эффективную эксплуатацию и эволюцию

Каждую проектируемую интеллектуальную систему трактовать как **результат интеграции следующих интеллектуальных подсистем:**

- предметной (основной) интеллектуальной системы;
- интеллектуального пользовательского интерфейса;
- интеллектуальной подсистемы адаптивного управления диалогом с конечным пользователем;
- интеллектуальной help-системы для информационного обслуживания и обучения конечных пользователей предметной интеллектуальной системы, которые, начиная работать с системой, не обязаны сразу иметь высокую квалификацию;
- интеллектуальные системы управления проектированием интеллектуальной системы, которая координирует деятельность разработчиков предметной интеллектуальной системы [Грибова, 2010];
- интеллектуальные системы управления информационной безопасностью предметной интеллектуальной системы.

Подчеркнём, что для обеспечения интегрируемости (семантической совместимости) перечисленных интеллектуальных систем они должны проектироваться на основе *одной и той же технологии*.

Таким образом, проектируя каждую интеллектуальную систему, необходимо одновременно проектировать:

- подсистему, которая осуществляет информационное обслуживание и обучение конечных пользователей данной интеллектуальной системы, т.е. фактически является оформлением документации по эксплуатации системы в виде интеллектуальной справочной и обучающей системы. Это существенно расширит контингент конечных пользователей, повысит эффективность эксплуатации системы и существенно упростит эту эксплуатацию;
- подсистему, которая обеспечивает координацию разработчиков проектируемой интеллектуальной системы, поскольку разработка (совершенствование) системы продолжается в ходе её эксплуатации и требует создания специальных методов и компьютерных средств постоянного совершенствования предметной интеллектуальной системы непосредственно в ходе её эксплуатации. Это существенно отодвинет срок её морального старения;
- подсистему, обеспечивающую управление информационной безопасностью проектируемой интеллектуальной системы.

Если подсистема управления проектированием интеллектуальной системы будет создаваться действительно как интеллектуальная система, интегрируемая с основной (предметной) интеллектуальной системой, то в перспективе она может стать не только координатором деятельности разработчиков, но и самостоятельным субъектом проектирования, способным тестировать, диагностировать, анализировать как основную проектируемую интеллектуальную систему, так и себя.

Принцип 19. Доступность и открытость технологии

Обеспечить максимально возможное **расширение контингента разработчиков** интеллектуальных систем, использующих предлагаемую технологию, за счёт максимальной доступности этой технологии и открытого характера её развития.

Если технология проектирования интеллектуальных систем ориентируется на широкое, массовое распространение и на интенсивное собственное развитие, опирающееся на накапливаемый опыт её использования, она должна быть доступной и открытой. Это означает:

- свободный доступ ко всей документации и основанным средствам автоматизации (компьютерной поддержки) проектирования интеллектуальных систем;
- открытость исходных текстов всех основных средств компьютерной поддержки проектирования интеллектуальных систем, всех основных многократно используемых (типовых) компонентов интеллектуальных систем;
- открытость исходных текстов всех «пилотных» проектов прикладных интеллектуальных систем, выполняющих роль «образцово-показательных» проектов;
- открытый характер организации (project-менеджмента) процесса развития технологии, имеющий форму открытого (open source) проекта, участником которого может быть любой желающий, в том числе, и любой пользователь этой технологии, указывающий на различные ошибки и высказывающий различные пожелания.

Завершая рассмотрение открытого характера предлагаемой технологии, сделаем следующие замечания:

- открытый характер технологии не является препятствием для реализации коммерческих интересов, связанных с этой технологией. Так, например, на коммерческой основе могут создаваться и предоставляться, во-первых, различные прикладные интеллектуальные системы; во-вторых, некоторые варианты реализации различных многократно используемых (типовых) компонентов интеллектуальных систем; в-третьих, некоторые варианты реализации интерпретатора абстрактных семантических логико-семантических моделей интеллектуальных систем, в частности, варианты построения семантических ассоциативных параллельных компьютеров;
- открытый характер технологии, при грамотном использовании фактора её открытости, способствует обеспечению информационной безопасности как самой технологии, так и прикладных интеллектуальных систем, созданных на её основе;
- открытый характер предлагаемой технологии проектирования интеллектуальных систем может быть эффективно реализован только на базе **технологии облачных вычислений**, в рамках которой вся предлагаемая технология проектирования интеллектуальных систем рассматривается как некий Internet-сервис [Грибова и др., 2011]

Принцип 20. Эволюционная методика проектирования

Использовать **методику поэтапного эволюционного проектирования** интеллектуальных систем.

Указанная методика предполагает:

- быстрое проектирование;
- скорейшее введение в эксплуатацию первых версий проектируемой системы с минимальными, но практически полезными возможностями;
- эволюционное поэтапное совершенствование проектируемой интеллектуальной системы путём её расширения новыми знаниями и навыками непосредственно в ходе её эксплуатации и активным привлечением конечных пользователей.

С формальной точки зрения, проектирование унифицированной логико-семантической модели (sc-модели) интеллектуальной системы, в конечном счёте, сводится к проектиро-

ванию sc-модели **базы знаний** этой интеллектуальной системы, поскольку scr-программы, описывающие поведение sc-агентов, можно рассматривать как часть базы знаний. Таким образом, проектируемая база знаний включает в себя:

- базу знаний предметной (основной) интеллектуальной системы;
- тексты всех scr-программ, описывающих поведение sc-агентов;
- текст документации, представленный в виде базы знаний интеллектуальной help-системы, обеспечивающей всестороннее информационное обслуживание пользователей проектируемой интеллектуальной системы.

Начальный этап проектирования базы знаний интеллектуальной системы — уточнение иерархической системы предметных областей, которые должны быть описаны в проектируемой базе знаний. Каждой такой предметной области ставится в соответствие определённый раздел проектируемой базы знаний. Среди выделенных разделов проектируемой базы знаний имеются разделы, которые делятся (декомпозируются) на подразделы, а также атомарные (недекомпозируемые) разделы. Далее процесс проектирования всей базы знаний сводится к проектированию каждого её атомарного раздела с последующей их интеграцией в единую базу знаний.

В целом начальную стадию проектирования всей интеллектуальной системы на основе предлагаемой технологии можно условно разбить на четыре этапа:

1. Разработка первой версии интеллектуальной системы, которая включает в себя:
 - первую версию её базы знаний;
 - *типовое ядро* интеллектуальной информационно-поисковой машины, которое входит в состав библиотеки многократно используемых компонентов интеллектуальных систем;
 - *типовое ядро* интеллектуального решателя;
 - *типовое ядро* пользовательского интерфейса.
2. Разработанная первая версия интеллектуальной системы уже обладает определённой целостностью, её можно тестировать и запускать в предварительную опытную эксплуатацию.
3. Разработка второй версии интеллектуальной системы, которая включает в себя:
 - вторую версию её базы знаний;
 - первую версию её информационно-поисковой машины;
 - типовое ядро её интеллектуального решателя;
 - типовое ядро её пользовательского интерфейса.
4. Разработка третьей версии интеллектуальной системы, включающей в себя:
 - третью версию её базы знаний;
 - вторую версию её информационно-поисковой машины;
 - первую версию её интеллектуального решателя;
 - типовое ядро её пользовательского интерфейса.
5. Разработка четвертой версии интеллектуальной систем, включающей в себя:
 - четвертую версию её базы знаний;
 - третью версию её информационно-поисковой машины;
 - вторую версию её интеллектуального решателя;
 - первую версию её пользовательского интерфейса.

Дальнейшее развитие проектируемой интеллектуальной системы может акцентировать внимание на самых разных направлениях, приоритетность которых определяется самим приложением.

Более подробно методика эволюционного коллективного проектирования унифицированных семантических моделей интеллектуальных систем, на основе содержательной структуризации знаний (см. Принцип 6), описана в работе [20].

Принцип 21. РЕАЛИЗАЦИЯ ПРЕДЛАГАЕМОЙ ТЕХНОЛОГИИ В ВИДЕ ИНТЕЛЛЕКТУАЛЬНОЙ МЕТАСИСТЕМЫ

Реализовать предлагаемую технологию как **интеллектуальную метасистему, ориентированную на поддержку проектирования интеллектуальных систем**, построенную по тем же самым принципам (т. е. по той же технологии), что и интеллектуальные системы, разрабатываемые на её основе.

Указанная интеллектуальная система должна включать в себя:

- теорию (принципы построения) проектируемых интеллектуальных систем, которая входит в состав базы знаний метасистемы;
- библиотеку типовых многократно используемых компонентов (ip-компонентов) интеллектуальных систем, которая также входит в состав базы знаний рассматриваемой метасистемы;
- средства автоматизации синтеза, анализа и имитационного моделирования проектируемых интеллектуальных систем и их компонентов;
- интеллектуальную help-систему, являющуюся подсистемой рассматриваемой интеллектуальной метасистемы, ориентированной на информационное обслуживание и обучение разработчиков интеллектуальных систем;
- методику проектирования интеллектуальных систем, которая оформляется как часть базы знаний метасистемы;
- методику обучения проектированию интеллектуальной системы, которая также является частью базы знаний метасистемы;
- интеллектуальную подсистему управления проектированием самой метасистемы;
- интеллектуальную подсистему управления информационной безопасностью метасистемы;
- семейство различных вариантов реализации интерпретаторов унифицированных абстрактных логико-семантических моделей интеллектуальных систем.

Учитывая рассматриваемые выше принципы построения предлагаемой нами технологии, она названа Открытой Семантической Технологией проектирования Интеллектуальных Систем (Open Semantic Technology for Intelligent Systems — OSTIS). Можно было бы назвать её **SC-технологией**, поскольку основой этой технологии является SC-код. Соответственно этому, интеллектуальную метасистему, ориентированную на поддержку проектирования интеллектуальных систем, будем называть **метасистемой OSTIS**.

В интеллектуальной метасистеме OSTIS можно выделить целый ряд подсистем, ориентированных на поддержку проектирования различных компонентов интеллектуальных систем, таких как:

- базы знаний и различные фрагменты баз знаний (онтологии, формальные теории, программы);
- информационно-поисковые машины, машины интеграции знаний, решатели задач;
- пользовательские интерфейсы (графические, естественно-языковые, мультимодальные).

В интеллектуальной метасистеме OSTIS можно также выделить семейство интеллектуальных подсистем, ориентированных на поддержку проектирования различных классов интеллектуальных систем:

- интеллектуальные справочные системы (системы информационного обслуживания);

- интеллектуальные обучающие системы (имеющие подсистемы интеллектуального управления обучением);
- интеллектуальные help-системы для пользователей различных компьютерных систем;
- интеллектуальные системы автоматизированного проектирования;
- интеллектуальные системы управления проектами.

Заключение

В предлагаемой технологии OSTIS существенными являются весь целостный комплекс этих принципов и их максимально возможная согласованность.

Ключевые проблемы, лежащие в основе предлагаемой технологии:

- обеспечение *семантической совместимости* (интегрируемости) различных моделей представления и обработки знаний;
- создание *общей теории* абстрактных семантических моделей интеллектуальных систем, не противопоставляя, а интегрируя самые различные подходы;
- обеспечение максимальной возможной *независимости* интеллектуальных систем от многообразия вариантов и платформ их технической реализации (в т. ч. и от будущих компьютеров, специально ориентированных на аппаратную поддержку обработки знаний).

Список литературы

1. Айзерман М.А. Динамический подход к анализу структур, описываемых графами (основы графодинамики) / М.А. Айзерман, Л.А. Гусев, С.В. Петров, И.М. Смирнова, Л.А. Тененбаум // Исследования по теории структур. М.: Наука, 1988. С. 5–76.
2. Амосов Н.М. Активные семантические сети в роботах с автономным управлением / Н.М. Амосов, А.М. Касаткин, Л.М. Касаткина // Труды IV Международной объединённой конференции по искусственному интеллекту: Т. 9. М., 1975.
3. Бениаминов Е.М. Основания категорного подхода к представлению знаний. Категорные средства / Е.М. Бениаминов // Изв. АН СССР. Техн. кибернет. 1988. № 2. С. 21–33.
4. Борщев В.Б. Схемы на клубных системах и вегетативная машина / В.Б. Борщев // Семиотика и информатика. 1983. Вып. 22. С. 3–44.
5. Берштейн Л.С. Использование расплывчатых ориентированных гиперграфов второго рода для представления фреймовых моделей / Л.С. Берштейн, Д.М. Башмаков // Методы построения алгоритмических моделей сложных систем. Таганрог, 1988. Вып. 7. С. 64–68.
6. Вагин, В.Н. Достоверный и правдоподобный вывод в интеллектуальных системах / В.Н. Вагин [и др.]. М.: ФИЗМАТЛИТ, 2008.
7. Вольфенгаген В.Э. Системы представления знаний с использованием семантических сетей / В.Э. Вольфенгаген, О.В. Воскресенская, Ю.Г. Горбанев // Вопросы кибернетики. Интеллектуальные банки данных. М.: АН СССР, 1979. С. 49–69.
8. Гаврилова Т.А. Базы знаний интеллектуальных систем / Т.А. Гаврилова, В.Ф. Хорошевский. СПб.: Питер, 2000.
9. Гастев Ю.А. Гомоморфизмы и модели. Логико-алгебраические аспекты моделирования / Ю.А. Гастев. М.: Наука, 1975.
10. Георгиев В.О. Модели представления знаний предметных областей диалоговых систем (обзор) / В.О. Георгиев. // Изв. АН СССР. Техн. кибернет. 1993. № 5. С. 24–44.

10. Гладун В.П. Процессы формирования новых знаний / В.П. Гладун. София: Педагог, 1994.
11. Голенков В.В. Представление и обработка знаний в графодинамических ассоциативных машинах / В.В. Голенков [и др.]. Мн.: БГУИР, 2001.
12. Голенков В.В. Графодинамические ассоциативные модели и средства параллельной обработки информации в системах искусственного интеллекта / В.В. Голенков, Н.А. Гулякина // Доклады БГУИР. 2004. № 1(5). С. 92–101.
13. Голенков В.В. Принципы построения массовой семантической технологии компонентного проектирования интеллектуальных систем / В.В. Голенков, Н.А. Гулякина // Материалы международной научно-технической конференции «Открытые семантические технологии проектирования интеллектуальных систем». Минск, 2011. С. 21–58.
14. Гостев Ю.Г. Описание семантики программ с помощью подстановок на графах // Программирование. 1981. № 5. С. 11–17.
15. Гостев Ю.Г. Описание структур данных с помощью графопорождающих грамматик / Ю.Г. Гостев // Программирование. 1981. № 2. С. 44–51.
16. Гуляева Д.М. Решение прикладных задач на расширенных семантических сетях. / Д.М. Гуляева // Математическое обеспечение ЭВМ и систем программирования. М., 1989.
17. Гулякина Н.А. Языки и технологии программирования, ориентированные на обработку семантических сетей / Н.А. Гулякина, Д.А. Лазуркин, О.В. Пивоварчик // Материалы международной научно-технической конференции «Открытые семантические технологии проектирования интеллектуальных систем». Минск, 2012.
18. Гусаков В.Я. Динамические алгебраические системы как математическая модель банка данных / В.Я. Гусаков, С.М. Гусакова // Семиотика и информатика. 1981. Вып. 17. С. 43–52.
19. Давыденко И.Т. Комплексная методика эволюционного коллективного проектирования семантических моделей интеллектуальных справочных систем / И.Т. Давыденко // Материалы международной научно-технической конференции «Открытые семантические технологии проектирования интеллектуальных систем». Минск, 2012.
20. Евгеньев Г.Б. Технология создания многоагентных прикладных систем / Г.Б. Евгеньев // Одиннадцатая национальная конференция по искусственному интеллекту: Труды конференции. Т. 2. М., 2008. С. 306–312.
21. Епифанов М.Е. Индуктивное обобщение в ассоциативных сетях / М.Е. Епифанов // Известия АН СССР. Техническая кибернетика. № 5, 1984. С. 132–146.
22. Ефимова С.М. П-графы для представления знаний / С.М. Ефимова. М: Вычислительный центр АН СССР, 1985.
23. Ефимова С.М. Поиск в базах знаний, опирающихся на модель П-графов, и его аппаратная реализация на основе метода M^3 / С.М. Ефимова, Е.В. Суворова. М: Вычислительный центр АН СССР, 1988.
24. Загоруйко Ю.А. Технология конструирования средств обработки знаний на основе семантических сетей. Средства спецификации и настройки / Ю.А. Загоруйко. Новосибирск, 1988.
25. Заливако С.С. Семантическая технология компонентного проектирования интеллектуальных решателей задач / С.С. Заливако, Д.В. Шункевич // Материалы международной научно-технической конференции «Открытые семантические технологии проектирования интеллектуальных систем». Минск, 2011.
26. Ивашенко В.П. Семантические модели и средства интеграции баз знаний / В.П. Ивашенко // Материалы международной научно-технической конференции «Открытые семантические технологии проектирования интеллектуальных систем». Минск, 2011.
27. Калиниченко Л.А. Методы и средства интеграции неоднородных баз данных / Л.А. Калиниченко. М.: Наука, 1983.
28. Кандрашина Е.Ю. Представление знаний о времени и пространстве в интеллектуальных системах / Е.Ю. Кандрашина, Л.В. Литвинцева, Д.А. Поспелов. М.: Наука, 1989.
29. Карабеков Б.А. Система «Бинарная Модель Знаний» как инструмент для концептуального моделирования бизнес-процессов // Одиннадцатая национальная конференция по искусственному интеллекту: Труды конференции. Т. 2. М., 2008. С. 282–291.

30. Клещев А.С. Семантические порождающие модели. Общая точка зрения на фреймы и продукции в экспертных системах / А.С. Клещев. Владивосток, 1986.
31. Колб Д. Г. Web-ориентированная реализация семантических моделей интеллектуальных систем / Д.Г. Колб // Материалы международной научно-технической конференции «Открытые семантические технологии проектирования интеллектуальных систем». Минск, 2011.
32. Колмогоров А.Н. К определению алгоритма / А. Н. Колмогоров // Успехи математических наук. 1958. Т. 13. № 4(82). С. 3–28.
33. Корончик Д.Н. Семантические модели мультимодальных пользовательских интерфейсов и семантическая технология их проектирования / Д.Н. Корончик // Материалы международной научно-технической конференции «Открытые семантические технологии проектирования интеллектуальных систем». Минск, 2011.
34. Котов В.Е. Асинхронные вычислительные процессы над общей памятью / В.Е. Котов, А.С. Нариньяни // Кибернетика. 1966. № 3. С. 64–71.
35. Касьянов В.Н. Графы в программировании: обработка, визуализация и применение / В.Н. Касьянов, В.А. Евстигнеев // ВHV — Санкт-Петербург, 2003.
36. Кузнецов В.Е. Представление в ЭВМ неформальных процедур / В.Е. Кузнецов. М.: Наука, 1989.
37. Кузнецов И.П. Семантические представления / И.П. Кузнецов. М.: Наука, 1986.
38. Лозовский В.С. Семантические сети / В. С. Лозовский // Представление знаний в человеко-машинных и робототехнических системах. М.: ВИНТИ, 1984. С. 84–121.
39. Лозовский В.С. Экстенциональная база знаний на основе семантических сетей / В.С. Лозовский // Известия АН СССР. Техн. кибернет. 1982. № 5. С. 23–42.
40. Любарский Ю.Я. Интеллектуальные информационные системы / Ю.Я. Любарский. М.: Наука, 1980.
41. Люгер Дж.Ф. Искусственный интеллект: стратегии и методы решения сложных проблем / Дж.Ф. Люгер. М.: Вильямс, 2003.
42. Мальцев А.И. Алгебраические системы / А.И. Мальцев. М.: Наука, 1970.
43. Марковский А.В. Анализ структуры знаковых ориентированных графов / А.В. Марковский. // Известия РАН: Теория и системы управления. № 5. 1997.
44. Мартынов В.В. Универсальный семантический код / В.В. Мартынов. Минск: Наука и техника, 1977.
45. Мельчук И.А. Опыт теории лингвистических моделей «Смысл-Текст». Семантика, синтаксис / И.А. Мельчук. М.: Наука, 1974.
46. Месарович М. Общая теория систем: математические основы / М. Месарович, Я. Тахакара. М.: Мир, 1978.
47. Молокова О.С. Методология анализа предметных знаний / О.С. Молокова. // Новости искусственного интеллекта. 1992. № 3. С. 11–60.
48. Нариньяни А.С. НЕ-факторы и инженерия знаний: от наивной формализации к естественной программатике / А.С. Нариньяни // КИИ-94. Сборник трудов Национальной конференции с международным участием по ИИ. «Искусственный интеллект-94»; в 2-х т. Т. 1. Тверь: АИИ, 1994. С. 9–18.
49. Осипов Г.С. Построение моделей предметных областей. Неординарные семантические сети / Г.С. Осипов. // Известия АН СССР. Техническая кибернетика. № 5. 1990.
50. Петров С.В. Графовые грамматики и автоматы (обзор) / С.В. Петров. // Автоматика и телемеханика. № 7. 1978. С. 116–136.

51. Петрушкин В.А. Экспертно-обучающие системы / В.А. Петрушкин. Киев: Наукова думка. 1992.
52. Плесневич Г.С. Денотационная семантика ассоциативных сетей / Г.С. Плесневич // Семантика и информатика. 1983. Вып. 21.
53. Плесневич Г.С. Представление знаний в ассоциативных сетях / Г.С. Плесневич // Изв. АН СССР. Техн. кибернет. 1982. № 5. С.6–22.
54. Плесневич Г.С. Бинарные модели знаний / Г.С. Плесневич // Труды Международных научно-технических конференций «Интеллектуальные системы» (AIS'08) и «Интеллектуальные САПР» (CAD-2008). Научное издание в 4-х томах. Т. 2. М: Физматлит, 2008.
55. Попков В.К. Гиперсети и их характеристики связности / В.К. Попков. // Исследования по прикладной теории графов. Новосибирск: Наука, 1986. С. 25–58.
56. Поспелов Д.А. Представление знаний. Опыт системного анализа / Д.А. Поспелов. // Системные исследования. Методологические проблемы. Ежегодник. М.: Наука, 1986. С. 83–102.
57. Поспелов Д.А. Ситуационное управление. Теория и практика / Д.А. Поспелов. М: Наука, 1986.
58. Рабинович З.Л. О концепции машинного интеллекта и её развитии / З.Л. Рабинович // Кибернетика и системный анализ. 1995. № 2. С. 163–173.
59. Рассел С. Искусственный интеллект: современный подход / С. Рассел, П. Норвиг. М.: Вильямс, 2006.
60. Резанов С.Н. Об одном методе обобщения на семантических сетях в системе управления энергообъединением / С.Н. Резанов // Изв. АН СССР. Техн. кибернет. 1989. № 5. С. 55–62.
61. Рубашкин В.Ш. Представление и анализ смысла в интеллектуальных информационных системах / В.Ш. Рубашкин. М.: Наука, 1989.
62. Рыбина Г.В. Основы построения интеллектуальных систем: учеб. пособ. / Г.В. Рыбина. М.: Финансы и статистика, 2010.
63. Сапатый П.С. Об эффективности структурной реализации операций над семантическими сетями / П.С. Сапатый // Техн. кибернет. 1983. № 5. С. 128–134.
64. Семёнов В.В. Семантические фреймовые сети как модели предметной области для САПР САУ / В.В. Сапатый // Представление знаний в системах искусственного интеллекта. М.: МДНТП, 1980. С. 117–122.
65. Скороходько Э.Ф. Семантические сети и автоматическая обработка текста. / Э.Ф. Скороходько. Киев: Наук. думка, 1983.
66. Скрэгг Г. Семантические сети как модели памяти / Г. Скрэгг // Новое в зарубежной лингвистике. Вып. 12. М.: Радуга, 1983. С. 228–271.
67. Соловьёв В.А. Формирование на семантической сети понятий и суждений с помощью рассуждений по аналогии / В.А. Соловьёв // II Всесоюзная конференция «Искусственный интеллект-90». Секционные и стендовые доклады. Минск, 1990. Т. 1. С. 166–169.
68. Тузов В.А. Математическая модель языка / В.А. Тузов. Л.: Изд-во ленингр. ун-та, 1984.
69. Тузов В.А. О формализации понятия задачи / В.А. Тузов. М.: Наука, 1986. С. 73–83.
70. Тыгу Э.Х. Интеграция знаний / Э.Х. Тыгу // Изв. АН СССР. Техн. кибернет. 1989. № 5. С. 3–13.
71. Финн В.К. Многозначные логики и их применения / ред. В.К. Финн. М.: ЛКИ, 2008. Т. 1, Т.2.
72. Уварова Т.Г. Формальное описание операционного языка для семантических сетей. / Т.Г. Уварова, Л.Л. Лифшиц. М.: ВЦ АН СССР, 1987.
73. Хельбиг Г. Семантическое представление знаний в вопросно-ответной системе FAS-80 / Г. Хельбиг. // Представление знаний и моделирование процессов понимания. Новосибирск, 1980. С. 97–123.
74. Хендрикс Г. О расширении применимости семантических сетей введением разбиений / Г.О. Хендрикс // Труды IV Международной объединённой конференции по искусственному интеллекту. М., 1975. Т. 1. С. 190–206.

75. Хорошевский В.Ф. Пространства знаний в сети Интернет и Semantic Web (Часть 1) / В.Ф. Хорошевский. // Искусственный интеллект и принятие решений, 2008, № 1. С. 80–97.
76. Цаленко М.Ш. Моделирование семантики в базах данных. / М.Ш. Цаленко. М.: Наука, 1989.
77. Шенк Р. Обработка концептуальной информации / Р. Шенк. Москва: Энергия, 1980.
78. Шрейдер Ю.А. Системы и модели / Ю.А. Шрейдер, А.А. Шаров. М.: Радио и связь, 1982.
79. Шуберт Л. Усиление выразительной мощности семантических сетей / Л. Шуберт // Кибернетический сборник. Новая серия. 1979. Вып. 16. С. 171–212.
80. Naidenova X. Machine Learning Methods for Commonsense Reasoning Processes: Interactive Models (Premier Reference Source)/ Xenia Naidenova // Information Science Reference; 1 edition (October 31, 2009).
81. Открытая семантическая технология проектирования интеллектуальных систем [Электронный ресурс]. 2011. Режим доступа: <http://ostis.net>
82. Russell S. Artificial Intelligence. A Modern Approach / S Russell, P Norvig. New Jersey: Prentice Hall, 1995.
83. Sowa J. Semantic networks / John F. Sowa // Encyclopedia of Artificial Intelligence, edited by S. C. Shapiro. New York: Wiley, 1992.
84. Sowa J. Conceptual Graphs/ John F. Sowa, F. van Harmelen, V. Lifschitz, B. Porter// eds., Handbook of Knowledge Representation, Elsevier, 2008, pp. 213–237.
85. Wooldridge M. Agent Theories, Architectures and Languages: A Survey / M. Wooldridge, N. Jennings // Intelligent Agents. Languages. Amsterdam: Springer Verlag, August, 1994. P. 3–39.

Сведения об авторах

Голенков В.В. —

доктор технических наук, профессор, заведующий кафедрой «Искусственный интеллект» Белорусского государственного университета информатики и радиоэлектроники. Golen@bsuir.ru

Гулякина Н.А. —

кандидат физико-математических наук, доцент, заместитель заведующего кафедрой «Искусственный интеллект» Белорусского государственного университета информатики и радиоэлектроники. Guliakina@bsuir.ru

Фильтрация речи на фоне полигармонических и стохастических помех

Петровский А.А., доктор технических наук, профессор,

Азаров И.С., кандидат технических наук,

Лихачёв Д.С., кандидат технических наук, доцент,

Ромашкин Ю.Н., кандидат технических наук,

Жигулёвцев Ю.Н., кандидат технических наук, доцент,

Харламов А.А., доктор технических наук

В статье исследуются алгоритмы выделения речи на фоне аддитивных помех с полигармоническими и шумовыми составляющими. Сравниваются гибридный алгоритм (субполосная фильтрация — спектральное вычитание) и фильтрация в модуляционной области. По методу SII оценивается разборчивость речи, обеспечиваемая каждым из рассмотренных алгоритмов фильтрации. Даются выводы о наиболее эффективном алгоритме в зависимости от типа помехи.

• речевой сигнал • подавление шума • полигармоническая и стохастическая помехи.

The paper analyses different noise reduction algorithms for speech corrupted by additive noises with polyharmonic and stochastic components. A hybrid noise reduction algorithm based on subband filtering and spectral weighting, compares with speech filtering in modulation domain. The speech intelligibility is evaluated using SII measure for each considered technique. The paper discusses the choice of the most suitable processing algorithm depending on the noise properties.

• speech signal • noise reduction • polyharmonic and stochastic noise.

Введение

Широкий класс акустических помех может быть представлен в виде комбинации стохастической (шумовой) составляющей с гладким спектром и детерминированной (периодической) составляющей, спектр которой имеет ряд гармоник при медленно меняющейся основной частоте. Источником такого рода помех может быть электрическая сеть с переменным напряжением, трансформаторы, электродвигатели и т.д.

Большинство классических методов шумоподавления рассматривают шум как случайный сигнал с некоторыми априорно известными параметрами во временной или частотной области. Наиболее популярным является метод спектрального вычитания [1], где спектр шума оценивается в паузах речи и вычитается из спектра принятой аддитивной смеси. В случае же, когда помеха представляет собой сумму стохастических и полигармониче-

ских составляющих, оценка спектра шума часто не может быть выполнена достаточно точно, а речь — отфильтрована без слышимых артефактов. Данный эффект обусловлен следующими особенностями. С одной стороны, для подавления узкополосных гармоник оценка спектра должна проводиться с достаточно высоким частотным разрешением. С другой стороны, ослабление шумовой компоненты требует сглаженной оценки спектра, чтобы исключить появления музыкальных тонов в результате фильтрации. Необходимость выполнить одновременно оба этих требования сильно ограничивает подавление таких комбинированных помех алгоритмами на основе метода спектрального вычитания. Логичное решение данной проблемы — применение гибридной схемы с отдельной обработкой полигармонической и стохастической составляющих помехи.

Подавление квазипериодической компоненты помехи требует применения специальных методов, учитывающих её основные свойства:

- локализацию мощности в ряде узких частотных полос;
- наличие множества гармоник основной частоты;
- медленное изменение основной частоты во времени.

Учитывая эти свойства, можно предположить, что эффективными подходами для подавления полигармонической помехи могут быть адаптивная и узкополосная фильтрации. Эффективность первой объясняется высокой коррелированностью помехи, вследствие чего её легко предсказывать по предыдущим отсчётам. В гибридной системе шумоподавления оставшаяся стохастическая составляющая помехи может быть подавлена, например, при помощи традиционного метода спектрального вычитания либо при помощи обработки сигнала в подпространствах [2].

Ещё одним заслуживающим внимания методом подавления комбинированных стохастических/полигармонических помех является фильтрация в модуляционной области [3]. Общий алгоритм такой обработки сводится к субполосному разделению речевого сигнала, выделению огибающих в каждой полосе, фильтрации каждой огибающей полосовым (модуляционным) фильтром и обратному синтезу речевого сигнала. Фильтрация в модуляционной области может быть так же эффективна, поскольку отдельные составляющие помехи и речи занимают различные частотные полосы в модуляционном спектре.

В статье исследуются и сравниваются по разборчивости отфильтрованной речи различные методы подавления полигармонических/стохастических помех.

1. Подавление полигармонической помехи субполосным адаптивным фильтром

Для подавления узкополосных тональных составляющих полигармонической помехи могут быть использованы различные подходы. Простым в реализации и достаточно эффективным является способ, основанный на предсказании текущих отсчётов коррелированной помехи по предыдущим отсчётам с помощью стандартного адаптивного фильтра [4]. Для более полного подавления полигармонической помехи в схему должен быть включён банк полосовых фильтров (рис. 1).

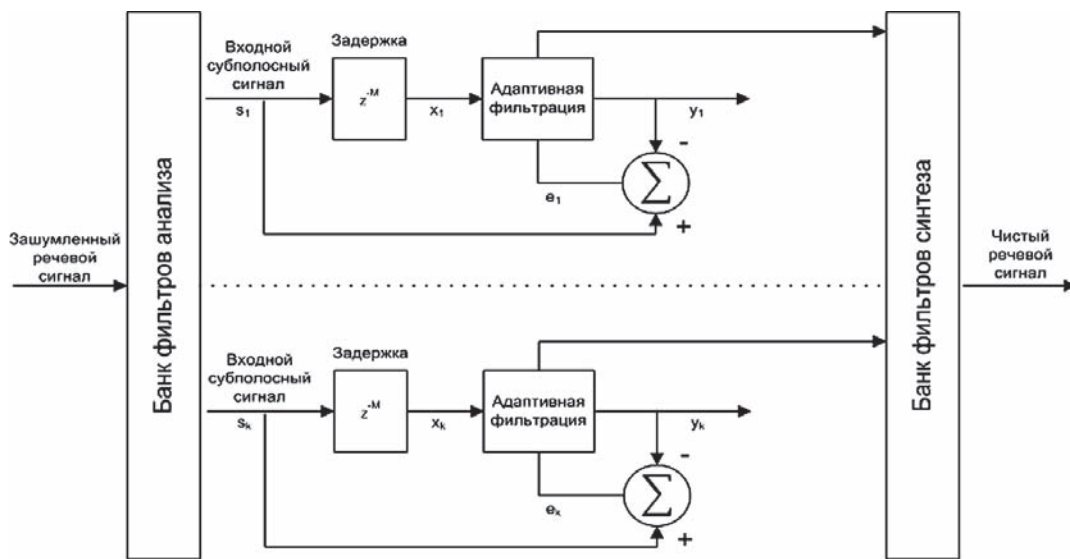


Рис. 1. Схема подавления полигармонической помехи с помощью алгоритма субполосной адаптивной фильтрации

Настройка коэффициентов адаптивного фильтра осуществляется в соответствии со следующими уравнениями [4]:

$$Y_k = \sum_{n=0}^{N-1} \omega_k(n) x_k(n), k = \overline{1, K}$$

$$e_k = s_k - y_k$$

$$\omega_{k+1}(n) = \omega_k(n) + 2\mu e_k x_k(n) = \overline{0, N-1},$$

где s_k — зашумленный сигнал в k -й полосе частот;

$x_k(n)$ — задержанный отсчёт s_k ;

$\omega_k(n)$ — коэффициенты фильтра;

μ — коэффициент скорости сходимости;

y_k — выделенный узкополосный тональный сигнал;

K — количество полосовых фильтров.

Пример фильтрации речевого сигнала, принятого на фоне полигармонической сетевой помехи (с отношением сигнал/шум — 20 дБ), с помощью данного алгоритма представлен на рис. 2 (стр. 48). В алгоритме применялись банк узкополосных фильтров шириной 80 Гц, длительность задержки 100 мс и порядок адаптивного фильтра $N = 100$.

В отфильтрованном сигнале полигармоническая помеха ослаблена более чем на 20 дБ при слабых искажениях речи. Данный способ подавления полигармонической помехи является весьма удобным в реализации, если учесть, что при незначительной модификации схемы обработки субполосовые сигналы могут быть децимированы и, тем самым, существенно снижен требуемый порядок адаптивных фильтров.

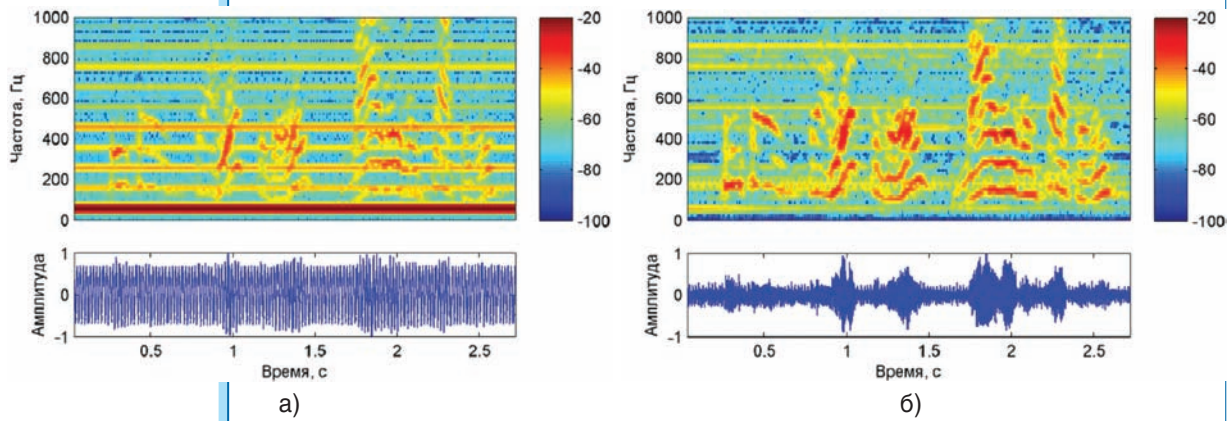


Рис. 2. Обработка речевого сигнала на фоне сетевой помехи (ОСШ = -20 дБ):
 а) исходный зашумлённый сигнал;
 б) выходной сигнал субполосного адаптивного фильтра

2. Подавление интенсивной полигармонической помехи

Для обеспечения большего подавления полигармонической помехи предлагается следующая схема обработки речевого сигнала (рис. 3).

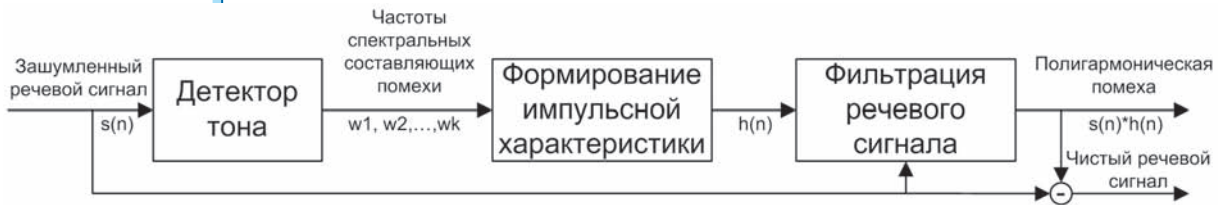


Рис. 3. Схема подавления интенсивной полигармонической помехи

Исходный сигнал обрабатывается детектором тона, выделяющим частоту каждой гармоники помехи. Затем в соответствии с найденными частотами формируется импульсная характеристика фильтра, пропускающего только полигармоническую помеху. Оценка речевого сигнала получается путём вычитания оценки помехи из принятого зашумленного сигнала. Импульсная характеристика фильтра обновляется в соответствии с изменениями параметров помехи.

Выделение тональных составляющих помехи в детекторе должно быть выполнено в соответствии со следующими требованиями:

- высокое частотное разрешение (до 1 Гц);
- нечувствительность детектора к гармоническим компонентам вокализованной речи;
- адаптация к медленным изменениям основной частоты помехи.

Для оценки текущего спектра в M заданных частотах $\omega_1, \omega_2, \dots, \omega_M$ отсчёт сигнала $s(n)$ умножается на вектор из M комплексных экспонент $(e^{-j\varphi_1}, e^{-j\varphi_2}, \dots, e^{-j\varphi_M})^T$, фазы которых $\varphi_1, \varphi_2, \dots, \varphi_M$ увеличиваются в соответствии с заданными частотами, где T — транспонирование. В результате формируется оценка кратковременного спектра в виде вектора отсчётов $(s_1, s_2, \dots, s_M)^T$,

по которому вычисляется среднее значение вектора спектральных амплитуд. Далее проводится оценка среднего спектра с использованием экспоненциального усреднения с коэффициентами $0 < \lambda_1 < 1$ (рис. 4).

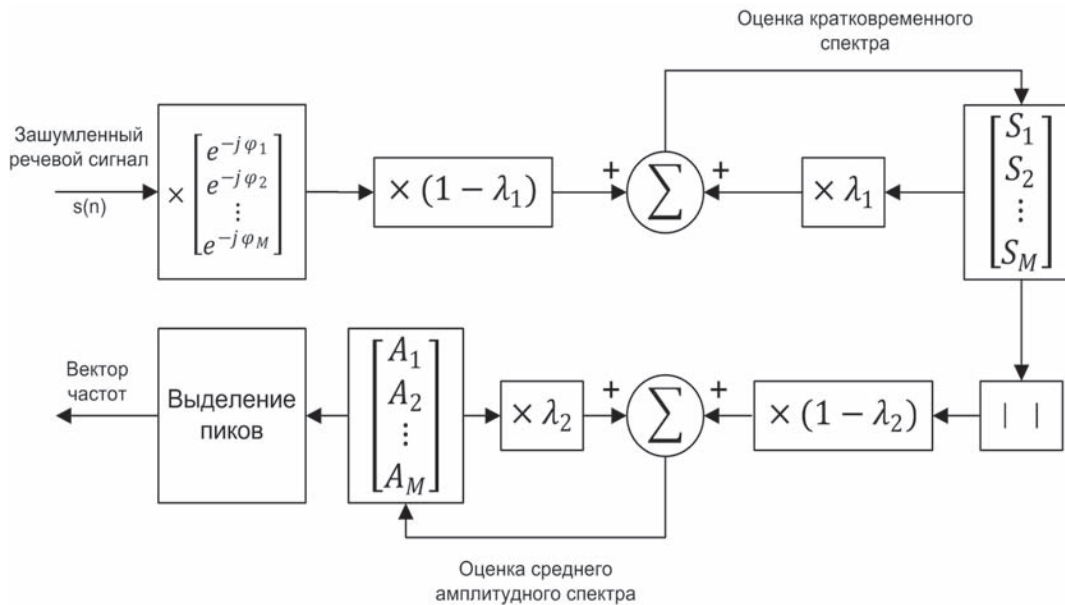


Рис. 4. Определение тональных компонент помехи

С помощью данной схемы можно получить оценку среднего спектра с требуемым частотным разрешением и длиной временного окна, регулируя значения коэффициентов. По среднему спектру вычисляются локальные максимумы, превышающие заданный порог (экспериментально определено, что в зависимости от типа помехи он составляет от 5 до 10 дБ).

В соответствии с предложенной схемой (рис. 3) блок фильтрации речевого сигнала, выполняющий выделение полигармонической помехи, должен пропускать лишь те частоты, которые выделены детектором тона, и задерживать все остальные. Удобным для этого является субполосный подход, поскольку общая импульсная характеристика фильтра может быть получена из соответствующих характеристик полосовых фильтров. Для увеличения скорости обработки полосовые фильтры предварительно синтезируются для каждой частоты $\omega_1, \omega_2, \dots, \omega_M$ и хранятся в оперативной памяти в процессе фильтрации.

Важным вопросом является выбор оптимальных значений для полосы пропускания (Δf) полосовых фильтров и длины импульсной характеристики. Расширение полосы пропускания способствует большему подавлению помехи, но в то же время увеличивает и подавление речевого сигнала. Увеличение длины импульсной характеристики позволяет уменьшить переходную полосу фильтра, но в то же время приводит к росту искажений сигнала.

Определение оптимальных значений выполнено опытным путём. К речевому сигналу (частота дискретизации 8 кГц) аддитивно добавлялась полигармоническая сетевая помеха с отношением сигнал/шум ОСШ = -20 дБ. Затем проводилась фильтрация сигнала предложенным алгоритмом (далее — алгоритм «TRem») и оценка качества обработанного сигнала по методу PESQ. Полученные результаты приведены в таблице 1 для различных значений Δf и длины импульсной характеристики фильтра.

Таблица 1

Зависимость оценки PESQ от ширины полосы пропускания и длины импульсной характеристики фильтра

Δf , Гц	Длина импульсной характеристики, отсчёты							
	512	1024	1536	2048	2560	3072	3584	4096
4	1,34	1,41	1,48	1,57	1,67	1,78	1,88	1,97
7	1,37	1,52	1,70	1,93	2,14	2,27	2,32	2,32
10	1,40	1,66	2,00	2,28	2,37	2,36	2,34	2,33
13	1,44	1,84	2,26	2,37	2,35	2,33	2,32	2,3
16	1,48	2,03	2,37	2,34	2,32	2,3	2,28	2,27
19	1,53	2,19	2,34	2,31	2,29	2,27	2,26	2,25
22	1,58	2,29	2,31	2,28	2,26	2,24	2,23	2,21
25	1,64	2,30	2,28	2,25	2,23	2,21	2,2	2,18
28	1,70	2,27	2,24	2,21	2,19	2,17	2,16	2,14

Исходя из полученных результатов, оптимальная ширина полосовых фильтров составила 13 Гц, а длина импульсной характеристики фильтра — 2048.

На рис. 5 приведены спектрограммы и временные реализации исходного речевого сигнала на фоне сетевой помехи с ОСШ = -30 дБ и результата фильтрации алгоритмами «TRem» и спектрального вычитания. Анализ экспериментальных результатов показывает, что предложенный алгоритм «TRem» обеспечивает ослабление сетевой помехи более чем на 30 дБ и заметно меньшее искажение речевого сигнала.

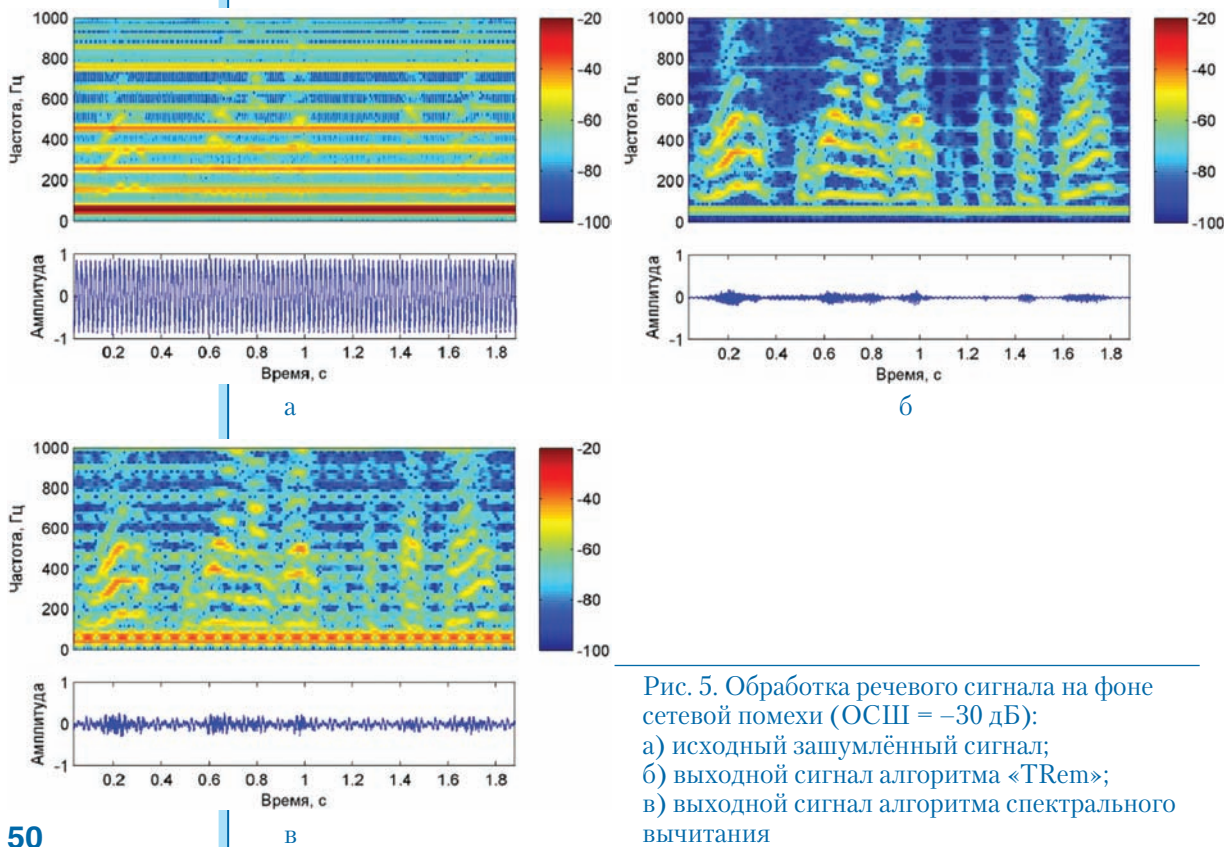


Рис. 5. Обработка речевого сигнала на фоне сетевой помехи (ОСШ = -30 дБ):
 а) исходный зашумлённый сигнал;
 б) выходной сигнал алгоритма «TRem»;
 в) выходной сигнал алгоритма спектрального вычитания

3. Гибридная фильтрация

Для подавления аддитивных помех, содержащих полигармонические и стохастические составляющие, может применяться последовательная схема обработки: сначала подавление полигармонических помех алгоритмом «TRem», а затем ослабление шумовой составляющей с помощью алгоритма спектрального вычитания. Результаты фильтрации речевого сигнала, принятого на фоне акустического шума пылесоса, показаны на рис. 6 а, б, в, г отдельно для метода спектрального вычитания, алгоритма «TRem» и гибридного метода соответственно. Из приведённых рисунков видно, что алгоритм спектрального вычитания не обеспечивает достаточного подавления периодических составляющих помехи. Гибридный метод позволяет получить более высокое подавление помехи при незначительном уровне искажений отфильтрованного речевого сигнала.

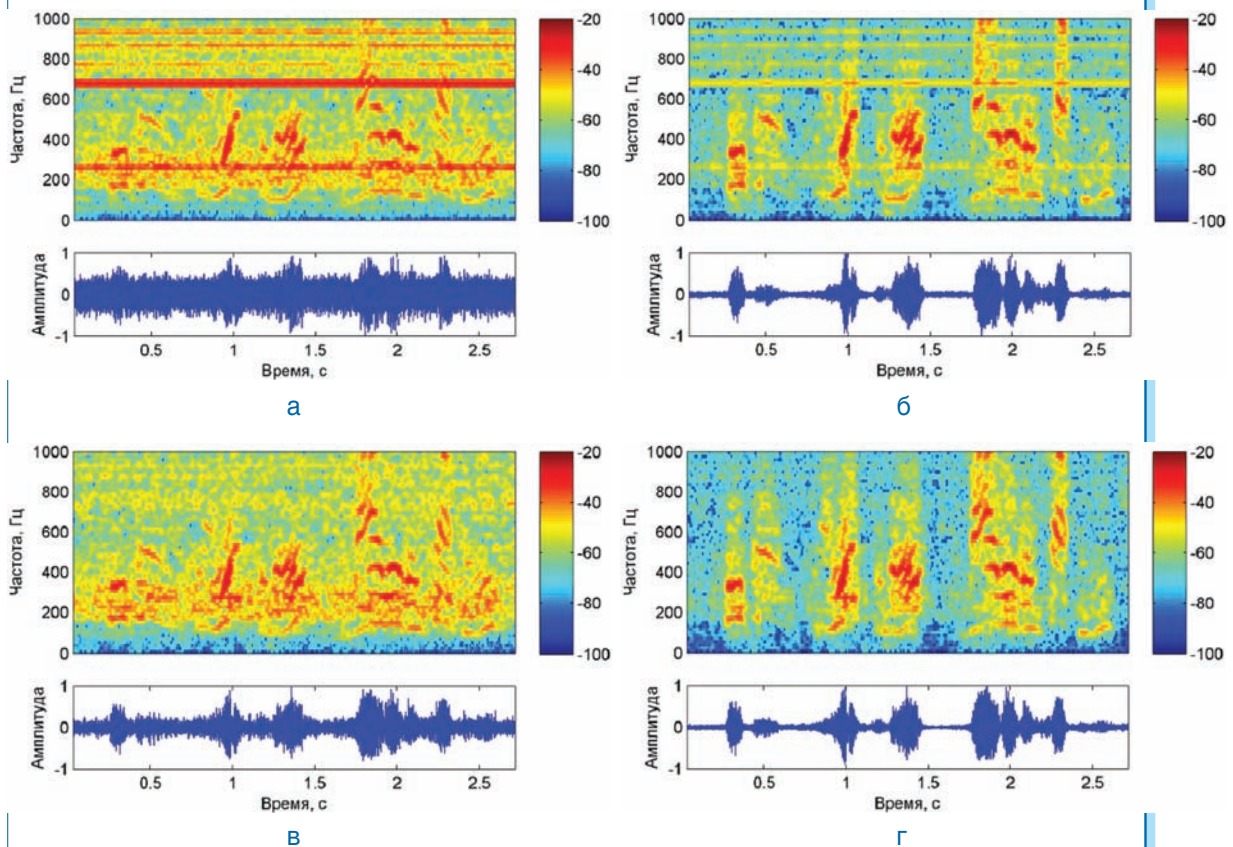


Рис. 6. Обработка речевого сигнала на фоне шума пылесоса (ОСШ = -10 дБ):

- исходный зашумлённый сигнал;
- выходной сигнал алгоритма спектрального вычитания;
- выходной сигнал алгоритма «TRem»;
- выходной сигнал гибридного алгоритма

4. Фильтрация в модуляционной области

Подход к обработке речи в модуляционной области появился сравнительно недавно и подтвердил информационную значимость амплитудных модуляций [5]. На основании ряда наблюдений было сделано предположение, что слуховой аппарат человека распознаёт речь при помощи частотно-временного анализа амплитудных модуляций

речи. В работе [6] показана тесная связь классических психоакустических принципов частотного и временного маскирования с аналогичными эффектами в модуляционном спектре.

Известны два метода обработки зашумленной речи: фильтрация модуляционного спектра — RASTA (RelATive SpecTrA) и перцептуальное линейное предсказание — PLP (Perceptual linear prediction). В целом алгоритм на основе RASTA-PLP состоит из следующих шагов [6]:

- вычисление энергии сигнала в критических частотных полосах;
- компрессия амплитудного спектра путём нелинейной трансформации;
- фильтрация временной последовательности каждой из частотных компонент;
- растягивание амплитудного спектра путём обратной нелинейной трансформации
- умножение на кривую равной громкости для выравнивания громкости.

Ключевой идеей является подавление постоянных составляющих в кратковременном спектре принятого сигнала.

Для выполнения нелинейной трансформации обычно используются логарифмические функции. Они приводят к тому, что искажения, свёрнутые во временной области, проявляются как аддитивный шум в логарифмическом спектре. Трудности возникают, если помеха является некоррелированным аддитивным шумом. Такой шум аддитивен в линейном спектре, но в логарифмическом становится зависим от самого сигнала, что делает невозможным его удаление путём фильтрации в частотных полосах. Решение этой проблемы было предложено в [7] и заключается в использовании следующей функции преобразования:

$$Y(k, n\Delta t) = \ln(1 + cX(k, n\Delta t)),$$

где $X(k, n\Delta t)$ — амплитуда входного сигнала в k -й полосе частот, c — положительная константа, зависящая от сигнала.

Такое преобразование линейно для малых значений амплитуды, для больших — логарифмическое. Формула обратного преобразования имеет вид:

$$Y(k, n\Delta t) = \frac{e^{X(k, n\Delta t)} - 1}{c},$$

В качестве фильтра, используемого для обработки сигнала в модуляционной области, авторами RASTA был предложен БИХ фильтр с передаточной функцией [6]:

$$H(z) = 0.1z^4 \frac{2 + z^{-1} - z^{-3} - 2z^{-4}}{1 - 0.98z^{-1}}.$$

Нижняя граница полосы пропускания фильтра определяет максимальную скорость изменения логарифмического спектра, подавляемую фильтром. Верхняя граница определяет максимальную скорость изменения сохраняемого спектра. Начиная с частоты 12 Гц, амплитудно-частотная характеристика фильтра имеет крутизну спада 6 дБ/окт., достигая нуля при частоте 30 и 50 Гц.

Ниже приведены результаты обработки речи на фоне различных помех, полученные с помощью алгоритма фильтрации в модуляционной области. На рис. 7а, б приведён пример обработки аддитивной смеси речевого сигнала с белым шумом. Подавление шума составило в среднем 8 дБ.

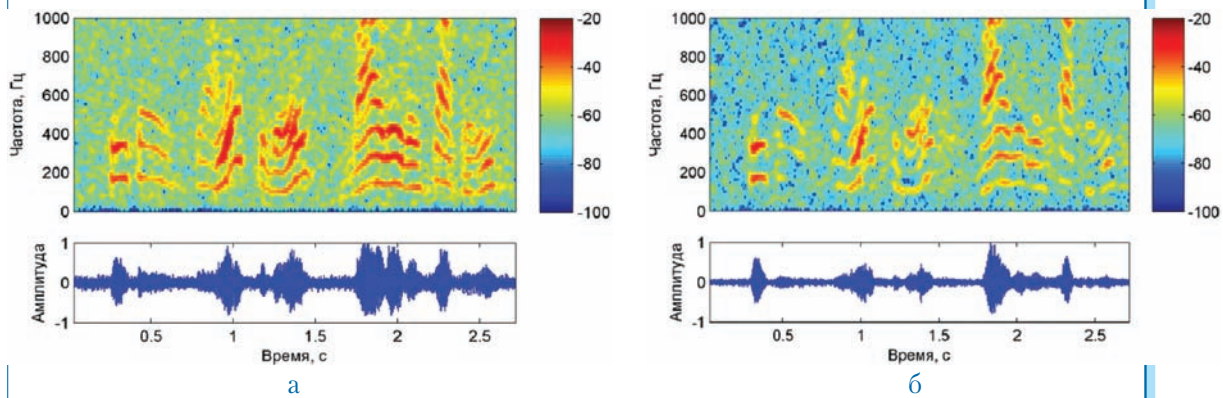


Рис. 7. Обработка речевого сигнала на фоне белого шума (ОСШ = -5 дБ):
а) исходный зашумлённый сигнал;
б) выходной сигнал модуляционного фильтра

Сетевая помеха, как отмечалось ранее, содержит множество гармоник с медленно изменяющимися амплитудами и потому может эффективно подавляться модуляционным фильтром. Пример обработки речевого сигнала на фоне сетевой помехи приведён на рис. 8а, б. На участках, где отсутствует голос диктора, снижение энергии сетевой помехи составляет, примерно, 33 дБ.

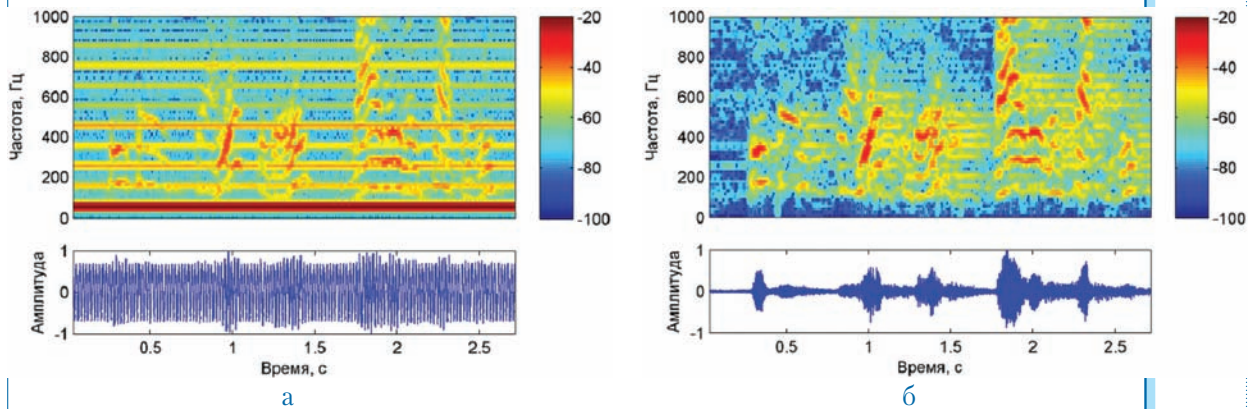


Рис. 8. Обработка речевого сигнала на фоне сетевой помехи (ОСШ = -20 дБ):
а) исходный зашумлённый сигнал;
б) выходной сигнал модуляционного фильтра

Акустический шум работающего пылесоса состоит из периодических и стохастических составляющих при его общей нестационарности. На рис. 9а, б (стр. 54) приведён пример обработки речевого сигнала на фоне шума пылесоса. В паузах речи подавление помехи составляет примерно 22 дБ.

Фильтрация речи на фоне музыкальной помехи одноканальными системами — крайне сложная задача. Тем не менее, применение модуляционного фильтра обеспечивает некоторое подавление тональных и переходных составляющих, характерных для музыкальных инструментов. Пример обработки речевого сигнала на фоне музыки приведён на рис. 10 (стр. 54). Снижение энергии музыки составило примерно 4 дБ.

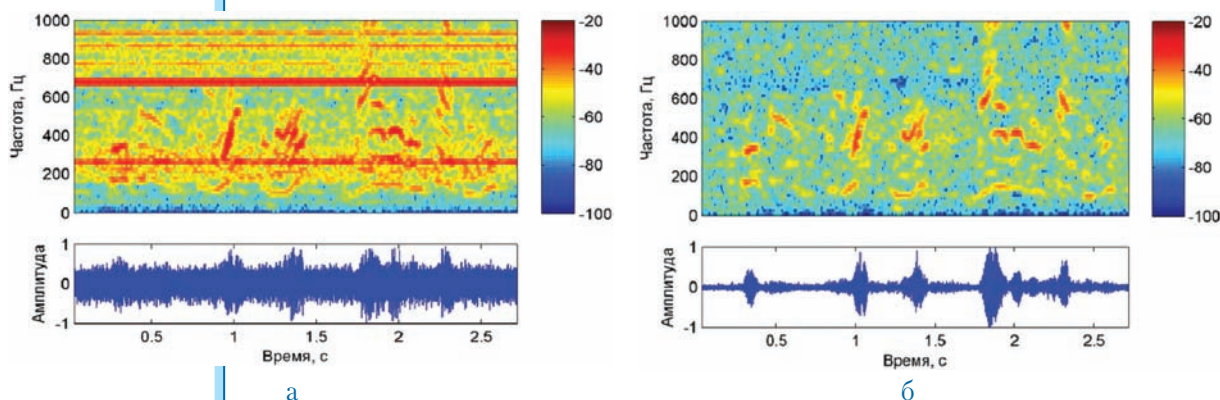


Рис. 9. Обработка речевого сигнала на фоне шума пылесоса (ОСШ = -10 дБ):
а) исходный зашумлённый сигнал;
б) выходной сигнал модуляционного фильтра

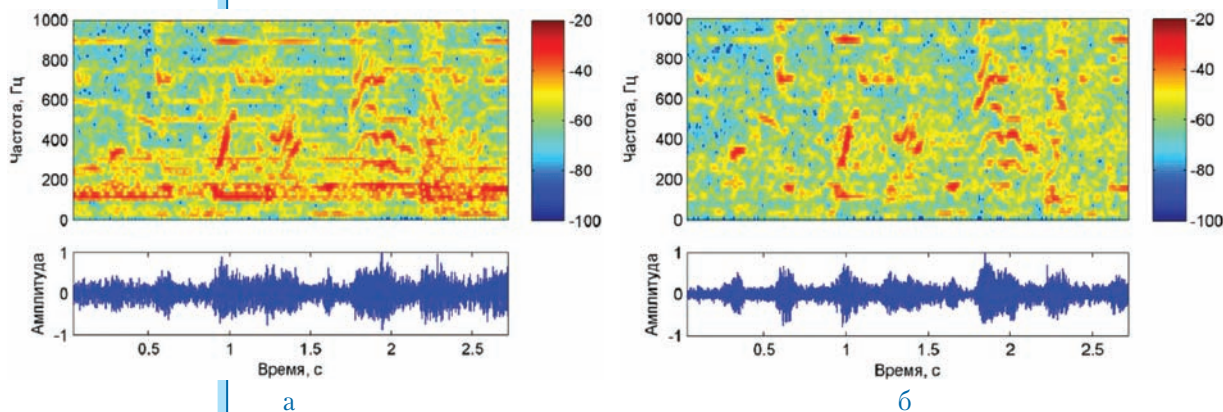


Рис. 10. Обработка речевого сигнала на фоне музыки (ОСШ = -5 дБ):
а) исходный зашумлённый сигнал;
б) выходной сигнал модуляционного фильтра

5. Оценка разборчивости речи

Оценка разборчивости по методу SII (speech intelligibility index) основана на вычислении т.н. индекса артикуляции, значения которого принято интерпретировать следующим образом [8]:

- ниже 0,3 — плохая разборчивость;
- от 0,3 до 0,5 — удовлетворительная;
- от 0,5 до 0,7 — хорошая;
- выше 0,7 — очень хорошая.

Для сравнения разборчивости отфильтрованной речи обрабатывались аддитивные смеси речевого сигнала (РС) на фоне трёх помех: белого шума, сетевой помехи и шума пылесоса с ОСШ от -30 до 20 дБ. Эти смеси обрабатывались следующими алгоритмами: спектрального вычитания (NRS), алгоритмом «TRem», фильтром в модуляционной области («Modul»), а также алгоритмом обработки сигнала в подпространствах (perceptually constrained signal subspace — PCSS, [2]). Полученные результаты усреднялись по мужским и женским голосам длительностью около 2-х минут. На рис. 12–14 показаны экспериментальные зависимости индекса SII от ОСШ.

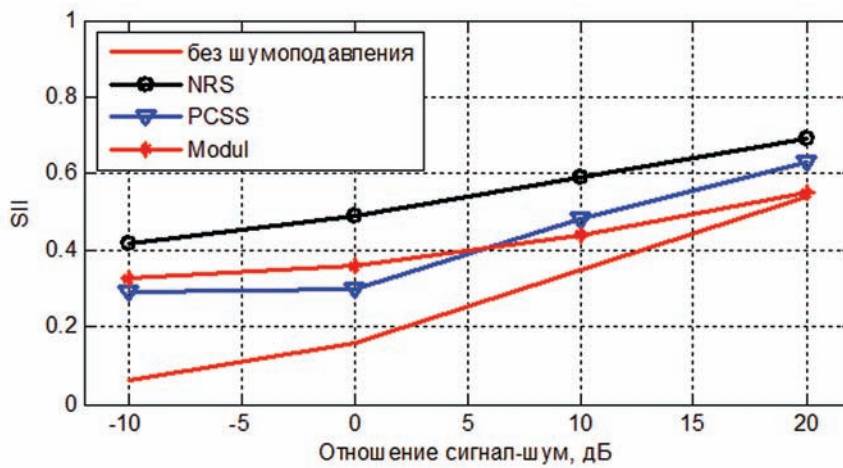


Рис. 11. Зависимость индекса SII от ОСШ для смеси PC с белым шумом

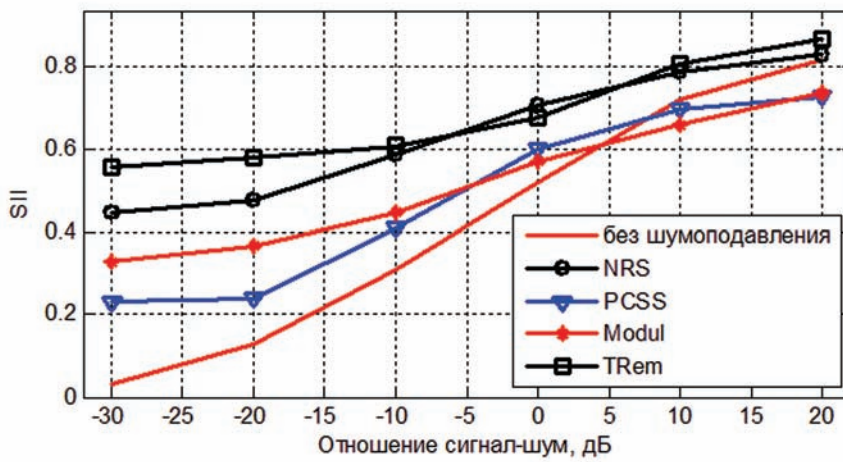


Рис. 12. Зависимость индекса SII от ОСШ для смеси PC с сетевой помехой

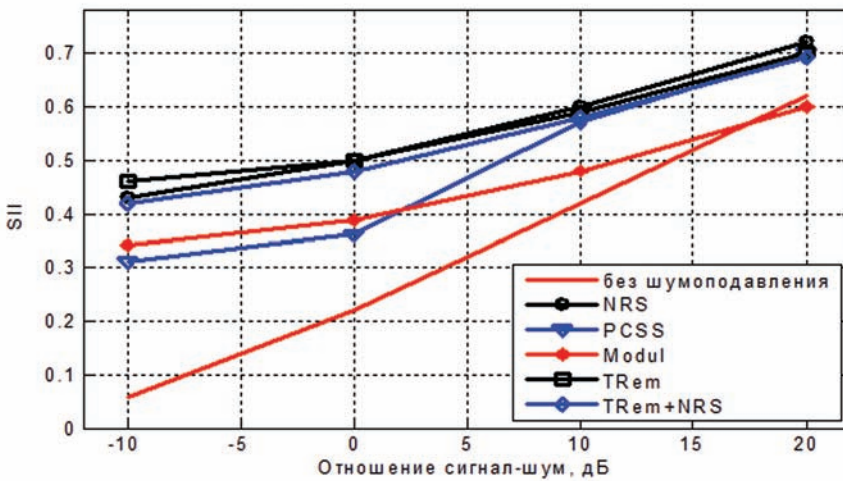


Рис. 13. Зависимость индекса SII от ОСШ для смеси PC с шумом пылесоса

Как видно из представленных результатов, при наличии белого шума алгоритмы спектрального вычитания, модуляционной фильтрации и PCSS позволяют заметно повысить разборчивость речи. Причём метод спектрального вычитания является предпочтительным при любом ОСШ.

Для случая воздействия полигармонической сетевой помехи алгоритм «TRem», предложенный в статье, позволяет достичь значительно большей разборчивости по сравнению с остальными алгоритмами.

При обработке речи на фоне акустического шума пылесоса, имеющего полигармонические и шумовые составляющие, эффективность метода PCSS значительно снижается при отрицательных ОСШ, в то время как алгоритмы «TRem» и спектрального вычитания заметно повышают разборчивость речи. Гибридный алгоритм «TRem+NRS» практически эквивалентен по разборчивости речи алгоритмам, его составляющим.

Алгоритм на основе модуляционной фильтрации в целом показывает средние результаты.

Выводы

Широко известные алгоритмы шумоподавления, такие как спектральное вычитание и субполосная адаптивная фильтрация, не обеспечивают наилучший результат в случае воздействия сложных полигармонических/стохастических помех.

Экспериментально доказано, что предложенный алгоритм подавления интенсивной полигармонической помехи обеспечивает хорошую разборчивость речи даже при ОСШ < -10 дБ.

Метод фильтрации в модуляционной области обеспечивает удовлетворительную разборчивость при различных видах помех. В частности, он подавляет белый шум на 8дБ, комбинированную полигармоническую/стохастическую помеху более чем на 20 дБ. Однако он уступает по разборчивости речи гибриднему алгоритму.

Список литературы

1. Loizou P. Speech Enhancement: Theory and Practice / P. Loizou. Boca Raton, FL: CRC Press, 2007.
2. Borowicz A., Petrovsky A. Signal subspace approach for psychoacoustically motivated speech enhancement. Speech communication 53 (2011), Elsevier. P.210–219.
3. Bashun J., Petrovsky A. Speech enhancement for cochlear implants based on the reducing slow temporal modulations // Proc. of The Acoustic Congress, Rom, Italy, 2000.
4. Haykin S. Adaptive filter theory, NJ.: Prentice — Hall, 1996.
5. Kowalski N., Depireux D. and Shamma S. Analysis of dynamic spectra in ferret primary auditory cortex: I. Characteristics of single unit responses to moving ripple spectra // Journal of Neurophysiology, 1996. Vol. 76, no. 5. P. 3503–3523.
6. Hermansky H. and Morgan N. RASTA processing of speech. IEEE Transaction on Speech and Audio Processing, 1994. Vol. 2, no. 4. P. 587–589.
7. Avendano Temporal processing of speech in a Time-Feature Space, Ph.D. thesis, Oregon Graduate Institute, April, 1997.
8. Основы психоакустики. Субъективные и объективные методы оценки разборчивости речи [Электронный ресурс] / И. Алдошина. Журнал «Звукорежиссёр», 2002. Режим доступа: www.kadva.ru/files/edu/AldoshinaPsychoacoustics.pdf — Дата доступа: 16.03.2011.

Сведения об авторах

Петровский Александр Александрович —

доктор технических наук, профессор. Работает в Учреждении образования «Белорусский государственный университет информатики и радиоэлектроники», кафедра «Электронные вычислительные средства». Закончил Учреждение образования «Белорусский государственный университет информатики и радиоэлектроники» по специальности «Электронные вычислительные машины». Главные научные интересы лежат в области цифровой обработки сигналов речи и звука для целей компрессии, распознавания, редактирования шума, а также в области проектирования проблемно-ориентированных средств вычислительной техники реального времени для систем мультимедиа. Член НТО РЭС им. А.С. Попова, IEEE, EURASIP, AES.

Азаров Илья Сергеевич —

кандидат технических наук. Учреждение образования «Белорусский государственный университет информатики и радиоэлектроники» (БГУИР), кафедра электронных вычислительных средств. Закончил Белорусский государственный университет, механико-математический факультет, отделение математической электроники. Область научных интересов: цифровая обработка речевых сигналов.

Лихачёв Денис Сергеевич —

кандидат технических наук, доцент. Учреждение образования «Белорусский государственный университет информатики и радиоэлектроники» (БГУИР), кафедра электронных вычислительных средств, г. Минск. Закончил БГУИР, факультет компьютерного проектирования. Область научных интересов: цифровая обработка речевых сигналов, системы компрессии речи, антропоморфическая обработка речи, конверсия голоса.

Ромашкин Юрий Николаевич —


кандидат технических наук.

Жигулёвцев Юрий Николаевич —

кандидат технических наук, старший научный сотрудник, доцент МГТУ им. Н.Э.Баумана. Окончил в 1969 г. МГТУ им. Н.Э.Баумана по специальности «Системы автоматического управления». Автор более 80 научных публикаций, 6 авторских свидетельств на изобретения, соавтор 2 монографий. Область научных интересов: методы и средства построения систем речевого взаимодействия.

Харламов Александр Александрович —

доктор технических наук, старший научный сотрудник Института высшей нервной деятельности и нейрофизиологии РАН. Область научных интересов: нейроинформатика, распознавание речи, анализ текстов, распознавание изображений, семантические представления, искусственные нейронные сети.



Гибридные модели – Скрытые марковские модели/ Многослойный персептрон – и их применение в системах распознавания речи. Обзор

*Маковкин К.А., научный сотрудник
Вычислительного центра
им. А.А. Дородницына РАН*



Большинство современных систем автоматического распознавания речи (АРР) построено на основе скрытых марковских моделей (СММ). Использование СММ предоставляет мощный и гибкий инструмент для разработки систем АРР, а также эффективно по многим другим критериям. Однако метод оценки эмиссионных вероятностей, который используется в СММ, обладает рядом очень важных ограничений, что сильно усложняет практическую реализацию и использование систем АРР в реальной обстановке.

На рубеже 80-х и 90-х годов прошлого века рядом исследователей был предложен и исследован новый подход, заключающийся в комбинировании СММ и многослойного персептрона (МСП) в рамках одной гибридной архитектуры. Основная цель такого объединения состояла в том, чтобы использовать преимущества и скомпенсировать недостатки каждой модели в отдельности СММ и МСП. В результате в литературе были предложены различные архитектуры и алгоритмы обучения гибридных моделей.

В предложенной статье приведен обзор различных наиболее успешных гибридных моделей, которые продемонстрировали, что МСП может быть обучен и использован для оценки эмиссионных вероятностей СММ. Отдельно отмечено, что, кроме теоретического интереса, гибридные модели позволили заметно повысить качество распознавания по сравнению со стандартными СММ. Приведенные сравнения с системами, основанными только на СММ, продемонстрировали преимущества гибридного подхода как в смысле точности распознавания, так и в смысле снижения размерности системы.

- автоматическое распознавание речи
- скрытые марковские модели
- искусственные нейронные сети
- многослойный персептрон.

Маковкин К.А.

Гибридные модели – Скрытые марковские модели/Многослойный перцептрон и их применение в системах распознавания речи. Обзор

Most of the modern automatic speech recognition (ASR) systems are based on hidden Markov models (HMMs). These models provide a fundamental structure that is powerful, flexible and effective under many circumstances, but the emission probability estimation techniques used with HMMs typically suffer from some major limitations that limit applicability of ASR technology in real-world environments.

Between the end of the 1980s and the beginning of the 1990s, some researchers began exploring a new research area, by combining HMMs and Multi-Layered Perceptrons (MLPs) within a single, hybrid architecture. The goal in hybrid systems for ASR is to take advantage from the properties of both HMMs and ANNs, improving flexibility and recognition performance. A variety of different architectures and novel training algorithms have been proposed in literature. This paper reviews a number of significant hybrid models for ASR that have demonstrated that MLPs can be discriminatively trained to estimate emission probabilities for HMMs.

It is pointed out that, in addition to their theoretical interest, hybrid systems have been allowing for tangible improvements in recognition performance over the standard HMMs in difficult and significant benchmark tasks. Given comparisons with pure HMM system illustrate advantages of the hybrid approach both in terms of recognition accuracy and number of parameters required.

• *automatic speech recognition* • *hidden markov model* • *artificial neural network* • *multi-layer perceptron*.

Введение

Применение методов статистической теории распознавания образов стало важным этапом в развитии автоматического распознавания речи (APP). Это позволило исследователям использовать мощный аппарат математической статистики и теории вероятностей, что, в свою очередь, привело к существенному повышению качества распознавания. В настоящее время практически все известные системы распознавания речи основаны на статистических методах.

В рамках такого подхода речевой сигнал представляется как случайный образ, который необходимо распознать: преобразовать в некоторую последовательность слов W . Тогда задача распознавания речевого сигнала может быть сформулирована как классическая задача классификации образов по критерию максимума апостериорной вероятности. Т.е. необходимо максимизировать апостериорную вероятность $P(W | X)$, где X — наблюдаемая последовательность акустических векторов параметров речевого сигнала, а W — последовательность слов. Согласно формуле Байеса, апостериорную вероятность можно переписать в виде

$$\arg \max_{W \in \Gamma} P(W | X) = \arg \max_{W \in \Gamma} P(X | W) \cdot P(W), \quad (1)$$

где Γ — множество всех возможных последовательностей слов, $P(X | W)$ — условная вероятность появления последовательности акустических векторов X для заданной последовательности слов W , а $P(W)$ — априорная вероятность появления последовательности слов W . Выражение $P(X | W)$ обычно называют акустико-фонетической моделью, а $P(W)$ — моделью языка [57][58].

Наиболее популярными технологиями акустико-фонетического моделирования речевого сигнала в настоящее время по праву являются технологии, основанные на скрытых марковских моделях (СММ) [56]. Использование СММ обеспечивает хорошее представление речевого сигнала и предоставляет мощный и гибкий инструмент для разработки систем APP, что в итоге позволяет разработчикам достигать высокой точности распознавания. К сожалению, при неоспоримых преимуществах СММ обладают целым рядом ограничений, например, слабой дискриминантной мощностью, т.е. способностью разделять



классы образов. Особенно это проявляется при обучении с использованием критерия максимума правдоподобия (МП) [19]. Правда, при использовании других критериев, например, критерия максимума взаимной информации (МВИ), можно достичь большей разрешающей способности. Однако эти алгоритмы математически более сложные и требуют большого числа ограничивающих предположений, что, в свою очередь, сильно усложняет их практическую реализацию. Кроме того, использование акустической и фонетической контекстуальной информации требует значительного усложнения СММ, а именно большего объема памяти для хранения параметров модели и большего количества обучающих данных.

Ещё один класс моделей, обеспечивающих акустико-фонетическое моделирование, — модели искусственных нейронных сетей (ИНС), которые с середины 80-х гг. XX в. стали активно использоваться в системах распознавания речи. Исследователями было предложено много различных архитектур нейронных сетей [39], показавших неплохие результаты по классификации речевых образов. Основные преимущества, обеспечившие ИНС популярность и широкое использование — присущие им мощные дискриминантные способности, а также возможность обучаться и представлять неявные знания. Несмотря на потенциальные возможности по классификации кратковременных акустико-фонетических единиц, таких как, например, фонемы, ИНС не стали основной моделью для создания систем АРР. Причиной тому послужил недостаток ИНС, связанный со сложностью моделирования длительных последовательностей наблюдений, например, слов или целых высказываний, поскольку эти последовательности обычно обладают сильной временной изменчивостью. Эту проблему не решило даже использование рекуррентных архитектур сети. Другими словами, ИНС хорошо работают только со статическими образами. Их эффективность сильно снижается, когда на входе появляется некоторая динамика, т.е. образы подвержены, например, нелинейным изменениям во времени.

В начале 90-х гг. XX в. факт существования двух взаимодополняющих подходов привёл исследователей к идее комбинировать СММ и ИНС в рамках одной, новой модели — гибридной СММ/ИНС модели [23][36][20][45][48][26]. Такая гибридная модель позволяет эффективно объединить преимущества марковских моделей и нейронной сети, при этом СММ обеспечивает возможность моделирования долговременных зависимостей, а ИНС — непараметрическую универсальную аппроксимацию, оценку вероятности, алгоритмы дискриминантного обучения, уменьшение числа параметров для оценки, которые обычно требуются в стандартных СММ.

Скрытые марковские модели

Краткое описание

Основные положения теории СММ были сформулированы и опубликованы на рубеже 60–70-х гг. прошлого века в серии статей Баума и др. [13][12][14][11], а первые практические результаты использования СММ в системах АРР описаны Бейкером [9] и Елинеком с коллегами из IBM [32][10][33][1]. Позднее было написано несколько обзорных статей, которые позволили использовать теорию СММ широкому кругу разработчиков в своих практических приложениях [37][4][3].

Рассмотрим пример марковской модели для звука, которая изображена на рис. 1. Эта модель состоит из последовательности состояний, обозначенных s_1, s_2, \dots, s_5 , которые связаны мгновенными вероятностными переходами, изображённые стрелками и имеющие вероятность a_{ij} , т.е. вероятность



перехода из i -го состояния в j -ое. Возможны переходы только в следующее состояние и заикливание. В каждый момент времени модель осуществляет вероятностный переход из одного состояния в другое или остаётся в том же самом состоянии, при этом происходит излучение выходного акустического вектора y_k с выходным вероятностным распределением $b_n(y_k)$, соответствующие этому состоянию. Эти вероятности называют *эмиссионными вероятностями*. Тогда некоторое высказывание, описываемое последовательностью акустических векторов параметров $X = \{x_1, x_2, \dots, x_N\}$, можно промоделировать последовательностью дискретных стационарных состояний $Q = \{q_1, q_2, \dots, q_K\}$ $K \leq N$ с мгновенными переходами между этими состояниями и последовательностью излученных при этом акустических векторов $Y = \{y_1, y_2, \dots, y_N\}$.

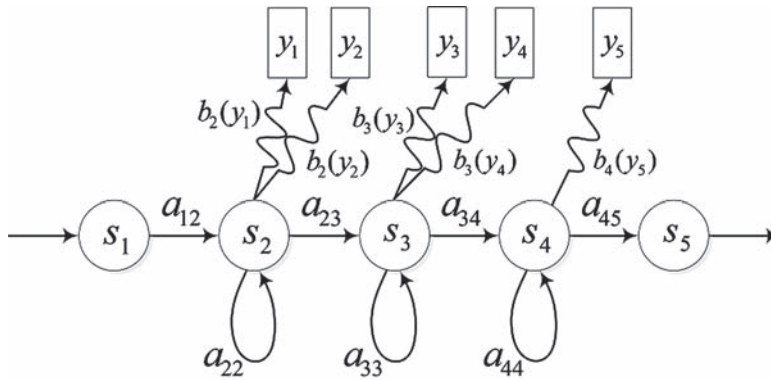


Рис. 1. Скрытая марковская модель

Таким образом, СММ состоит из марковской цепи с конечным числом состояний s_i и матрицей переходных (транзитивных) вероятностей a_{ij} , которые определяют длительность пребывания системы в данном состоянии, т.е. марковская цепь моделирует временные изменения речевого сигнала. А также конечного множества распределений эмиссионных вероятностей $b_n(y_k)$, которые позволяют моделировать спектральные вариации сигнала. Этот подход определяет два одновременных стохастических процесса, один из которых является основным и ненаблюдаемым (т.е. скрытым) — это последовательность СММ состояний. Мы можем судить о нём только с помощью другого случайного процесса, т.е. по последовательности наблюдений.

Для определения СММ необходимо задать следующие элементы:

1. Множество состояний модели $S = \{s_1, s_2, \dots, s_M\}$, где M — число состояний в модели. Состояние модели в момент времени n обозначается q_n .

2. Множество различных символов наблюдения, которые могут породиться моделью $Y = \{y_1, y_2, \dots, y_K\}$, где K — общее число символов наблюдения модели. Символы наблюдения соответствуют физическому выходу моделируемой системы.

3. Распределение вероятностей переходов между состояниями (матрица переходных вероятностей) $A = \{a_{ij}\}$, где

$$a_{ij} = P[q_{n+1} = s_j | q_n = s_i], \quad 1 \leq i, j \leq M, \quad (2)$$

при этом предполагается, что a_{ij} не зависят от времени.

4. Множество распределений вероятностей появления символов наблюдения (эмиссионные или выходные вероятности) в состоянии j , $B = \{b_j(k)\}$, где

$$b_j(k) = P[y_k \text{ в момент } n | q_t = s_j], \quad 1 \leq j \leq M, 1 \leq k \leq K \quad (3)$$

5. Начальное распределение вероятностей состояний $\Pi = \{\pi_i\}$

$$\pi_i = P[q_1 = s_i], \quad 1 \leq i \leq M \quad (4)$$

Использование СММ в системах распознавания речи

Чтобы использовать СММ в системе АРР, необходимо сделать несколько упрощающих, но очень важных предположений о речевом сигнале:

- последовательные наблюдения являются статистически независимыми и, следовательно, вероятность последовательности наблюдений есть произведение вероятности отдельных наблюдений;
- речь представляет собой нестационарный процесс, однако, он моделируется последовательностью векторов наблюдений, которые представляют кусочно-стационарный процесс;
- собственно марковское допущение — вероятность пребывания в некотором состоянии в момент времени n зависит только от состояния, в котором процесс находился в момент времени $n - 1$.

Теперь рассмотрим простую систему распознавания. Идеально было бы иметь СММ для каждого из возможных высказываний. Однако, очевидно, что это выполнимо только для очень ограниченных задач, например, распознавание изолированных команд из небольшого словаря. Поэтому используют более мелкие речевые единицы, например, фонемы, которые с лингвистической точки зрения соответствуют фонемам. Для каждого фона необходимо создать свою отдельную СММ, т.е. создать множество $M = \{m_1, m_2, \dots, m_U\}$ марковских моделей для всех возможных фонов, множество связанных с ними параметров обозначим $\Theta = \{\lambda_1, \lambda_2, \dots, \lambda_U\}$. Тогда M_i будет представлять марковскую модель некоторого слова, полученную конкатенацией элементарных моделей m_j из множества M . При этом M_i состоит из L_i состояний $q_l \in S$ и $l = 1, 2, \dots, L_i$, а множество параметров этой модели будет A_i , которое является подмножеством Θ . Произнесение каждого фона описывается последовательностью векторов спектральных характеристик сигнала. На этапе обучения для каждого слова M_i имеется множество его произнесений одним или несколькими дикторами. Каждое из произнесений представлено последовательностью векторов параметров X_{M_i} . При этом необходимо выбрать такое множество параметров Θ , которое максимизировало бы вероятность $P(M_i | X_{M_i}, \Theta)$ для всех обучающих высказываний X_{M_i} , связанных с M_i , т.е.

$$\arg \max_{\Theta} \prod_{i=1}^I P(M_i | X_{M_i}, \Theta), \quad (5)$$

где I — число произнесённых реализаций слова M_i , использованных для обучения. Таким образом, обучение состоит в подборе параметров модели Θ в соответствии с некоторым критерием оптимальности. К сожалению, не существует известного аналитического выражения для вычисления этих параметров. Кроме того, на практике, располагая некоторой последовательностью наблюдений в качестве обучающих данных, нельзя указать оптимальный способ оценки этих параметров. Однако, используя итеративные процедуры, например, алгоритм Баума-Уэлча (Baum-Welch Algorithm), который является частным случаем EM-алгоритма (Expectation-maximization (EM) algorithm) [22], или градиентные методы [37], можно вычислить значения параметров модели, соответствующие локальному максимуму вероятности $P(M | X, \Theta_{LM})$. Следует отметить, что эти алгоритмы принадлежат классу алгоритмов обучения «без учителя», так как они производят ненаблюдаемую оценку параметров распределений вероятностей, не требуя предварительной разметки. На этапе распознавания неизвестного высказывания X необходимо найти наиболее подходящую модель M_i , которая максимизирует вероятность $P(M | X, \Theta)$ при уже фиксированном множестве параметров Θ и наблюдаемой в данный момент последовательности X . Таким образом,

результатом распознавания последовательности X будет слово, соответствующее модели M_i

$$i = \arg \max_{\forall j} P(M_j | X, \Theta). \quad (6)$$

Метод нахождения наилучшей модели основан на динамическом программировании и называется алгоритмом Витерби [56].

Обучение и распознавание связаны с выбором некоторого критерия оптимальности. Таких критериев существует несколько, например, максимум правдоподобия или максимум апостериорной вероятности. Все они имеют физический смысл и используются на практике. Выбранный критерий оптимальности оказывает влияние на такие параметры модели, как объём данных для обучения и требования к вычислительным ресурсам, точность распознавания, способность к обобщению данных из обучающей выборки. Одним из наилучших критериев может считаться Байесовский классификатор (классификатор по максимуму апостериорной вероятности, MAP-оценитель), основанный на апостериорной вероятности $P(M_i | X, \Theta)$ того, что последовательность акустических векторов X была порождена моделью M_i с множеством параметров Θ .

Используя правило Байеса, $P(M_i | X, \Theta)$ можно записать в виде выражения

$$P(M_i | X, \Theta) = \frac{P(X | M_i, \Theta)P(M_i | \Theta)}{P(X | \Theta)}, \quad (7)$$

которое разделяет процесс оценки вероятности на две части: задачу акустико-фонетического моделирования

$$\frac{P(X | M_i, \Theta)}{P(X | \Theta)} \quad (8)$$

и модель языка $P(M_i | \Theta)$. Задача модели языка — оценка априорных вероятностей моделей высказываний $P(M_i | \Theta)$. Эта модель обычно полагается независимой от акустических моделей и описывается в терминах независимого множества параметров Θ^* . Параметры модели языка обычно оцениваются на больших текстовых базах данных [49].

Задача акустико-фонетического моделирования — оценка плотностей вероятностей (8), как правило, независимо от других моделей. Так как вероятность $P(X | M_i, \Theta)$ обусловлена только M_i , то она зависит только от параметров M_i модели, и опуская $P(X | \Theta)$, как в [18], выражение (8) можно переписать как $P(X | M_i, A_i)$, где A_i — множество параметров, связанных с моделью M_i . Таким образом, и обучение, и распознавание требует оценки вероятности $P(X | M_i, A_i)$, которая называется глобальным правдоподобием последовательности векторов параметров X при заданной модели M_i .

Вероятность $P(X | M_i, A_i)$ можно оценить как сумму

$$P(X | M_i, A_i) = \sum_{\{\Gamma_i\}} P(X, \Gamma_i | M_i, A_i), \quad (9)$$

где $\{\Gamma_i\}$ — множество всех возможных путей (последовательностей состояний) длины L в модели M_i . При этом для каждой последовательности состояний вероятность появления последовательности наблюдений $X_1^L = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_L\}$ определяется выражением

$$P(X_1^L | q_1^L, M_i, A_i) = \prod_{l=1}^L P(\mathbf{x}_l | q_l^l, X_1^{l-1}, M_i, A_i), \quad (10)$$

где $Q_1^L = \{q_1, q_2, \dots, q_L\}$ — последовательность состояний. Можно показать [17], что (10) вычисляется с помощью алгоритма прямого-обратного хода [56], для которого необходимо рекурсивно вычислять т.н. прямую переменную

$$P(q_l^n, X_1^n | M_i, A_i) = \sum_{k=1}^L P(q_k^{n-1}, X_1^{n-1} | M_i, A_i) p(q_l^n, \mathbf{x}_n | q_k^{n-1}, X_1^{n-1}, M_i, A_i), \quad (11)$$

где $P(q_l^n X_1^n | M_i, A_i)$ — вероятность того, что частичная подпоследовательность наблюдений $X_1^n = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n\}$ порождена моделью M_i , а в момент времени n наблюдалось состояние $q_l^n = s_l$, и был сгенерен вектор наблюдений \mathbf{x}_n .

Второй сомножитель в правой части равенства (11) можно представить в виде произведения вероятностей

$$p(q_l^n, \mathbf{x}_n | q_k^{n-1}, X_1^{n-1}, M_i, A_i) = p(\mathbf{x}_n | q_l^n, q_k^{n-1}, M_i, A_i) p(q_l^n | q_k^{n-1}, M_i, A_i), \quad (12)$$

где первый сомножитель $p(\mathbf{x}_n | q_l^n, q_k^{n-1}, M_i, A_i)$ — *эмиссионная вероятность*, а второй $p(q_l^n | q_k^{n-1}, M_i, A_i)$ — *транзитивная вероятность*. Обычно эмиссионную вероятность упрощают, чтобы снизить число свободных параметров, полагая, что наблюдаемый акустический вектор \mathbf{x}_n зависит только от текущего состояния процесса q_l^n , т.е. используют эмиссионную вероятность в виде $p(\mathbf{x}_n | q_l^n)$.

Описанная стандартная СММ — мощный инструмент, позволивший разработчикам существенно повысить качество распознавания речевого сигнала. Это демонстрирует целый ряд лабораторных систем распознавания слитной речи с большими словарями (1000–40000 слов), которые занимают высокое место в сравнительных испытаниях, проведённых в рамках проекта SQALE [76]. В экспериментах участвовали три системы построенные на СММ:

- 1) система распознавания Cu-НТК, которая была разработана Стивом Янгом (Steve Young) в Кэмбриджском университете в 1987 г. [77];
- 2) система распознавания LIMSI, разработанная в Laboratoire d'Informatique pour la Mecanique et les Sciences le l'Ingenieur во Франции;
- 3) система Philips, разработанная в лаборатории человеко-машинного интерфейса фирмы Philips в Германии.

Завершая краткое описание СММ, необходимо отметить, что наряду с неоспоримыми достоинствами, такими как:

- мощный математический аппарат;
- эффективное моделирование как временных, так и спектральных вариации речевого сигнала;
- достаточно гибкая топология — СММ могут легко включать не только фонологические правила или, например, строить модели слов из моделей фонем, но и позволяют использовать синтаксические правила;
- глубокая практическая проработка — разработаны мощные обучающие и распознающие алгоритмы, которые обеспечивают эффективное обучение на больших речевых базах данных и распознавание изолированных слов и слитной речи без адаптации под диктора в реальном масштабе времени;

исследования выявили целый ряд недостатков:

- слабые дискриминантные способности, поскольку во время обучения акустические модели формируются на основе критерия максимума правдоподобия, а не более точного максимума апостериорной вероятности;
- последовательности векторов наблюдений считаются статистически независимыми, что неверно для речевого сигнала;
- кусочно-постоянный характер модели, т.е. каждое марковское состояние имеет стационарную статистику, а это значит, что независимо от времени нахождения в данном состоянии распределения эмиссионных вероятностей одинаковы;
- априорный выбор топологии модели и статистических распределений;
- отсутствие эффективных и адекватных природе речевого сигнала моделей длительности состояний и их реализации в рамках марковских моделей;

Маковкин К.А.

Гибридные модели – Скрытые марковские модели/Многослойный перцептрон и их применение в системах распознавания речи. Обзор

- марковская модель полагается моделью первого порядка, т.е. состояние в момент времени n зависит только от предыдущего состояния в момент времени $n - 1$;
- обучение и оптимизация лингвистической модели происходят отдельно от акустических моделей.

Перечисленные предположения и недостатки существенно ограничивают возможности такого класса моделей [47] по более точному представлению речевого сигнала и препятствуют дальнейшему росту качества систем распознавания. Несмотря на многочисленные исследования последних лет, эти ограничения преодолеть пока не удастся. Поэтому в сложившейся ситуации стремление повысить точность распознавания побуждает исследователей к поиску альтернативных или дополняющих подходов к решению проблемы акустико-фонетического моделирования речевого сигнала.

Нейронные сети

Многослойный перцептрон

Другим классом моделей, которые используются для акустико-фонетического моделирования речевого сигнала, являются модели искусственных нейронных сетей (ИНС). Структуры и принципы их работы основываются на биологических моделях нервных систем, и прежде всего на моделях головного мозга. Нейронные сети — множество однотипных и параллельно функционирующих элементов или нейронов. Они могут рассматриваться как разновидность самоорганизующихся алгоритмов. Каждый нейрон обладает набором входных связей, с помощью которых он соединяется с «внешним миром» или с другими нейронами. В дискретные моменты времени на входные связи нейрона подаётся информация, на основе которой в соответствии с некоторыми принципами формируется выходной сигнал, который, в свою очередь, передаётся на входы других нейронов или во «внешний мир».

Наиболее распространённой является модель нейрона МакКаллока-Питса (McCulloch-Pitts) (рис. 2), предложенная в 1943 г. [42][64]. В соответствии с ней нейрон имеет набор входных связей и один выход, который может распараллеливаться.

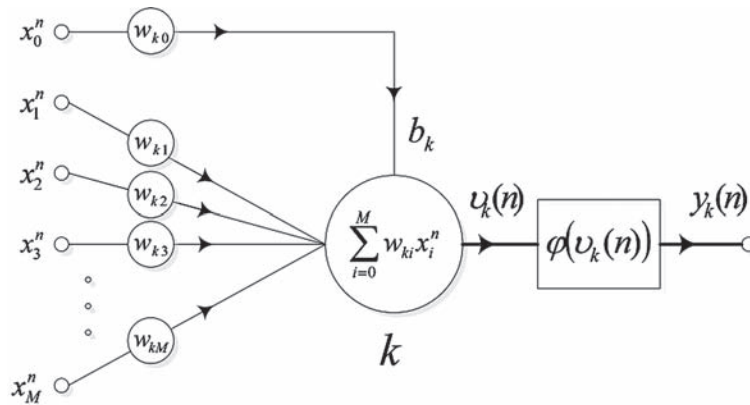


Рис. 2. Модель нейрона

Формально функционирование такого нейрона с индексом k можно описать в виде следующих уравнений:

$$u_k(n) = \sum_{i=0}^M w_{ki} x_i^n, \quad (13)$$

$$y_k(n) = \phi(u_k(n) + b_k), \quad (14)$$

где x_i^n — компоненты входного вектора $\mathbf{X}_n = \{x_1^n, x_1^n, \dots, x_M^n\}$ размерностью M в момент времени n , $\mathbf{w}_k = \{w_{k1}, w_{k2}, \dots, w_{kM}\}$ — вектор синаптических весов k -го нейрона, b_k — порог чувствительности нейрона, $\varphi(\cdot)$ — функция активации, $y_k(n)$ — выходной сигнал нейрона. Сумма $v_k(n) = u_k(n) + b_k$, — индуцированное локальное поле. Порог b_k позволяет выполнить смещение индуцированного локального поля по горизонтальной оси и тем самым изменить порог чувствительности нейрона. Для упрощения формулы (14) порог b_k можно представить в виде дополнительного постоянного воздействия $x_i^n = +1$, а значение веса соответствующей ему связи $w_{k0} = b_k$. Тогда уравнения (13) и (14) можно переписать как

$$v_k(n) = \sum_{i=0}^M w_{ki} x_i^n, \quad (15)$$

$$y_k(n) = \varphi(v_k(n)). \quad (16)$$

Функция активации $\varphi(\cdot)$, определяющая зависимость сигнала на выходе нейрона от индуцированного локального поля $v_k(n)$, в большинстве случаев является нелинейной монотонно возрастающей и имеет область значений $[-1, 1]$ или $[0, 1]$. Кроме того, $\varphi(\cdot)$ является непрерывно дифференцируемой, что необходимо при использовании градиентных алгоритмов обучения. Сигмоидальная функция — самая распространённая функцией, используемая в качестве функции активации. Это быстро возрастающая функция, которая поддерживает баланс между линейным и нелинейным поведением. Примером сигмоидальной функции может служить логистическая функция

$$\varphi(v) = \frac{1}{1 + e^{-\alpha v}}, \quad (17)$$

где α — параметр наклона функции, v — индуцированное локальное поле, или гиперболический тангенс

$$\varphi(v) = \tanh(\alpha v), \quad (18)$$

где α — также параметр наклона функции, т.е. влияет на форму функции активации. При малых значениях α графики функций достаточно пологие, а по мере роста их крутизна увеличивается. При $\alpha \rightarrow \infty$ функция активации превращается в функцию ступенчатого типа. Значение α в формулах (17) и (18) обычно подбирается пользователем. Наличие нелинейности играет очень важную роль, так как в противном случае комбинация линейных функций даст на выходе опять линейную функцию, и отображение «вход-выход» сети можно свести к линейному преобразованию. Более того, использование логистической функции мотивированно биологически, поскольку в ней учитывается восстановительная фаза реального нейрона.

С конца 80-х гг. XX в. многие исследователи начали активно использовать модели нейронных сетей в системах АРР. Это отразилось на числе работ, посвящённых распознаванию речи с помощью нейронных сетей, которое возросло в несколько раз. При этом исследователями было предложено много различных архитектур нейронных сетей [39], которые использовались для классификации кратковременных речевых образов и продемонстрировали неплохие результаты.

Самой известной и наиболее распространённой моделью нейронной сети является многослойный персептрон (МСП), архитектура которого была предложена в 1986 г. [68] Структурная схема МСП представлена на рис. 3.

Маковкин К.А.

Гибридные модели — Скрытые марковские модели/Многослойный перцептрон и их применение в системах распознавания речи. Обзор

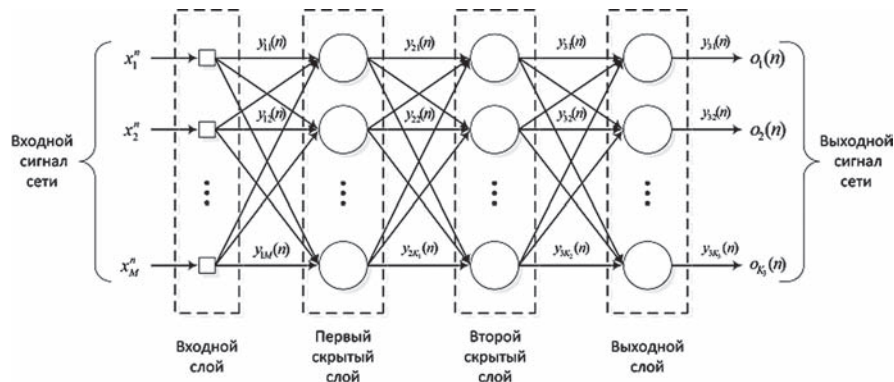


Рис. 3. Многослойный перцептрон

Топология МСП — иерархическая сетевая структура, в которой нейроны разделены на несколько слоёв. Внутри слоя нейроны можно считать линейно упорядоченными и не взаимодействующими между собой. На низшем уровне иерархии находится входной слой, состоящий из множества сенсорных элементов (рецепторов). Этот слой — вырожденный. Его задача состоит только в приёме и распространении по сети входной информации. Далее имеются один или два, реже несколько скрытых слоёв. Последний слой — выходной. На выходах нейронов этого слоя формируется отклик сети. Таким образом, каждый нейрон сети получает входной сигнал от каждого нейрона предыдущего слоя, т.е. МСП — сеть прямого распространения, в которой входной сигнал распространяется только в прямом направлении от слоя к слою.

Самый известный алгоритм обучения МСП — алгоритм обратного распространения ошибки (BP-алгоритм) (Back Propagation Error), описанный Rosenblatt в 1959 г. [64], или его модификация, предложенная Rumelhart в [68]. BP-алгоритм основан на классическом методе градиентного спуска и в настоящее время считается одним из наиболее простых и эффективных алгоритмов обучения, который позволяет осуществить управляемое обучение (обучение «с учителем»).

Алгоритм обратного распространения — алгоритм оптимизации, который минимизирует функцию расстояния (целевую функцию) между желаемым и сгенерированным выходом сети. Цель обучения — установление желаемого функционального соотношения входа и выхода путём коррекции значений весов связей между нейронами. После выбора некоторых начальных значений весов, в процессе обучения итерационно на сеть одновременно подаются входной и желаемый выходной (целевой) вектор. Сеть выполняет отображение входного вектора в выходной. Разность полученного и целевого выхода k -го нейрона выходного слоя — ошибка ε_k , т.е.

$$\varepsilon_k(n) = y_k^{trg}(n) - g_k(\mathbf{w}_k, \mathbf{x}_n), \quad (19)$$

где $y_k^{trg}(n)$ — целевой выход k -го нейрона на n -ом шаге алгоритма, $\mathbf{w}_k = \{w_{k1}, w_{k2}, \dots, w_{kM}\}$ — вектор весов k -го нейрона, \mathbf{x}_n — входной вектор и $g_k(\cdot)$ — функция нелинейного отображения «вход-выход», реализуемая МСП для k -го выхода МСП. ε_k используется для подстройки w_{kj} при её обратном распространении от выхода сети ко входу. В качестве целевых выбирают различные функции, так, например, среднеквадратичную ошибку

$$E = \sum_{n=1}^N \|\mathbf{y}(n) - \mathbf{y}^{trg}(n)\|^2 \quad (20)$$

или функцию относительной энтропии

$$E_e = \sum_{n=1}^N \sum_{k=1}^K \left[y_k^{trg}(n) \ln \frac{y_k^{trg}(n)}{y_k(n)} + (1 - y_k^{trg}(n)) \ln \frac{1 - y_k^{trg}(n)}{1 - y_k(n)} \right], \quad (21)$$

где $y_k^{trg}(n)$ — целевой, а $y_k(n)$ — наблюдаемый выход k -го нейрона выходного слоя на n -ом шаге алгоритма, K — число нейронов в выходном слое и N — общее число обучающих образов.

Основной момент в обучении сети — способ коррекции весов связей (22). Поскольку обучение проводится методом наискорейшего спуска, то уточнение весов связей проводится в направлении противоположном градиенту целевой функции в соответствии с дельта-правилом (23)[27]

$$w_{ij}(n+1) = w_{ij}(n) + \Delta w_{ij}(n), \quad (22)$$

$$\Delta w_{ij}(n) = -\eta \frac{\partial E}{\partial w_{ij}(n)} x_i(n), \quad (23)$$

где η — коэффициент обучения, значение которого, как правило, выбирается из интервала $[0, 1]$. Чем меньше параметр скорости обучения η , тем меньше корректировка синаптических весов, осуществляемая на каждой итерации, и тем более гладкой является траектория в пространстве весов, которая строится в процессе оптимизации. Однако это происходит за счёт замедления процесса обучения. С другой стороны, если увеличивать параметр η для повышения скорости обучения, то результирующие большие изменения синаптических весов могут привести систему в неустойчивое состояние. Простейший способ повышения скорости обучения без потери устойчивости — изменение дельта-правила (23) за счёт добавления к нему момента инерции [66][59]

$$\Delta w_{ij}(n) = \mu \Delta w_{ij}(n) - \eta \frac{\partial E}{\partial w_{ij}(n)} x_i(n), \quad (24)$$

где μ — как правило, положительное значение из интервала $0 \leq \mu < 1$, которое называют постоянной момента. При использовании момента, процесс модификации весов определяется не только информацией о градиенте функции, но и фактическим трендом изменений весов $\Delta w_{ij}(n)$, что в процессе обучения проявляется следующим образом. Если частная производная $\frac{\partial E}{\partial w_{ij}(n)}$ имеет один и тот же алгебраический знак на нескольких

последовательных итерациях, то благодаря моменту $\Delta w_{ij}(n)$ возрастает по абсолютному значению, поэтому веса $w_{ij}(n)$ могут изменяться на очень большую величину. Таким образом, включение момента в алгоритм обратного распространения ведёт к ускорению спуска в некотором постоянном направлении. Если же $\frac{\partial E}{\partial w_{ij}(n)}$ на нескольких последовательных итерациях

меняет знак, то $\Delta w_{ij}(n)$ уменьшается по абсолютному значению и $w_{ij}(n)$ меняется незначительно, что ведёт к стабилизирующему эффекту для направлений, изменяющих знак.

Включение момента в алгоритм обратного распространения обеспечивает незначительную модификацию метода корректировки весов, оказывая положительное влияние на работу алгоритма обучения. Кроме того, слагаемое момента может предотвратить попадание в локальный минимум на поверхности ошибок. Как видно из (24), влияние момента особенно сильно проявляется в непосредственной близости к локальному минимуму, где значение градиента стремится к нулю. Это приводит к возрастанию значений целевой функции и ее выходу из области локального минимума. Однако сильное влияние момента (при больших значениях μ) может привести к нестабильности, т.е. расходимости алгоритма обучения.

Маковкин К.А.

Гибридные модели – Скрытые марковские модели/Многослойный персептрон и их применение в системах распознавания речи. Обзор

В заключение описания алгоритма обучения МСП следует отметить, что несмотря на многочисленные успешные применения метода обратного распространения ошибки, он обладает рядом недостатков. Во-первых, в общем случае не существует доказательства сходимости алгоритма обратного распространения. Во-вторых, не существует какого-либо четко определённого критерия для остановки алгоритма, т.е. прекращения корректировки весов. Это приводит к тому, что процесс обучения может стать неопределённо долгим, поэтому обычно используют несколько практически обоснованных критериев. В-третьих, алгоритм обратного распространения построен на основе метода градиентного спуска, а все градиентные методы гарантируют достижение локального минимума целевой функции, т.е. минимальной точки в некоторой своей окрестности, но лежащей выше глобального минимума. При этом необходимо учитывать, что в случае нейронной сети с сигмоидальной функцией активации нейронов поверхность ошибок может иметь очень сложное строение и обладать множеством локальных минимумов, плоскими участками, седловыми точками и длинными узкими «оврагами» в пространстве высокой размерности. Это повышает вероятность попадания в случайный локальный минимум, а для достижения высокого качества обобщения необходимо, чтобы процесс обучения завершился в точке, максимально близкой к глобальному минимуму. В связи с этим возникает необходимость применения методов глобальной оптимизации. В настоящее время в этой области разработано довольно много различных методов и алгоритмов, например, имитация отжига [5] и генетические алгоритмы [44][55].

В первых экспериментах [64] однослойный персептрон показал очень хорошие результаты при обучении в простых нелинейных задачах. Можно показать [6], что однослойный персептрон, как классификатор образов, формирует в пространстве признаков дискриминантные гиперплоскости, которые при пересекающихся классах образов и слабо нелинейной пороговой функции минимизируют среднеквадратическую ошибку между y_k и y_k^{trg} , т.е. однослойные персептроны эквивалентны параметрическим гауссовым классификаторам (их использование приводит к оценке максимального правдоподобия). Другими словами, для двух классов, образы которых распределены по нормальному закону, и в предположении, что признаки, описывающие образы, некоррелированы, можно построить однослойный персептрон с такой же решающей функцией, как у параметрического гауссова классификатора.

Однако однослойный персептрон не может разделить образы, требующие для разделения более сложные поверхности в пространстве признаков. Так, например, однослойный персептрон не может решить проблему исключяющего «или» путём построения простой гиперплоскости.

С увеличением количества слоёв классификационные свойства персептрона качественно улучшаются. Двухслойный персептрон может решить проблему исключяющего «или» посредством формирования выпуклой поверхности в качестве разделяющей (как результат пересечения гиперплоскостей, формируемых элементами первого слоя). Однако двухслойный персептрон также обладает ограниченными возможностями. Так, Minsky и Papert в своей работе [43] доказали, что и двухслойный персептрон не может успешно представить или аппроксимировать функции вне очень узкого и специфического класса. Правда, Minsky и Papert оставили открытым вопрос о возможностях МСП по аппроксимации общего отображения из одного конечно размерного пространства в другое.

Использование трёхслойного персептрона открывает ещё большие возможности в аппроксимации отображения из одного конечно размерного пространства в другое, т.е. трёхслойный персептрон может формировать разделяющие поверхности любой формы и получать любые, заранее заданные непрерывные функции входных сигналов. В частности, с помощью выбора соответствующей решающей функции он может эмулировать любой традиционный детерминированный классификатор [38].



Теоретические основания о подобных выводах о потенциальных свойствах трёх-слойного перцептрона обеспечивает теорема об универсальной аппроксимации для нелинейного отображения «вход-выход». Она утверждает, что МСП с одним скрытым слоем достаточно для построения равномерной аппроксимации с точностью ε для любого обучающего множества [27]. Теоретические основания также обеспечивает результат А.Н. Колмогорова о возможности представления всякой действительной непрерывной функции N переменных в виде суперпозиции конечного числа непрерывных действительных функций с глубиной вложения не более трёх, в которой используется только линейное суммирование аргументов и непрерывно возрастающие функции одной переменной [2], или более поздние работы [40][28].

Основными мотивационными факторами к использованию МСП послужили следующие преимущества нейронных сетей:

- МСП может осуществить дискриминантное обучение между речевыми единицами, которые представляют выходные классы перцептрона. При этом МСП не только обучается и оптимизирует параметры для каждого класса на данных принадлежащих ему, но и пытается отклонять данные принадлежащие другим классам;
- МСП может найти оптимальную комбинацию ограничений для классификации. При этом нет необходимости в строгих предположениях о распределении входных признаков, что обычно требуется в стандартных СММ;
- МСП — структура с высокой степенью параллелизма.

Первые работы по использованию МСП в системах распознавания речи [7][52] выявили один важный недостаток МСП и вообще ИНС. Эти модели были разработаны для распознавания статических сигналов, а не для их последовательностей или сигналов, подверженных временной вариативности. Поэтому МСП достаточно удачно использовался как классификатор речевых классов, например, изолированных слов [39], а попытки использовать его в системах распознавания слитной речи не увенчались успехом.

Модификации многослойного перцептрона

Как известно, на спектральные характеристики таких речевых единиц, как фонема, оказывает сильное влияние контекст, т.е. то какой звук был произнесён до и какой будет произнесён после. Кроме того, при распознавании речевого сигнала очень важна его динамика или то, как сильно меняются спектральные характеристики сигнала от фрейма к фрейму. Для того, чтобы учитывать эти свойства речевого сигнала, исследователями были предложены различные модификации МСП, например нейронная сеть с задержкой по времени (Time-Delay Neural Network (TDNN)) [71][70][72] или рекуррентная нейронная сеть (Recurrent Neural Network (RNN)) [8][53][67].

Нейронная сеть с задержкой по времени

Сеть с задержкой по времени (рис. 4) реализует одну из попыток использовать статический МСП для распознавания динамической временной последовательности речевых данных путём преобразования временной последовательности в пространственную последовательность соответствующих нейронов. В этом случае модификация МСП состоит в том, что в каждый момент времени на нейроны, образующие входной слой, поступает не только текущий вектор параметров \mathbf{x}_n в момент времени n , но и часть последовательности векторов, взятых с запаздыванием — $\mathbf{X}_{n-c}^{n-1} = \{\mathbf{x}_{n-c}, \mathbf{x}_{n-(c-1)}, \dots, \mathbf{x}_{n-1}\}$



Маковкин К.А.

Гибридные модели – Скрытые марковские модели/Многослойный перцептрон и их применение в системах распознавания речи. Обзор

и с опережением $X_{n+1}^{n+c} = \{\mathbf{x}_{n+1}, \mathbf{x}_{n+2}, \dots, \mathbf{x}_{n+c}\}$. Получается, что активность каждого нейрона из скрытого слоя зависит от активности нейронов входного слоя на некотором конечном временном интервале X_{n-c}^{n+c} длиной $2c + 1$.

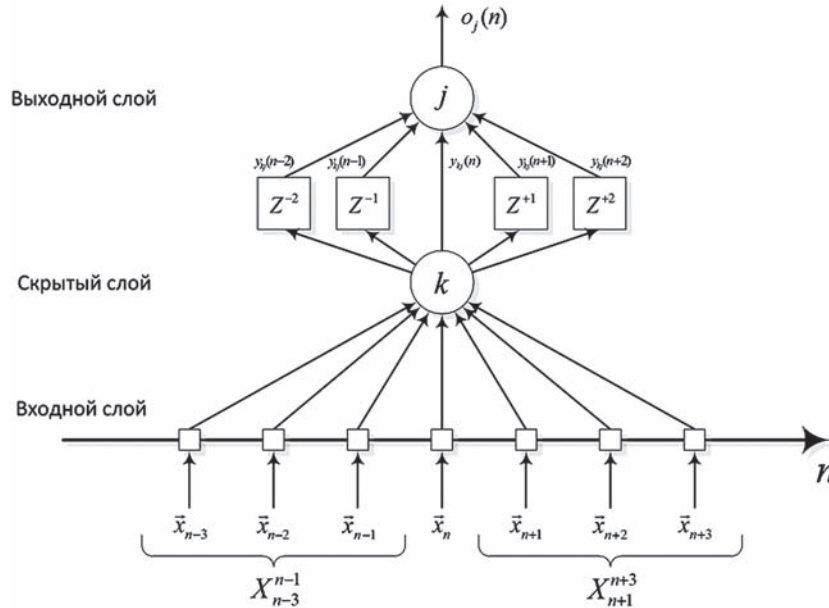


Рис. 4. Нейронная сеть с задержкой по времени

Аналогично выходной слой связан со скрытым слоем. Как видно на рис. 4, активность выходного нейрона определяется активностью нейрона из скрытого слоя, взятой на временном интервале $[n - 2, n + 2]$. Число шагов, на которое МСП «заглядывает» вперед и назад во времени, выбирается разработчиком модели. Для обучения сети с такой топологией также может использоваться алгоритм обратного распространения.

Одним из первых эту модель исследовал А. Waibel и др. [73]. Lang и Hinton [35] использовали TDNN в эксперименте по распознаванию изолированных звуков «B, D, E, V» без подстройки под диктора. Для обучения сети использовался акустический материал, собранный от 100 дикторов-мужчин. В результате была достигнута точность 7,8% ошибок. Последующие эксперименты с синтезом модульной сети [72][74], в которой каждый отдельный модуль представлял собой TDNN сеть, специфицированную для распознавания звуков, показали возможность надёжной идентификации всех согласных японского языка, изолированно произносимых дикторами-японцами. Точность распознавания в этих экспериментах достигла 95,9%. При этом точность распознавания гласных звуков в тех же экспериментах достигла 98,6%.

Рекуррентная нейронная сеть

Другой способ моделировать контекстную информацию состоит в модификации МСП за счёт добавления в него обратных связей. При этом в каждый контур таких связей включён элемент единичной задержки, благодаря которому поток сигналов остаётся однонаправленным, т.е. выходной сигнал предыдущего временного цикла рассматривается как априори заданный, который просто увеличивает размерность входного вектора нейрона (рис. 5). Такие связи называют рекуррентными, а модель в целом — рекуррентной нейронной сетью (Recurrent Neural Network (RNN))[65] или динамической [51] нейронной сетью.

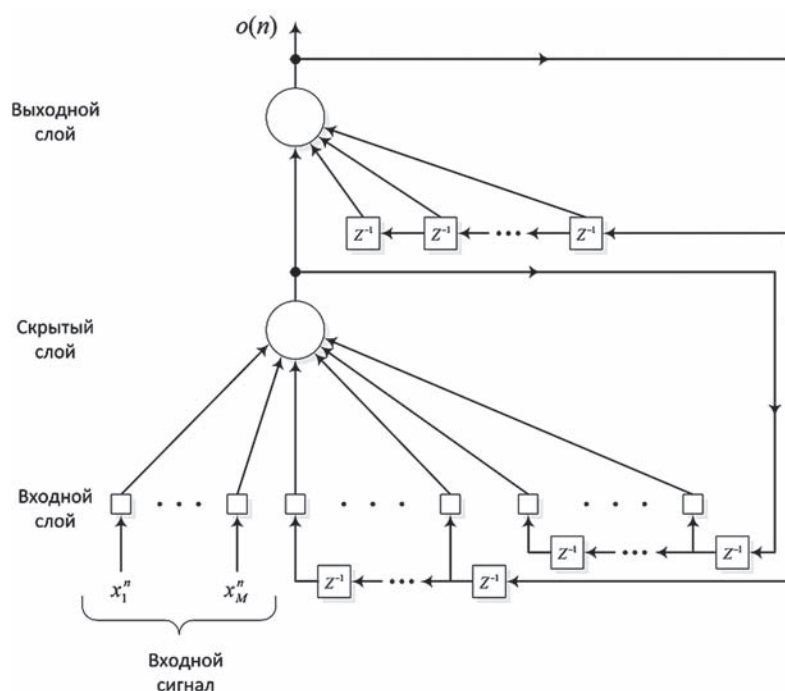


Рис. 5. Рекуррентная нейронная сеть

Поначалу RNN мало использовались для систем распознавания речи из-за больших сложностей с обучением, анализом и разработкой. Так, например, алгоритм обучения, адаптирующий значения синаптических весов такой сети, является более сложным вследствие зависимости сигналов в момент времени n от их значений в предыдущие моменты и соответственно ввиду более громоздкой формулы для расчёта вектора градиента. Однако в результате ряда исследований было предложено несколько модификаций алгоритма обратного распространения, например, такие, как рекуррентный BP [54], BP для последовательностей [25], рекуррентное обучение в реальном времени [75], время-зависимый рекуррентный BP алгоритм [50][69] и наиболее популярный BP во времени [68], которые значительно облегчили использование рекуррентных структур в системах распознавания речи [62].

Результатом применения таких модификаций многослойного перцептрона стало повышение качества распознавания кратковременных акустико-фонемических единиц, таких как фонемы, и лишь незначительно улучшение распознавание длительных последовательностей акустических наблюдений, которые необходимы для представления таких лингвистических единиц, как, например, слова. Теоретическое обоснование этого результата приводится в [15]. Кроме того, эти исследования выявили ряд существенных недостатков, которые не позволили сделать МСП основной структурой для систем распознавания речи. Во-первых, МСП не имеют механизмов, которые бы адекватно представляли временную вариативность и последовательную природу речевого сигнала. Во-вторых, для целого ряда параметров, определяющих динамику и топологию МСП, пока не существует теоретических основ, позволяющих вычислить или выбрать эти параметры (они выбираются по усмотрению разработчика). В-третьих, несмотря на то, что разработан целый ряд алгоритмов, которые ускоряют процедуру обучения, она остаётся очень ресурсоёмким и длительным процессом.

Гибридные модели МСП и СММ

Существование двух подходов, таких как СММ и ИНС, взаимно дополняющих друг друга и компенсирующих присущие им недостатки, в начале 90-х гг. XX в. привело исследователей к идее комбинировать эти структуры в рамках одной новой модели, которую определили как гибридную СММ/МСП модель [23][36][20][45][48][26]. Такая гибридная модель позволяет эффективно объединить преимущества марковских моделей и нейронной сети, при этом СММ обеспечивает возможность моделирования долговременных зависимостей, а МСП — непараметрическую универсальную аппроксимацию, оценку вероятности, алгоритмы дискриминантного обучения, уменьшение числа параметров для оценки, которые обычно требуются в стандартных СММ. Результатом использования таких гибридных структур явилось значительное повышение качества распознавания по сравнению со стандартными методами.

Архитектура гибридной модели

Как отмечалось выше, при использовании СММ в формуле (11) необходимо иметь оценку эмиссионной вероятности $p(x_n | q_l)$, которая представляет собой вероятность наблюдения вектора x_n при заданном гипотетическом СММ состоянии q_l . В начале 90-х гг. прошлого века Bourlard и др. [20][45][16][17] предложили использовать МСП для оценки вероятности $p(q_l | x_n)$, которая является апостериорной вероятностью СММ состоянии q_l при заданном наблюдаемом акустическом векторе x_n . Эту вероятность в соответствии с правилом Байеса можно пересчитать в эмиссионную вероятность.

Формально это выглядит следующим образом. Пусть $g_k(\cdot)$ при $k = 1, \dots, K$ — функция, реализуемая k -м выходом перцептрона, тогда $g_k(\cdot)$ можно связать с дискретным СММ состоянием s_k . Теперь, если объединить множество параметров Θ_{HMM} , определенное для СММ с множеством параметров МСП Θ_{MLP} , и использовать для обучения последовательность акустических векторов параметров $X = \{x_1, x_2, \dots, x_N\}$, размеченное в терминах состояний s_k , т.е. в момент времени n входным вектором для МСП является акустический вектор x_n с меткой $q_n = s_k$. Тогда можно показать [17][24][60], что если:

- МСП содержит достаточное количество скрытых нейронов, чтобы аппроксимировать функцию отображения входного вектора в выходной;
- МСП не «переобучен» («переобучение» выражается в слишком детальной адаптации весов к несущественным флуктуациям или нерегулярностям обучающих данных, что приводит к значительным погрешностям при распознавании);
- МСП после процедуры обучения находится достаточно близко к глобальному минимуму, то значение выходного вектора МСП — распределение вероятностей по дискретным СММ состояниям, обусловленное входным вектором

$$g_k(x_n, \Theta_{MLP}^{opt}) = p(s_k | x_n, \Theta_{HMM}), \quad (25)$$

где Θ_{MLP}^{opt} — множество параметров, полученное в результате обучения МСП. Кроме того, в [17] было предложено для использования контекстной информации применить сеть с задержкой по времени, на вход которой подавать последовательность из $2c + 1$ акустических векторов $X_{n-c}^{n+c} = \{x_{n-c}, \dots, x_n, \dots, x_{n+c}\}$. Тогда (25) можно переписать

$$g_k(x_n, \Theta_{MLP}^{opt}) = p(q_n = s_k | X_{n-c}^{n+c}, \Theta_{HMM}) \quad \forall k = 1, \dots, K. \quad (26)$$

Такое усовершенствование позволяет учитывать корреляцию акустических векторов, что позволяет преодолеть ограничения, связанные со статистической независимостью векторов наблюдений.

Кроме того, в [17] предложено использовать в качестве входного параметра СММ состояние, вычисленное на предыдущем временном шаге

$$g_k(x_n, \Theta_{MLP}^{opt}) = p(q_n = s_k | X_{n-c}^{n+c}, \Theta_{HMM}) \quad \forall k = 1, \dots, K. \quad (27)$$

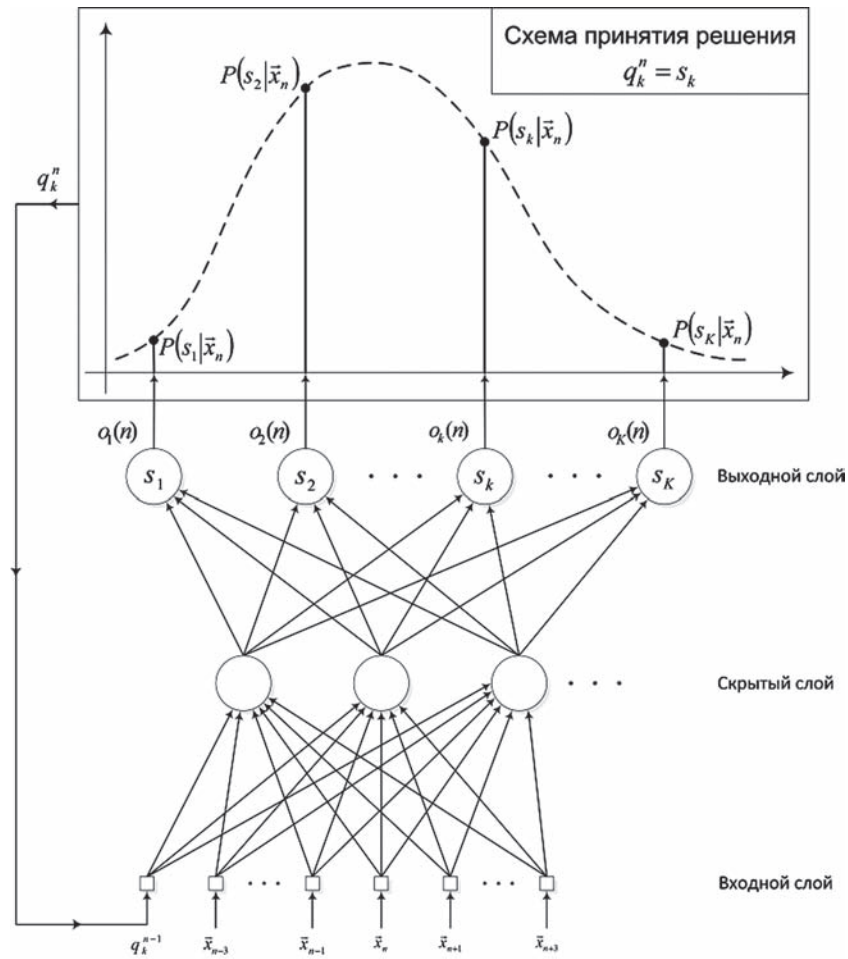


Рис. 6. Оценка вероятности с помощью TDNN сети

Предложенная вычислительная структура (рис. 6) работает следующим образом.

В каждый момент времени n на входной слой МСП подаётся последовательность акустических векторов X_{n-c}^{n+c} и СММ состояние на предыдущем временном шаге q_k^{n-1} , при этом на выходном слое будет формироваться распределение вероятностей по текущему состоянию СММ, обусловленное X_{n-c}^{n+c} и q_k^{n-1} .

Таким образом, поскольку выходной вектор МСП представляет собой аппроксимацию апостериорной вероятности, то $g_k(\mathbf{x}_n, \Theta_{MLP}^{opt})$ является оценкой

$$p(q_k | \mathbf{x}_n) = \frac{p(\mathbf{x}_n | q_k) p(q_k)}{p(\mathbf{x}_n)}, \quad (28)$$

которая неявно включает в себя эмиссионную вероятность $p(\mathbf{x}_n | q_k)$ и априорную вероятность СММ состояния $p(q_k)$. Поскольку вероятность $p(q_k)$ в (28) участвует как мультипликативный член, то это даёт возможность изменять априорную вероятность состояния во время классификации без переобучения персептрона, нормировать выходные вероятности персептрона в зависимости от используемого обучающего речевого корпуса данных. И тогда, чтобы правдоподобие $p(\mathbf{x}_n | q_k)$ можно было бы использовать в качестве эмиссионной вероятности для СММ, необходимо выход персептрона

Маковкин К.А.

Гибридные модели – Скрытые марковские модели/Многослойный перцептрон и их применение в системах распознавания речи. Обзор

$g_k(\mathbf{x}_n)$ поделить на относительную частоту встречаемости состояния s_k в обучающей выборке, что в результате даёт нам оценку выражения $\frac{p(\mathbf{x}_n | q_k)}{p(\mathbf{x}_n)}$. При распознавании масштабирующий член $p_k(\mathbf{x}_n)$ остаётся постоянным для всех состояний и не влияет на классификацию.

Аналогичная модель может быть построена с использованием рекуррентной нейронной сети [62][61][29][31], которая также используется для оценки эмиссионных вероятностей СММ.

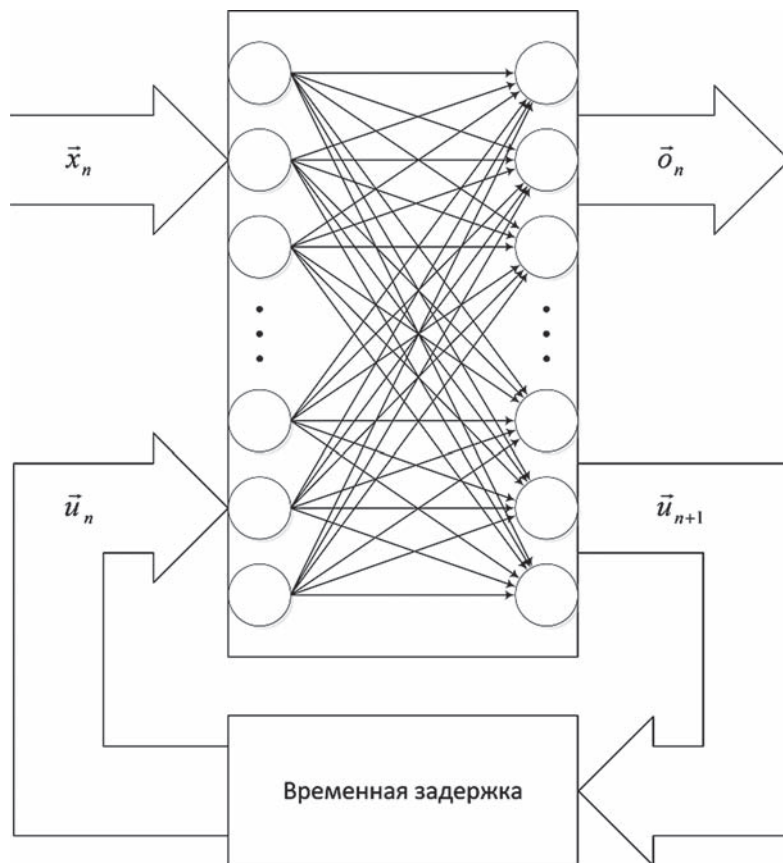


Рис. 7. Оценка вероятности с помощью RNN сети

Форма применённой рекуррентной сети (рис. 7) в данной модели впервые была предложена в работе [62]. В отличие от сети с задержкой по времени на вход МСП подаётся только один текущий акустический вектор параметров \mathbf{X}_n . При этом акустический контекст моделируется с помощью обратной связи между выходным и входным слоем, которая позволяет получить контекст большой длины. Эта обратная связь реализована в виде вектора текущего состояния \mathbf{u}_n . Таким образом, в каждый момент времени на вход сети поступает текущий вектор параметров и вектор текущего состояния. Далее сигналы, сформированные на входном слое, распространяются по МСП, в результате чего на выходном слое формируется выходной вектор \mathbf{O}_n и следующий вектор состояния \mathbf{u}_n . Формально это выглядит следующим образом:

$$\mathbf{z}_n = \begin{bmatrix} 1 \\ \mathbf{x}_n \\ \mathbf{u}_n \end{bmatrix}, \quad (29)$$

$$o_k^n = \frac{\exp(\mathbf{W}_k \mathbf{z}_n)}{\sum_j \exp(\mathbf{W}_j \mathbf{z}_n)}, \quad (30)$$

$$u_k^{(n+1)} = \frac{1}{1 + \exp(-\mathbf{V}_k \mathbf{z}_n)}, \quad (31)$$

где \mathbf{z}_n — комбинированный входной вектор, состоящий из \mathbf{X}_n и \mathbf{u}_n , а \mathbf{W} и \mathbf{V} — матрицы весов связей сети, которые используются для получения выходного вектора \mathbf{O}_n и вектора состояния $\mathbf{u}_{(n+)}$, соответственно. Включение 1 в (29) даёт возможность создать смещение для обеспечения нелинейности.

Аналогично модели Bourlard с использованием TDNN сети, выход рекуррентной сети — оценка апостериорной вероятности СММ состояния o_k^n в момент времени n :

$$g_k^n = P(q_k^n | X_1^n, \mathbf{u}_0). \quad (32)$$

Теоретические основания для такой интерпретации приведены в работе [41].

Как уже отмечалось выше, при использовании СММ необходимо сделать предположения, что наблюдения статистически независимы и марковский процесс первого порядка, т.е.

$$p(\mathbf{x}_n | Q_1^n, X_1^{n-1}) = p(\mathbf{x}_n | q_n^n), \quad (33)$$

где $Q_1^n = \{q_1, q_2, \dots, q_n\}$ — последовательность СММ состояний в моменты времени $t = 1, 2, \dots, n$. Использование рекуррентной сети позволяет сократить число предположений, т.е. исключить предположение о независимости наблюдений

$$p(\mathbf{x}_n | Q_1^n, X_1^{n-1}) = p(\mathbf{x}_n | q_n, X_1^{n-1}) \quad (34)$$

и учитывать акустический контекст для локальной модели наблюдений. Тогда, переформулировав (10) для модели M_i с учётом (34), получим

$$p(X_1^L | Q_1^L, M_i, A_i) = \prod_{l=1}^L p(\mathbf{x}_l | X_1^{l-1}) \frac{P(q_l | \mathbf{x}_l)}{P(q_l | X_1^{l-1})}. \quad (35)$$

Так как сомножитель $p(\mathbf{x}_l | X_1^{l-1})$ не зависит от последовательности фонов, то на этапе распознавания его можно игнорировать. Поскольку рекуррентная сеть используется для оценки $p(q_l | \mathbf{x}_l)$, то необходимо вычислить оставшийся член $P(q_l | X_1^{l-1})$. Один из простейших способов вычисления — предположить, что текущее состояние не зависит от наблюдаемого контекста [63], т.е.

$$P(q_l | X_1^{l-1}) = P(q_l), \quad (36)$$

где $p(q_l)$ — относительная частота встречаемости состояния q_l в обучающей выборке, т.е. получается результат, аналогичный модели Bourlard'a.

Обучение гибридной модели

Обучение гибридной модели заключается в оценке параметров как СММ, так и весов МСП. Пока не существует алгоритма, который бы позволил одновременно оценить оба множества параметров. Кроме того, поскольку для нейронной сети используется обучение «с учителем», то требуется значительный объём акустических данных, размеченных вручную, который в настоящее время также отсутствует. Bourlard предложил итерационную процедуру обучения, которая стартует с начальной разметки обучающих акусти-

Маковкин К.А.

Гибридные модели — Скрытые марковские модели/Многослойный перцептрон и их применение в системах распознавания речи. Обзор

ческих данных. На этих данных происходит обучение сети. Далее, совместно используя обученную сеть для оценки эмиссионных вероятностей и алгоритм Витерби, происходит переразметка обучающих данных. На полученной разметке снова происходит обучение сети и итерация повторяется. Начальная сегментация может быть получена с помощью стандартной СММ или просто делением последовательности акустических наблюдений на равные сегменты, причём каждый сегмент должен быть помечен соответствующим СММ состоянием. Аналогичный метод был предложен в [23].

Для гибридных моделей с рекуррентными нейронными сетями в [63] был предложен вариант обучения с использованием алгоритма Витерби для оценки параметров системы, который изложен ниже.

Параметры системы модифицируются, используя алгоритм Витерби для максимизации логарифма правдоподобия наиболее вероятной последовательности состояний для обучающих данных. Первый проход алгоритма Витерби делается, чтобы разметить последовательность векторов параметров в терминах СММ состояний. Затем параметры системы подстраиваются так, чтобы увеличить правдоподобие последовательности векторов параметров. Эта максимизация происходит в два этапа: максимизация эмиссионных вероятностей и максимизация транзитивных вероятностей.

Эмиссионные вероятности максимизируются с использованием метода градиентного спуска, а транзитивные вероятности — переоценкой моделей длительностей. Таким образом, обучающий цикл состоит из следующих шагов.

Шаг 1. Расстановка меток фонов на каждый фрейм обучающих данных. Эта начальная разметка обычно выполняется экспертом вручную.

Шаг 2. На основе ручной разметки строится модель длительности фонов и вычисляется априорная вероятность фона, которая используется для преобразования выхода рекуррентной сети в оценку правдоподобия.

Шаг 3. Аналогично производится обучение рекуррентной сети.

Шаг 4. Используя параметры, вычисленные на шаге 2, и рекуррентную сеть, обученную на шаге 3, выполняется разметка дополнительных обучающих данных и переход к шагу 2.

В экспериментах [63] было установлено, что для обучения достаточно четырёх итераций.

Результаты применения гибридных моделей

Использование гибридных моделей во многих экспериментальных системах приводило к росту точности распознавания по сравнению со стандартными СММ. Так, Bourlard и коллеги в период с 1988 г. по 1994 г. провели целый ряд успешных экспериментов по встраиванию гибридной модели в системы APP [17]. Например, в систему распознавания слитной речи DECIPHER [21], которая использовалась для задачи управления ресурсами проекта DARPA. Система DECIPHER представляла собой дикторо-независимую систему распознавания слитной речи, построенную на СММ. Размер словаря составлял 998 слов как с использованием модели языка для пар слов (перплексия — 60), так и без модели языка (перплексия — 998). Кроме того, использовали множество вероятностных производительных транскрипций для слов, фонологическое и акустическое моделирование кросс-слов, контекстно зависимые модели фонов с множеством плотностей вероятностей. При этом в системе DECIPHER были использованы как контекстно-независимые, так и контекстно-зависимые модели. В одном из экспериментов МСП был интегрирован в контекстно-независимую модель. Базовая система имела 69 моделей фонов с одним распределением эмиссионных вероятностей, а каждое слово имело одну произносительную транскрипцию. В качестве модели фонов использовалась модель «слева направо» с двумя или тремя состояниями и с параметрическим связыванием плотностей вероятностей для состояний. Эта гибридная модель сравнивалась с СММ системой DECIPHER, в которой эмиссионные вероятности моделировались Гауссовыми

смесями. При этом система DECIPHER использовалась в качестве стартовой системы для получения начальной фонетической разметки на первой итерации обучения МСП. В результате экспериментов было получено значительное улучшение качества распознавания по сравнению с контексто-независимой системой, основанной на СММ. Так, на одном из тестовых множеств — (February 91) гибридная контекстно-независимая модель продемонстрировала уровень ошибок 5.8%, что значительно лучше контекстно-независимой СММ модели, уровень ошибок которой составил 11% [46]. Кроме того, в одном из экспериментов была использована совместная оценка эмиссионных вероятностей как МСП, так и Гауссовыми смесями. Для комбинирования этих вероятностей было использовано несколько эвристик, например, вида

$$\log(P(x | q_j)) = \lambda_1 \log\left(\frac{P_{mp}(q_j | x)}{P(q_j)}\right) + \lambda_2 \log(P_{gn}(x | q_j)), \quad (37)$$

где P_{mp} обозначает вероятность, оцененную с помощью перцептрона, а P_{gn} — с помощью Гауссовых смесей. Набор коэффициентов λ_i был выбран одним для всех состояний. Такой способ оценки продемонстрировал наилучшее качество с уровнем ошибок порядка 5,5%.

Аналогичные эксперименты были проведены с гибридной моделью СММ и рекуррентной нейронной сетью. Гибридная модель была встроена в систему распознавания слитной речи ABBOT (Cu-Cop). Полученная система была успешно протестирована в рамках проекта November 1993 ARPA Wall Street Journal Test, а также в европейском проекте SQALE (Speech Quality Assessment for Linguistic Engineering)[76], посвящённом сравнению нескольких ведущих мировых систем распознавания таких, как Cu-Cop и Cu-HTK, созданных в Cambridge University Engineering Department (Великобритания), LIMSI из Laboratoire d'Informatique pour la Mecanique et les Sciences de l'Ingenieur (Франция) и PHILIPS the Man-Machine-Interface group with Philips Research Laboratories (Германия). Системы Cu-HTK, LIMSI и PHILIPS построены на базе СММ. Для акустико-фонетического моделирования они использовали непрерывные плотности, а система Cu-Cop — четыре рекуррентные нейронные сети [30]. Каждая сеть состояла из одного скрытого слоя. Её выходом для каждом акустического вектора параметров был вектор оценок вероятностей фонов, при этом в качестве обратной связи использовался 256-размерный вектор состояния, который заводился на входной слой. Полученные таким образом четыре вероятности с выхода каждой сети далее сливались в одну вероятность фона для каждого входного вектора параметров. При этом используемые рекуррентные сети обучались для оценки контекст-классов для каждого фона. Затем выходы такого оценщика сливались и умножались на контексто-независимую вероятность фона, чтобы получить постериорную контексто-зависимую вероятность. Контексты выбирались с использованием решающей процедуры на основе кластеризующего дерева [34]. В качестве модели языка в системе использовались триграммы и биграммы. Результаты сравнительных экспериментов для американского английского языка при использовании триграмм и биграмм приведены в таблице.

Таблица

Система	Триграммы	Биграммы
Cu-Cop	12,9%	17,0%
Cu-HTK	13,2%	16,7%
LIMSI	13,5%	17,2%
PHILIPS	14,7%	20,3%

Маковкин К.А.

Гибридные модели – Скрытые марковские модели/Многослойный перцептрон и их применение в системах распознавания речи. Обзор

Заключение

Описанные гибридные модели нашли применение во многих системах распознавания слитной речи с большими словарями и продемонстрировали очень неплохие результаты по сравнению с системами, построенными на основе каждой из моделей, составляющих гибридную модель, в отдельности. Исследования, проведенные с описанными системами, показали, что несмотря на относительную простоту структуры, они обладают целым рядом потенциальных преимуществ по сравнению со стандартными СММ.

Точность модели

Оценка вероятностей с помощью нейронной сети не требует детальных предположений о форме статистических распределений, которые должны быть промоделированы. В результате можно получить более точные акустико-фонетического модели.

Дискриминантная способность

С помощью нейронной сети значительно проще реализовать дискриминантное обучение.

Учитывание контекста

Поскольку описанные модели МСП могут использовать акустико-фонетический контекст, то локальная корреляция акустических векторов может быть учтена при вычислении распределений вероятностей. По различным причинам нечто подобное трудно реализовать в стандартных СММ.

Экономное использование параметров (снижение размерности системы)

Все распределения вероятностей представлены одним и тем же множеством разделяемых параметров. Хорошо известно, что более «экономично» моделировать границы между акустико-фонетическими классами, чем поверхности функций плотностей вероятностей или правдоподобий.

Гибкость

Использование нейронных сетей для оценивания эмиссионных вероятности позволяет легко сочетать разнообразные параметры, например, такие, как смесь нерывных и дискретных измерений.

Несмотря на достигнутые успехи, необходимо продолжать исследовательские работы, направленные на разработку гибридных структур, позволяющие проводить глобальное дискриминантное обучение, т.е. моделей, основанных на одновременном оценивании обоих множеств параметров как СММ, так и нейронной сети и с использованием одного критерия оптимальности. Кроме того, пока остаются открытыми вопросы, связанные с адаптацией таких систем, например, к диктору или к каналу связи. Также необходимо повышать устойчивость систем при работе в шумной обстановке.

Список литературы

1. *Елинек Ф.* Распознавание непрерывной речи статистическими методами // ТИИ-ЭР, 1976. Т. 64. № 4. С. 131–160.
2. *Колмогоров А.Н.* О представлении непрерывных функций нескольких переменных в виде суперпозиции непрерывных функций одного переменного и сложения // ДАН АН СССР. 1957. Т. 114. № 5. С. 953–956.
3. *Левинсон С.Е.* Структурные методы автоматического распознавания речи // ТИИ-ЭР, 1985. Т. 73. № 11. С. 100–128.
4. *Макхоул Дж., Рукос С., Гиш Г.* Векторное квантование при кодировании речи // ТИИЭР, 1985. Т. 73. № 11. С. 19–61.
5. *Осовский С.* Нейронные сети для обработки информации // Пер. с польского И.Д. Рудинского. М.: Финансы и статистика, 2002.
6. *Ту Д., Гонсалес Р.* Принципы распознавания образов // Пер. с англ. под ред. Ю.И. Журавлева. М.: Мир, 1987. 411 с.



7. Цыпкин Я.З. Обучение и адаптация в автоматических системах // М.: Наука, 1968. 400 с.
8. Almeida L.B. A Learning Rule for Asynchronous Perceptrons with Feedback in a Combinatorial Environment // Proceedings of the IEEE 1st International Conference on Neural Networks, 1987, vol. II. P. 609–618.
9. Baker J.K. The DRAGON system — An overview // IEEE Transactions on Acoustics, Speech and Signal Processing, 1975, vol. 23, ASSP-23, № 1. P. 24–29, 1975.
10. Bahl L.R. and Jelinek F. Decoding for channels with insertions, deletions, and substitutions with applications to speech recognition // IEEE Transactions on Information Theory, 1975, vol. 21. P. 404–411.
11. Baum L.E. An inequality and associated maximization technique in statistical estimation for probabilistic functions of Markov processes // Inequalities, 1972, vol. 3. P. 1–8.
12. Baum L.E., Egon J.A. An inequality with applications to statistical estimation for probabilistic functions of a Markov process and to a model for ecology // Bulletin of the American Mathematical Society, 1967, vol. 73. P. 360–363.
13. Baum L.E., Petrie T. Statistical inference for probabilistic functions of finite state Markov chains // The Annals of Mathematical Statistics, 1966, vol. 37. P. 1554–1563.
14. Baum L.E., Petrie T., Soules G., and Weiss N. A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains // The Annals of Mathematical Statistics, 1970, vol. 41, № 1. P. 164–171.
15. Bengio Y., Simard P., Frasconi P. Learning long-term dependencies with gradient descent is difficult // IEEE Transaction on Neural Networks, 1994, vol. 5, № 2. P. 157–166. (Special Issue on Recurrent Neural Networks, March 94).
16. Bourlard H., Morgan N. Continuous speech recognition by connectionist statistical methods // IEEE Transaction on Neural Networks, 1993, vol. 4, № 6. P. 893–909.
17. Bourlard H., Morgan N. Connectionist Speech Recognition. A Hybrid Approach // The Kluwer International Series in Engineering and Computer Science, 1994, vol. 247, Kluwer Academic Publishers, Boston.
18. Bourlard H., Morgan N. Hybrid connectionist models for continuous speech recognition // Lee C.H., Soong F.K., Paliwal K.K. (Eds), Automatic Speech and Speaker Recognition: Advanced Topics, The Kluwer International Series in Engineering and Computer Science, Kluwer Academic Publishers, Boston, USA 1996.
19. Bourlard H., Wellekens C.J. Speech pattern discrimination and multilayer perceptrons, // Computer, Speech and Language, 1989, vol. 3. P. 1–19.
20. Bourlard H., Wellekens C. Links Between Markov Models and Multilayer Perceptrons // IEEE Transactions on Pattern Analysis and Machine Intelligence, 1990, vol. 12, № 12. P. 1167–1178.
21. Cohen M., Murveit H., Bernstein H., Price P., Weintraub M. The DECIPHER speech recognition system // ICASSP-90, 1990, vol. 1. P. 77–80, Albuquerque.
22. Dempster A.P., Laird N.M., Rubin D.B. Maximum likelihood from incomplete data via the EM algorithm // Journal of the Royal Statistical Society, Series B, 1977, vol. 39, № 1. P. 1–38.
23. Franzini M.A., Lee K.F., Waibel A. Connectionist Viterbi training: a new hybrid method for continuous speech recognition // ICASSP-90, 1990, vol. 1. P. 425–428.



Маковкин К.А.

Гибридные модели – Скрытые марковские модели/Многослойный перцептрон и их применение в системах распознавания речи. Обзор

24. Gish H. A probabilistic approach to the understanding and training of neural network classifiers // ICASSP-90, 1990, vol. 3. P. 1361–1364.
25. Gori M., Bengio Y., R. De Mori. BPS: a learning algorithm for capturing the dynamical nature of speech // Proceedings of the International Joint Conference on Neural Networks, 1989, vol. 2. P. 417–423, Washington, DC, USA.
26. Haffner P., Franzini M.A., Waibel A. Integrating time alignment and neural networks for high performance continuous speech recognition // ICASSP-91, 1991, vol. 1. P. 105–108.
27. Haykin S.S. Neural Networks: A Comprehensive Foundation, Second Edition, Prentice Hall, 1999. Русский перевод: Хайкин С. Нейронные сети: Полный курс, Второе издание, Москва, «Вильямс», 2006.
28. Hecht-Nielsen R. Kolmogorov's mapping neural network existence theorem // IEEE First International Conference on Neural Networks, vol. III. P. 11–14, San Diego: SOS Printing.
29. Hochberg M. M., Renals S. J., Robinson A. J., Kershaw D. J. Large vocabulary continuous speech recognition using a hybrid connectionist-HMM system // Proceedings of CSLP, 1994, Yokohama. P. 1499–1502.
30. Hochberg M., Renals S. and Robinson A. ABBOT: the CUED hybrid connectionist-HMM large vocabulary recognition system // Proceedings of the Spoken Language Technology Workshop, 1995. P. 170–178, Austin, TX, USA.
31. Hochberg M. M., Renals S. J., Robinson A. J., Cook G. D. Recent improvements to the ABBOT large vocabulary csr system // ICASSP-95, 1995, vol. 1, pp. 62–72, Detroit, MI, USA.
32. Jelinek F. A fast sequential decoding algorithm using a stack // IBM Journal of Research and Development, 1969, vol. 13, Issue 6. P. 675–685.
33. Jelinek F., Bahl L.R., and Mercer R.L. Design of a linguistic statistical decoder for the recognition of continuous speech // IEEE Transactions on Information Theory, 1975, vol. 21. P. 250–256.
34. Kershaw D. J., Hochberg M. M., Robinson A. J. Context dependent classes in a hybrid recurrent network-HMM speech recognition system // Cambridge University Engineering Department, Technical Report, CUED/F-INFENG/TR. 217. 1995.
35. Lang K.J., Hinton G.E. The development of the time-delay neural network architecture for speech recognition // Technical Report CMU-CS-88-152, Carnegie-Mellon University, 1988.
36. Levin E. Word recognition using hidden control neural architecture // ICASSP-90, 1990, vol. 1. P. 433–436.
37. Levinson S.E., Rabiner L.R., and Sondhi M.M. An introduction to the application of the theory of probabilistic function of a Markov process to automatic speech recognition // Bell System Technical Journal, Apr. 1983, vol. 62, № 4. P. 1035–1074.
38. Lippmann R.P. Neural nets for computing // IEEE ICASSP-88, 1988, vol. 1. P. 1–6.
39. Lippmann R.P. Review of neural networks for speech recognition // Neural Computing, 1989, vol. 1. P. 1–38.
40. Lorentz G.G. The thirteenth problem of Hilbert // Browder F.E. (Ed), Proceedings of Symposia in Pure Mathematics, vol. 28, pp. 419–430. Providence, RI: American Mathematical Society.
41. McCullagh P., Nelder J. A. Generalized Linear Models // London: Chapman and Hall, 1983.
42. McCulloch W. S., Pitts W. H. A logical calculus of ideas immanent in nervous activity // Bulletin of Mathematical Biology, 1943, vol. 5, № 1–2. P. 99–115.
43. Minsky M., Papert S. Perceptrons // Cambridge: MIT Press. 1969.
44. Montana D.J. Training Feedforward Neural Networks Using Genetic Algorithms // Proceedings of the Eleventh International Joint Conference on Artificial Intelligence, Detroit, MI., 1989. P. 762–767.

45. *Morgan N., Boulard H.* Continuous speech recognition using multilayer perceptrons with hidden Markov models // ICCASP-90, 1990, vol. 1. P. 413–416.
46. *Morgan N., Boulard H.* Hybrid neural network/hidden Markov model system for continuous speech recognition // Intl. Journal of Pattern Recognition and Artificial Intelligence, Special Issue on Advances in Pattern Recognition Systems using Neural Networks (I. Guyon and P. Wang, Eds.), 1993, vol. 7, № 4.
47. *Morgan N., Boulard H.* Neural Network for Statistical Recognition of Continuous Speech // Proceedings of the IEEE, 1995, vol. 83, no. 5. P. 742–770.
48. *Niles L.T., Silverman H.F.* Combining hidden Markov models and neural networks classifiers // ICASSP-90, 1990, vol. 1. P. 417–420.
49. *Paul D.B., Baker J.K., Baker J.M.* On the interaction between true source, training and testing language models // ICASSP-91, 1991, vol. 1. P. 569–572.
50. *Pearlmutter B. A.* Learning state space trajectories in recurrent neural networks // Neural Computation, 1989, vol. 1, № 2. P. 263–269.
51. *Pearlmutter B. A.* Dynamic Recurrent Neural Networks // Technical Report CMU-CS-88–191, Carnegie-Mellon University, Computer Science Dept. Pittsburgh, PA. 1990.
52. *Peeling S.M. and Moore R.K.* Experiments in Isolated Digit Recognition Using Multi-Layer Perceptron // Technical Report 4073, Royal Speech and Radar Establishment, Malvern, Worcesber, Great Britain, 1987.
53. *Pineda F.J.* Generalization of Back-Propagation to Recurrent Neural Networks // Physical Review Letters, 1987, vol. 59. P. 2229–2232.
54. *Pineda F.J.* Recurrent back-propagation and the dynamical approach to adaptive neural computation // Neural Computing, 1989, vol. 1, № 2. P. 161–172.
55. *Prudencio R.B.C., Ludemir T.B.* Design of Neural Networks for Time Series Prediction Using Case-Initialized Genetic Algorithms // Proceedings of the 8th International Conference on Neural Information Processing, ICONIP, 2001. P. 990–995.
56. *Rabiner L.R.* A tutorial on hidden Markov models and selected application in speech recognition // Proceedings of the IEEE, 1989, vol. 77, no. 2, pp. 257–286. Русский перевод: Л.П. Рабинер. Скрытые Марковские модели и их применение в избранных приложениях при распознавании речи: Обзор. ТИИЭР. 1989. Т. 77. № 2 февраль. С. 86–120.
57. *Rabiner L.R., Juang B.-H.* Fundamentals of speech recognition. Prentice-Hall International, Inc. 1993.
58. *Rabiner L.R., Juang B.-H., Lee C.H.* An overview of automatic speech recognition, //~Lee C.H., Soong F.K., Paliwal K.K. (Eds), Automatic Speech and Speaker Recognition: Advanced Topics, The Kluwer International Series in Engineering and Computer Science, Kluwer Academic Publishers, Boston, USA 1996.
59. *Rahim M. R.* Artificial Neural Networks for Speech Analysis/Synthesis, Chapman and Hall, 1994.
60. *Richard M.D., Lippmann R.P.* Neural network classifiers estimate Bayesian a posteriori probabilities // Neural Computation, 1991, vol. 3, № 4. P. 461–483.
61. *Robinson T.* An application of recurrent nets to phone probability estimation // IEEE Transaction on Neural Networks, 1994, vol. 5, № 2. P. 298–305.
62. *Robinson A.J., Fallside F.* Static and dynamic error propagation network with application to speech coding // D.Z. Anderson (Ed.), Neural Informa-

Маковкин К.А.

Гибридные модели – Скрытые марковские модели/Многослойный перцептрон и их применение в системах распознавания речи. Обзор

tion Processing System, American Institute of Physics, New York, Denver, CO, 1988. P. 635–641.

63. *Robinson T., Hochberg M., Renals S.* The use of recurrent neural networks in continuous speech recognition // C.H. Lee, F.K. Soong, K.K. Paliwal (Eds), *Automatic Speech and Speaker Recognition: Advanced Topics*, The Kluwer International Series in Engineering and Computer Science, Kluwer Academic Publishers, Boston, USA 1996.

64. *Rosenblatt F.* *Principles of Neurodynamics* // Spartan Books, New York, 1959. Русский перевод: *Розетблатт Ф.* *Принципы нейродинамики (перцептрон и теория механизмов мозга)*. М.: Мир, 1965. 480 с.

65. *Rumelhart D. E., Hinton G. E., Williams R. J.* Learning internal representations by error propagation // Rumelhart, D. E., G. E. Hinton, (eds.), *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, 1986, vol. 1: Foundations., chapter 8, Bradford Books/MIT Press, Cambridge, MA.

66. *Rumelhart D. E., Hinton G. E. and Williams R. J.* Learning representations of back-propagation errors // *Nature (London)*, 1986, vol. 323. P. 533–536.

67. *Rumelhart D. E., Hinton G. E. and Williams R. J.* Interactive Processes in Speech Perception: The TRACE Model // *Parallel Distributed Processing: Vol. 2, Psychological and Biological Models*, eds. D. E. Rumelhart and J.L. McClelland. Cambridge, MA: MIT Press. 1986.

68. *Rumelhart, D.E., McClelland J.L. and the PDP Research Group.* *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, 1986, vol. 1: Foundations, Bradford Books/MIT Press, Cambridge, MA.

69. *Sato M.* A real time learning algorithm for recurrent analog neural networks // *Biological Cybernet.*, 1990, vol. 62. P. 237–241.

70. *Sawai H., Waibel A., Miyatake M., Shicano K.* Spotting Japanese SV-syllables and phonemes using time-delay neural networks // *ICASSP-89*, 1989, vol. 1. P. 25–28.

71. *Tank D.W., Hopfield J.J.* Concentrating information in time: analog neural network with application to speech recognition problems // *International Conference on Neural Networks, ICNN-87*, 1987. P. 455–468.

72. *Waibel A.* Modular construction of time-delay neural networks for speech recognition // *Neural Computing*, 1989, vol. 1. P. 39–46.

73. *Waibel A., Hanazawa T., Hinton G., Shikano K., Lang K.* Phoneme Recognition Using Time-Delay Neural Networks // *IEEE Transaction on Acoustic Speech Signal Processing*, 1989, vol. 37, № 3. P. 328–339.

74. *Waibel A., Sawai H., Shikano K.* Modularity and scaling in large phonemic neural networks // *IEEE Transaction Acoustic Speech Signal Processing*, 1989, vol. 37. P. 1888–1898.

75. *Williams R. J., Zipser D.* A learning algorithm for continually running fully recurrent neural networks // *Neural Computation*, 1989, vol. 1, № 2. P. 270–280.

76. *Young S.J., Adda-Dekker M., Aubert X.* Multilingual large vocabulary speech recognition: the European SQALE project // *Computer Speech and Language*, 1997, vol. 11. P. 73–89. <http://htk.eng.cam.ac.uk>

Сведения об авторе

Маковкин Константин Александрович —

окончил Московский государственный технический университет им. Н.Э. Баумана в 1990 г. С 1990 года сотрудник Вычислительного центра им. А.А. Дородницына РАН. Область интересов: разработка систем автоматического распознавания речи; цифровая обработка сигналов; скрытые марковские модели; модели нейронных сетей; VoIP протоколы. E-mail: k.makovkin@gmail.com

Философия в эпоху компьютеров. Несколько комментариев на тему философского измерения информационных технологий

Мариуш Шынкевич, кандидат наук



Статья посвящена философскому осмыслению информатики. В частности — метатеоретическим проблемам развития компьютерных технологий.

• философия • наука • технология • компьютер.

This text pertains to philosophical aspects of computer science and information technologies. Particularly, metatheoretical aspects of the computer science are discussed here from the perspective of philosophy of science and philosophy of technology.

• philosophy • science • technology • computer.

На формирование западной цивилизации существенное влияние оказали два фактора: наука и техника. В информационном обществе знание воспринимается не только как категория товара, но и как инструмент управления. Подтверждение этому можно найти в работе британского философа Фрэнсиса Бэкона «Новый органон», где он писал, что власть и познание неразрывно связаны между собой, идентифицируя познание с областями науки и техники [1]. Изменение техники, будучи следствием развития науки, следует рассматривать как в контексте познания (описание, объяснение), так и в практическом плане (прогноз, приложения).

В настоящее время всё сложнее провести границу между областями науки и техники. Их связь неоднократно принимает диалектический характер: наука создаёт теоретические основы для развития техники (например, исследования из области ядерной физики и развитие атомной энергетики), техника же предоставляет инструменты, разрешающие прогресс базовых наук (использование новых измерительных и сенсорных устройств). Однако философия науки XX в. концентрировалась, прежде всего, на базовых дисциплинах (в основном на теоретической физике). Поэтому прикладные науки и техника, в широком её понимании, до сих пор требуют метауровневого анализа. Это становится существенным в контексте развития компьютерных наук — области, в развитии которой чисто теоретические вопросы неразрывно соединяются с проблемами практического характера.

Философия и информатика

Наука — явление комплексное и многогранное. Попытки её систематизации предпринимались с древних времён. Оригинальные классификации наук предлагали Аристотель, Ф. Бэкон, Т. Гоббс, Г.В. Лейбниц, Д. Юм, Г.В.Ф. Гегель, О. Конт, Ф. Энгельс. В настоящее время, принимая за *principum divisionis* онтологические критерии, предмет исследования, а также методологические — прикладные методы в пределах конкретной области знания, можно разделить науки на две основные категории: формальные (логика, математика) и эмпирические, которые, в свою очередь, подразделяются на естественные (физические и биологические), гуманитарные и общественные науки [2. С. 46–53]. Рядом с классическими науками находятся пограничные науки (физическая химия, геофизика), а также междисциплинарные науки (когнитивная наука, социология). Современную классификацию замыкают прикладные науки. Все предыдущие исследования ведущих философов науки (Э.Мах, Р. Карнап, К.Р. Поппер, Э. Нагель, Т. Кун, И. Хакинг) концентрировались в основном на классических науках.

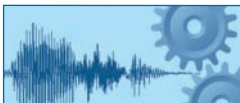
Большая часть исследователей относилась к технике как к незначительной, с философской точки зрения, сфере. Чаще всего техника описывалась как сфера практического применения теоретических знаний либо как источник потенциальной опасности. Эта ситуация изменилась после Великой индустриальной революции. Техника в то время стала темой для множества философских анализов. Тем не менее, эти размышления концентрировались, прежде всего, на общественных и цивилизационных эффектах технологического развития.

Следует задуматься над методологическими аспектами техники и прикладных наук, среди которых особое значение в последние десятилетия приобрели компьютерные науки. В первую очередь, зададим вопрос о характере и специфике явления, которое некоторые авторы именуют «информационной революцией». Эту проблему следует рассматривать в двух измерениях: внутреннем, касающемся изменений, происходящих в пределах самих информационных технологий, и внешнем, относящемся к трансформации общественных и цивилизационных явлений, вызванных развитием информационных технологий. Этот анализ должен быть проведён в двух разных плоскостях: подробной, предмет которой составляет компьютер как определённый тип машины, и обобщённой, связанной с развитием сетевых технологий (Интернет). Используя описательные инструменты, функционирующие на почве современной философии техники, постараемся описать особенность компьютеров в качестве специфических технических устройств (так называемых диалектических машин).

Анализ философского и методологического измерения информационных технологий — не простая задача. Во-первых, информатика — относительно молодая наука, во-вторых, она характеризуется большой внутренней сложностью. В настоящее время специалисты выделяют, как минимум, три основных измерения информатики: техническое (уровень *hardware*), теоретическое (уровень *software*) и культурное (информационное содержимое — контент) [3]. Каждое из них требует своих философских компетенций и употребления различного понятийного аппарата. Как следствие, теоретические соображения, касающиеся информатики, приобретают при необходимости междисциплинарный характер. Итак, сконцентрируемся на техническом уровне, главную онтологическую основу которого составляют компьютерные устройства.

Компьютер в философии науки и философии техники

Специфику компьютерных устройств можно описать, обратившись к концепции бельгийского философа Х. ван Лиера. Интересно, что эта концепция родилась в 60-е гг. прошлого века, т.е. за несколько десятилетий до возникновения сетевых технологий и распространения Интернета. Философ разделил технические устройства, которые появлялись на протяжении веков, на три категории: а) статические машины; б) динамические машины; в) диалектические машины. Каждое из поколений машин



было связано с определённой эрой технологического развития европейских сообществ: статические машины — эотехническая эра (до середины XVIII в.), динамические машины — палеотехническая эра (XIX и XX вв.), диалектические машины — неотехническая эра (вторая половина XX в.) [4]. По мнению ван Лиера, настоящая эпоха технологического развития начинается в X в. н.э. Устройства этого периода (рычаги, водяные мельницы) были сконструированы, прежде всего, с целью увеличения физической силы человека. Они находились в непосредственной зависимости от природных условий. Их функционирование не порождало существенных философских проблем.

Эта ситуация подверглась значительному изменению во время Великой индустриальной революции, символом которой стали динамические машины. Философы, до того концентрирующиеся на онтологических и эпистемологических соображениях, стали замечать значение практического измерения науки, философские и общественные последствия развития техники. Новый тип машин перестал быть единым с природой инструментом, облегчающим работу. Описываемый класс устройств уже не имел такой тесной связи с природными условиями. Динамические машины делились на два принципиальных класса: а) энергетические машины (например, паровой двигатель Ватта), которые служили не столько преобразованию природной энергии (например, энергия воды, превращённая в механическую силу мельничного колеса), сколько производству энергии; б) упорядочивающие машины, которые благодаря энергии, производимой энергетическими машинами, и соответствующей конструкции были способны на создание новых артефактов (ткацкая мастерская, типография). Ван Лиер включил в эту категорию также использующие электричество носители информации (радио и телеграф).

Развитие новых технологий вызвало ряд коллизий общественной и мировоззренческой природы. Символом этих противоречий стало луддистское движение, представители которого совершали акты уничтожения машин. Действия луддистов вскоре стали серьёзной экономической проблемой. С целью противодействия подобным актам в 1812 г. Британский парламент принял закон (*Frame Breaking Act*), разрешающий смертную казнь активистов движения за уничтожение машин [5]. С этого времени мир технических устройств навсегда обосновался в философских дискуссиях. На тему «Внедрение техники в мире», как оптимистически, так и пессимистически, высказывались величайшие европейские мыслители: К. Маркс, А. Бергсон, Ф. Ницше и М. Хайдеггер.

Последнее — третье — поколение технических устройств — составляют диалектические машины, в том числе современные компьютеры. Диалектические машины уже не так явно выделяются из среды, как их динамические предки. Их природную черту составляет стремление к синергии и отсутствие строгой специализации. Это отклонение от явной специализации становится существенно в случае компьютеров, основной чертой которых является то, что американский теоретик информационной этики Дж.Х. Мур назвал *логической пластичностью* [6]. Мур обращает внимание на универсальность компьютеров. По его мнению, они достаточно приспособлены к выполнению любого действия, которое можно охарактеризовать в категориях входа, выхода и объединяющих их логических операций. Однако мы должны помнить, что анализ философского значения компьютерных технологий происходит на двух уровнях: отдельные цифровые машины следует анализировать иначе, чем компьютеры, функционирующие в распределённых сетевых системах. Уже ван Лиер заметил, что только онтология распределённой сети позволяет проявить настоящий характер

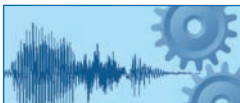


этих устройств. В диалектических машинах особую роль играет совокупность составляющих их элементов. Редукционизм динамических машин, в которых каждая часть выполняла строго определённую задачу, уступает место холизму устройств нового поколения [7, 8]. В этих машинах отдельные элементы спроектированы так, чтобы облегчить работу других компонентов (например, современные авиаконструкции). В контексте их функциональности существенным становится фактор синергии. Синергии подразделяются на следующие типы: а) машина/окружающая среда (аэродинамические конструкции), б) форма/материя (логические свойства транзисторов и физические свойства полупроводников), в) машина/машина (комплексные промышленные системы), г) машина/человек (конструкция компьютерной клавиатуры).

Важное отличие диалектических машин — открытость к связи (коммуникациям) и информационным потокам. Это относится как к синергии машина/машина, так и машина/человек. Качественное измерение функционирования устройств этого поколения связано с понятием сети, в контексте которой становится понятен характер типичной для этой эпохи синергии машина/машина. Полная функциональность диалектических устройств проявляется только на уровне децентрализованной сети. В этом случае комплексный подход к сети становится более существенным, чем проводимый в духе редукционизма анализ отдельных её элементов. Современные сети характеризуются специфической онтологической конструкцией. В отличие от централизованных систем, типичных для динамических машин (центральный источник энергии и использующие его периферийные устройства), они имеют распределённый характер. В новой архитектуре сети центробежная логика начинает преобладать над центростремительной. Функциональность диалектических машин связана также с синергиями машина/человек. Компьютерные устройства всё чаще проектируются в контексте их удобства для пользователя. Эта зависимость имеет обратный характер. Как подчёркивают исследователи познавательных процессов, логика гипертекста порождает в самих получателях, особенно в младших поколениях интернет-пользователей, определённые познавательные навыки: мы запоминаем не столько содержание информации, сколько её местонахождение в сетевых ресурсах.

Современные компьютерные устройства следует отнести к группе диалектических машин, в работе которых главное значение получают две группы синергий: машина/машина (сетевая система) и машина/человек. Новейшие устройства неотехнической эры служат не только для пассивной обработки и передачи информации, как это было в случае использования радио и телевидения, но и обладают очень важной чертой — программируемостью. По мнению некоторых исследователей, именно она является критерием, разрешающим принять данное устройство за компьютер. Однако программируемость ведёт к двум важным последствиям. Во-первых, машины, зависящие от внешней информации, должны иметь способность сохранять внутреннюю стабильность, гарант которой — реализованная в устройстве программа. Во-вторых, они уже не являются искусственными созданиями — посредниками между человеком и природой. Их онтологический статус претерпевает значительное изменение: диалектические машины становятся новым типом окружающей среды (виртуальная реальность). Человек неотехнической эпохи всё чаще вынужден добывать информацию в мире техники, примером этого служат эксперименты с компьютерной поддержкой, а также феномен компьютерных систем научных открытий [9].

Философия техники XX в., прежде всего, концентрировалась на следствиях технологического развития и его влиянии на функционирование окружающей действительности. Экспансия компьютеров и возникновение информационного общества привели к появлению новых предметных областей. Современные технологии вместе с разнообразным их практическим использованием порождают ряд проблем познавательной, социальной и этической природы. Уже в самом начале можно выделить, как минимум, четыре принципиальных типа социальных реакций, появляющихся в ответ на развитие компьютерных технологий. Часть из них связана с самой концепцией кибернетизации, другая — с общей идеей информационного общества. Оптимисты



стараясь акцентировать внимание на позитивных сторонах процессов компьютеризации и распространения сети Интернет. Пессимисты обращаются к апокалиптическим точкам зрения прогрессирующего внедрения технологий и инструментализации этических ценностей. Крайняя версия описываемого положения приняла вид *неолуддизма*, представители которого (К. Сале и К. Ласн) указывают на общественную угрозу, моральную и этическую, вытекающую из процесса информатизации мира [10]. Сторонники нейтральной точки зрения подчёркивают аксиологическую нейтральность новых технологий, отдавая связанные с ними этические вопросы на решения самих пользователей. Последняя из групп подчёркивает, однако, амбивалентный характер компьютерных технологий, которые, по её мнению, имеют как позитивные, так и негативные стороны.

Один из самых существенных философских аспектов развития цифровых технологий соединяется с процессом децентрализации и распространения сети в мире — то же происходит в общественных и политических измерениях. Эти перемены непосредственно относятся к процессам глобализации. По мнению испанского социолога М. Кастельса, появление новых областей реальности (*cyberreality*) и изменение онтологической специфики реального мира (распространение сети) требуют модификации прежних систем ценностей и традиционных культурных тенденций. Эти проблемы были описаны Кастельсом в трёх работах: *The Rise of the Network Society* [11], *The Power of Identity* [12] и *End of Millennium* [13].

В общеполитической перспективе большое значение приобретают также множественные мифы, касающиеся возможностей современных информационных устройств. Они возникают, например, из ошибочного понимания реального смысла муровского тезиса о логичной пластичности компьютеров и безосновательной веры в их постулируемую универсальность. Другой миф касается уверенности в почти безграничных возможностях вычислительных машин. В настоящее время известно, что компьютерные технологии имеют множественные внутренние ограничения, которые можно разделить на физические (например, закон Мура [15]) и формальные [14].

Кроме того, компьютерные технологии порождают ряд важных онтологических вопросов. Статические машины были только третьим универсумом — посредником между природой и человеком. Динамические машины выделились из мира природы впечатляющим образом. Онтологический статус мира техники изменился ещё больше в современной нам неотехнической эре. Диалектические машины создают область новой реальности. Это отношение, однако, имеет диалектический характер — современная техника подвергается специфически понимаемой натурализации, поскольку внедряется в саму действительность. Одними из субъектов процесса внедрения техники являются его цифровизация и виртуализация, которые достигаются, в основном, с помощью сетевых систем (Интернет). Как правильно замечает ван Лиер: «Каждый отдельно взятый технический предмет является мерой какого-нибудь действия или какого-нибудь другого предмета, но сеть, как целое, равнозначна миру» [4. С.81]. Компьютер, как и все ранее сконструированные технические устройства, воспринимается как инструмент, служащий расширению человеческих компетенций. Однако в отличие от статических и динамических машин, в развитии компьютерных технологий речь идёт не о физической силе, а об интеллектуальных способностях пользователей. Это касается как механических счётных машин (машина Паскаля, машины Лейбница и Бэббиджа), так и современных, программированных компьютерных устройств.

Развитие компьютерных технологий порождает множество проблем из области философии науки и эпистемологии. Одна из самых важных проблем свя-



зана с методологическим статусом самой информатики — науки, в которой основные исследования неразрывно связаны с практикой. Наука и техника, хоть и становятся всё более зависимыми друг от друга, тем не менее, всё так же остаются отдельными областями культуры. В фундаментальных науках приоритет отдаётся познавательным вопросам — мы стремимся к открытию абстрактных законов и формулированию теорий, описывающих определённые классы явлений. В прикладных науках, которые сами по себе не являются методологически однородным целым, приоритет отдаётся практическим вопросам. И хоть прикладные науки функционируют в пределах основных законов, однако их целью не являются ответы на вопросы «Как?» и «Почему?», скорее, это поиск способов результативного решения практических проблем. Важной проблемой является вопрос критериев оценки и адекватности продуктов информационных технологий. В случае фундаментальных дисциплин критерии оценки адекватности или правдивости их законов и теорий, в зависимости от принятой методологической ориентации, относительно понятны. Однако в прикладных науках эти принципы имеют другой, менее объективный характер, связанный с критической адекватностью, фундаментальностью, экономическими факторами, а также удобством и эстетикой. Следующая проблема относится к значению термина «информационная революция». Если развитие информатики доходит до непредвиденных случаев — революционных, то существенное значение приобретает модель, описывающая такие явления, например, в контексте суждений Т. Куна, И. Хакинга и С. Шапина. В рамках этих моделей динамика развития компьютерных наук рассматривается как в контексте предложенного авторами разделения на внутренние и внешние революции, так и по отношению к факторам, влияющим на развитие этой области знания.

Проведённый к настоящему времени анализ наиболее распространённых методологических моделей позволяет сомневаться в их адекватности. Кажется, что ни постпопперовские, ни посткуновские идеи не описывают правильно изменений, достигнутых в информационных технологиях. Общим недостатком обеих самых важных парадигм философии науки XX в. является тот факт, что возникли они как попытки уловить специфику фундаментальных наук. В то же время информатика — дисциплина, принадлежащая к другой методологической области — области прикладных наук. В ней чисто теоретические вопросы смешиваются с проблемами практической природы. Информатика — наука, в которой существенную, если не самую важную, роль играют технические устройства. Новая методология, способная описать компьютерные науки, потребует совместной работы в разных философских субдисциплинах, главным образом в философии науки и в философии техники.

Анализ развития компьютерных наук привносит много нового в такие традиционные разделы философии, как онтология (вопрос о статусе виртуальной действительности и значение понятия информации), эпистемология (проблема компьютерной науки, вопрос источников знания) и этика (киберпреступность, проблема эстетических ценностей в виртуальном мире, статус хакерской этики). Другая важная задача — необходимость ответить на вопрос о том, описывают ли подобные методологические схемы изменения, происходящие как в направлении software (теоретический аспект), так и в направлении hardware (материальная область). С точки зрения философа науки, интересными кажутся также качественные фазы развития самих компьютерных устройств. Каждая из них связана с техническими вопросами (очередные поколения компьютеров), функциональными аспектами (изменения, происходящие в пределах интерфейса) и аспектами приложений (компьютеры как инструменты). Эволюция компьютеров в функциональном аспекте проходила, как минимум, в три этапа: а) компьютеры в качестве счётных машин; б) приборы, обрабатывающие данные; в) компьютеры, основную задачу которых составляют коммуникация и функционирование в новой онтологической структуре виртуального мира. Следующим этапом этих революционных изменений могло бы быть возникновение компьютеров, наделённых сознанием (проблема искусственного интеллекта). На значение этой очередной важной проблемы обращают внимание сами информатики. В практическом измерении всё более существенным становится не то, как сконструированы информационные



ресурсы (программа или инструмент), а то, как они будут применены пользователями. Производной описываемого здесь вопроса является всё более заметная специализация отрасли информатики.

Последний вопрос, на который стоит обратить внимание, связан с проблемой методологической специфики компьютерных экспериментов. Опыт такого рода играет огромную роль в естественных науках (например, химии и физике). К сожалению, функционирующие в современной философии науки методологии экспериментальных исследований не описывают эти познавательные важные вопросы. Даже самая важная из методологических интерпретаций научного эксперимента — *новый экспериментализм* (И. Хакинг) — игнорирует важность компьютеров в лабораторной практике. Интересным вопросом является сфера компьютерной науки. Это проблема, которую польский философ Л. Колаковский заключил в вопросе: «Способны ли компьютеры к самостоятельному занятию наукой? Если «Да», то в каких областях науки» [16].

Резюме

XX в. часто называют «Столетие науки». Крупные парадигмы естественных наук, такие как квантовая механика, общая теория относительности или синтетическая теория эволюции, стали своеобразными символами современности. В настоящее время мы всё чаще сознаём тот факт, что переломный момент, изменивший облик современного мира наиболее радикальным способом, был тот, что ассоциировался с информационной революцией. Цифровые технологии способствовали перестройке способа функционирования современного мира как в его социальных аспектах (внедрение Интернета, глобализация, электронная экономика), так и онтологических (например, феномен виртуальной действительности). В настоящее время сложно указать такую сферу человеческой культуры (материальной или духовной), в которой бы мы не замечали изменений, вызванных развитием компьютерных технологий. Мир понимаемых в широком смысле информационных технологий — источник фундаментальных проблем философской природы. В прошлом всё техническое противопоставлялось естественному, а область техники виделась как мост между миром людей и универсумом природы. В настоящее время, благодаря специфике диалектических машин, эта ситуация подверглась значительной трансформации: мир техники уже не является отдельным бытием. Он, скорее, новый тип окружающей среды. Это приводит к необходимости изменения способа понимания и определения многих традиционных философских категорий. По мнению ван Лиера, в эпоху статических и динамических машин именно природа играла роль одной из главных основ мира ценностей, равно как эпистемологических, так и этических. Однако в новой ситуации изменению подвергаются онтологические основы. Философы техники и философы науки пытаются определить и выяснить проблемы, связанные с развитием компьютерных технологий. Основная задача первых — понятие природы кибердействительности и отношения, связывающего мир компьютеров с миром людей, а также попытка выявления потенциальной опасности ситуации и возможностей, которые связаны с компьютеризацией мира и внедрением Интернета. Философы науки должны будут понять методологическую специфику информационных технологий, а также описать их внутреннюю сложную структуру. Это не простая задача, так как, анализируя информатику, в настоящее время мы сталкивались со сложной и быстро развивающейся областью знания.

Итак, метатеоретическое видение компьютерных наук должно принимать во внимание динамику их развития. Кроме того, прогрессирующая компью-



теризация действительности вынуждает нас к серьёзному занятию маргинализированными в методологическом отражении прикладными науками, в качестве наиболее интересного примера которых выступает современная информатика. Рождение цифровых технологий привело к созданию нового типа реальности, философское измерение которого формируется параллельно с развитием самой информатики. Важность этого процесса определяется, во-первых, способом, которым компьютерные технологии изменяют окружающую действительность, во-вторых, философской уникальностью информационной революции — революции, последствия которой мы стараемся понять.


Список литературы

1. Bacon F. The New Organon or: True Directions Concerning the Interpretation of Nature. http://www.constitution.org/bacon/nov_org.htm
2. Such J., Szcześniak M. Filozofia nauki. Poznan, 2006.
3. Gawrysiak P. Rewolucja cyfrowa, Rozwój cywilizacji informacyjnej, Warszawa, 2008.
4. Van Lier H. Nowy wiek. Warszawa, 1970.
5. Fox N. Against the Machine: The Hidden Luddite Tradition in Literature, Art and Individual Lives Island Press 2002.
6. Moor J. H. What is computer ethics? (in.) Metaphilosophy, vol 16, No. 4, Oxford and New York, 1985.
7. Szynekiewicz M. Rozwój technologii komputerowych w świetle filozoficznej refleksji nad nauką i techniką. W: Czas informacji nr 3(8)/2011.
8. Szynekiewicz M. Rewolucja informatyczna XX wieku i jej kontekst filozoficzno — społeczny, (w:) A. Kapczyński, S. Smugowski, Współczesne systemy informatyczne i ich zastosowania. Katowice, 2010.
9. Giza P. Filozoficzne i metodologiczne aspekty komputerowych systemów odkryć naukowych. Lublin, 2006.
10. Sale K. Rebels Against the Future: The Luddites and Their War on the Industrial Revolution: Lessons for the Computer Age. Addison Wesley, 1995.
11. Castells M. The Rise of the Network Society, The Information Age: Economy, Society and Culture, Vol. I. Cambridge, MA; Oxford, 2000.
12. Castells M. The Power of Identity, The Information Age: Economy, Society and Culture. Vol. II. Cambridge, MA; Oxford, 2004.
13. Castells M. End of Millennium, The Information Age: Economy, Society and Culture. Vol. III. Cambridge, MA; Oxford, 2000.
14. Harel D. Algorithmics: The Spirit of Computing. Addison Wesley, 2004.
15. Moore G. E. Cramming more components onto integrated circuits. W: Electronics Magazine 38 (8), 19.04.1965 (09.08.2011).
16. Kołakowski L. Moje wróżby w sprawie przyszłości religii i filozofii. W: Gazeta Wyborcza, (15.01.2010).

Сведения об авторе

Мариуш Шынкевич —

кандидат наук. Польский философ, специалист по философии и этике науки и техники. Университетский преподаватель и заместитель директора Института философии (Университет им. А. Мицкевича в Познани).



Методика тестирования систем автоматического синтеза и распознавания речи в целях определения коммерческой целесообразности их использования

*Корсакова Н.С., Засыпкина К.А.
ЗАО «ЭсТуЭс Некст»*

В данной статье рассматривается опыт создания тестового материала для определения качества систем синтеза и распознавания речи, а также демонстрируются сравнительные результаты тестирования нескольких промышленных систем с помощью этого материала.

• синтез речи • распознавание речи • методика тестирования • тестовый материал • база данных.

A new testing method of automatic speech synthesis and recognition systems is proposed. The authors focus mainly on the problem of choice of testing material which would be efficient for testing and comparison of commercial systems. Several particular speech synthesis and recognition systems are tested, results are presented and compared, so some conclusions concerning commercial viability of their use can be made.

• speech synthesis • speech recognition • testing method • testing material • database.

Введение

Рынок речевых технологий и средств компьютерной обработки речи — один из самых быстрорастущих на сегодняшний день. По данным компании J'son & Partners, он оценивается примерно в 3 млрд. долларов. Рост рынка, согласно аналитикам Voice Information Associates, составляет около 25% в год [1]. В настоящее время речевые технологии — это, прежде всего, технологии автоматического распознавания речи (ASR) и автоматического синтеза речи (TTS).

Выбор системы ASR или TTS для применения в какой-либо определенной сфере деятельности должен быть тщательным и взвешенным. Такая необходимость обусловлена следующими факторами:

- 1) денежные затраты. Ошибка в выборе техники (метода или алгоритма), которая будет использована в будущем приложении, может привести к лишним денежным затратам, которые могут быть вызваны необходимостью ликви-

дировать (сгладить) ошибки и неточности работы системы; исключением из результата работы системы лишней информации, переплатой за счёт реализации ненужных функций и т.д.;

2) временные затраты. При внедрении сложной технологии для реализации простых функций придётся затратить гораздо больше времени, чем требовалось бы, если бы разработчик ограничился только необходимыми методами и алгоритмами;

3) объём занимаемой памяти. При неверном выборе метода для реализации системы обработки речи возможны излишние затраты памяти для хранения информации, баз знаний, которых можно было бы избежать. Это, в свою очередь, приводит к избыточным денежным затратам;

4) качество. Существующие системы автоматического синтеза и распознавания речи различаются по качеству синтеза/распознавания. При выборе той или иной системы необходимо определить тип приложения, в которое будет внедрена технология, так как требования к качеству системы должны зависеть от сферы применения. Качество, в свою очередь, влияет на вышеперечисленные факторы, поэтому целесообразно перед выбором системы проверить ее работу на практике.

Несмотря на большое число разработок, проблемы синтеза и распознавания речи до сих пор считаются нерешенными, так как качество синтеза и распознавания только в отдельных случаях можно считать удовлетворительным и хорошим. Улучшению качества автоматического синтеза речи препятствует сложность разрешения языковой неоднозначности при автоматическом анализе текста, который используется в синтезе устной речи для расстановки пауз; определения главноударного слова в предложении; задания интонации вопроса, восклицания; для правильной расстановки ударения в словах [2]. На качество автоматического распознавания речи оказывают влияние условия прикладной области, в частности, состав и размер словаря. Причина невысокого качества распознавания кроется в вариативности речевого сигнала, которая обуславливается, например, индивидуальными особенностями дикторов, характеристиками каналов связи, а также влиянием окружающей обстановки [3].

Для оценки качества рассматриваемых систем обработки речи была разработана методика тестирования, которая может быть применена для оценки любых продуктов ASR и TTS. Данная методика направлена на оценку широкого класса систем автоматического синтеза и распознавания речи и не ограничена определённой предметной областью или сферой применения.

Тестирование систем ASR

Тестовый материал для систем автоматического распознавания речи состоит из двух частей: лексической и синтаксической. Лексическая часть предназначена для проверки точности распознавания изолированных слов, а синтаксическая – для проверки точности распознавания слитной речи. Тестовый материал предполагает оценку качества распознавания на русском и английском языках.

Лексическая часть для каждого языка состоит из квази-омонимов или так называемых «минимальных пар», а также других пар слов, способных вызвать трудности при распознавании. Материал фонетически сбалансирован, содержит основные реализации всех фонем обоих языков. Ср.:

выгореть-выгорать, вырвать-вырыть, довольно-довольный, заказ-закат, мол-мор, в охотку-в охотку, корпусной-корпусный и т.д.

Assert-asset, await-awake, back-bag, bald-balk, mad-maid, file-fire, journal-journey, sport-spot, you-your, etc.

Синтаксическая часть представляет собой адаптированный вариант материала, часто используемого для определения качества передачи речи, взятого из ГОСТа 16600-72 [4], и состоит из трёх групп для каждого языка, которые различаются по типу предложений:

простые, сложноподчинённые и сложносочинённые предложения. Материал также фонетически сбалансирован — в каждой группе фраз встречаются все фонемы русской речи и их основные варианты произношения. Слова для предложений взяты из разряда нейтральной лексики. Ср.:

*Фильм снимают целый год. Самолёт оказался в воздушной яме.
I cannot find candies. Phil, focus please.*

*Оператор стирал старые записи, как вдруг неожиданно раздался звонок.
Prove it before I give you that promise.*

*Диктора поразило это сообщение, но план уже утверждается в области.
I came to talk, but you took everything wrong.*

Кроме того, тестовый материал может быть адаптирован под готовые приложения с учётом сферы применения, типа приложения, объёма и специфики словаря и прочих факторов.

Зачитывая тестовый материал, участники тестирования проверяют результат распознавания и фиксируют ошибки (замены, вставки и выпадения). Затем производится подсчет ошибок, и вычисляются WER (Word Error Rate) [5] и WES (Word Error Rate per Sentence) [6] для слов и предложений соответственно по формулам:

$$WER = \frac{sub + del + ins}{nwords} \times 100\% ; WES = \frac{sub(s) + del(s) + ins(s)}{nwords(s)} \times 100\% ,$$

где *sub*, *del* и *ins* — это количество замен, выпадений и вставок соответственно, *nwords* — количество слов, (s) означает «в предложении». Вычисляется WER для каждой группы слов, WES — для каждого предложения, а затем среднее WES для каждой группы предложений. Также выявляются основные ошибки и трудности распознавания.

По разработанной методике нами были протестированы четыре системы автоматического распознавания речи. Для каждой ASR были вычислены WER и WES, проведено их сравнение, сделаны выводы о качестве распознавания каждой системы. Кроме того, были выявлены наиболее частые ошибки, возникающие при распознавании. Рис. 1 отражает долю ошибок каждой системы от общего числа тестовых единиц при распознавании самых «проблемных» звуков. На диаграмме четко видно, что худшее качество показала система ASR 2, тогда как ASR 3 допустила меньше всего ошибок при распознавании. Эти результаты подтверждаются также числовыми показателями WER и WES.

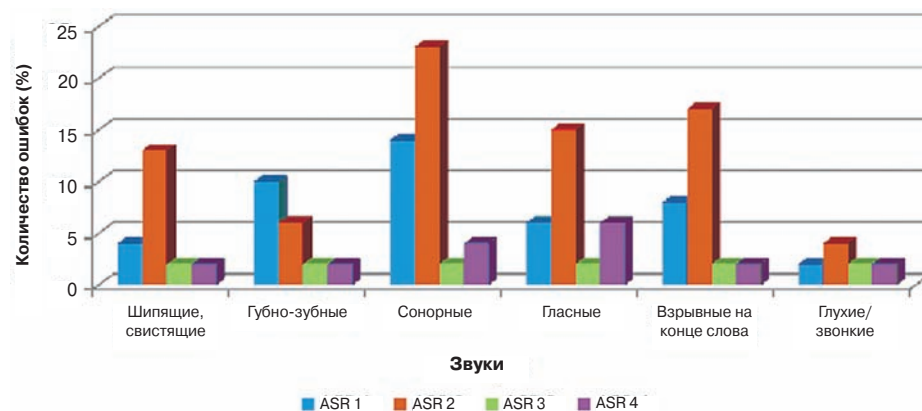


Рис. 1

Тестирование систем TTS

Тестовый материал для систем автоматического синтеза речи разделен на две части: лексическую и синтаксическую. В лексическую часть входят слова, обладающие определёнными характеристиками, способными повлиять на качество синтеза. С их помощью проверяется качество синтеза изолированных слов. Синтаксический блок включает в себя разного типа предложения для проверки качества синтеза слитной речи. Тестовый материал предполагает оценку качества распознавания на русском и английском языках. Оба блока содержат труднопроизносимые звукосочетания: сонорные звуки на стыке слов и слогов, гласные на стыке слов и слогов, шипящие и свистящие звуки и их сочетания, сочетания взрывных, взрывные на конце слова, и т.д. Выбор в пользу такого рода сочетаний звуков сделан на основе опыта работы с системами автоматического синтеза речи и обусловлен существованием проблем, с которыми в этой связи приходилось сталкиваться. Данная методика является комбинацией и адаптацией различных методов тестирования систем TTS, таких как DRT (Diagnostic Rhyme Test), MRT (Modified Rhyme Test), направленные на проверку качества синтеза начальных и/или конечных согласных; DMCT (Diagnostic Medial Consonant Test), проверяющий синтез интервокальных согласных; PB (Phonetically Balanced Word Lists), MOS (Mean Opinion Score), определяющие общее качество синтезированной речи, и т.д [7].

Первый блок тестового материала разделен на 6 групп:

- многосложные слова;
- сложные слова и аббревиатуры;
- слова с сочетанием нескольких гласных подряд;
- имена и фамилии;
- географические названия;
- числительные.

Поскольку данное тестирование проводится для оценки систем широкой сферы применения, конкретный языковой материал берется из числа общеупотребительной и деловой лексики. Материал фонетически сбалансирован и содержит звукосочетания, способные вызвать проблемы при синтезе. Ср.:

Группа №1.1: многосложные слова

Аббревиатура, деформировать, инфраструктура, сертификат, и т.д.

Группа №1.1 (eng): polysyllables

Abbreviation, character, government, handkerchief, significance, etc.

Группа №1.2: сложные слова и аббревиатуры

Работоспособность, ОАО, США, бухучёт, мосгорсуд, и т.д.

Группа №1.2 (eng): complex words and abbreviations

Ballot-paper, crisscross, SOS, PhD, etc, VIP, D.C, etc.

Группа №1.3: слова с сочетанием нескольких гласных подряд

Актуальный, закоулочек, зоопарк, киоск, материал, хаотичный и т.д.

Группа №1.3 (eng): words with several vowels in a row

Alien, alleviate, cooperate, curiosity, dubious, loathe, moreover, etc.

Группа №1.4: имена и фамилии

Пётр, Илья, Анастасия, Ксения, Коваленко, Кривич, Залевская, и т.д.

Группа №1.4 (eng): first and last names

Duncan, Bernard, Anna, Alice, Katherine, Smith, Cameron, Darwin, etc.

Группа №1.5: географические названия

Санкт-Петербург, Швейцария, Средняя Азия, Ближний Восток, и т.д.

Группа №1.5 (eng): geographical names

New York, Los Angeles, Manchester, Wales, Middle East, Asia, etc.

Группы №1.6 и №1.6 (eng): числительные / Numerals

1, 2, 3, 4, 5, 10, 11, 12, 13, 15, 20, 26, 100, 200, 1000.

Второй блок тестового материала состоит из предложений разного типа. Языковой материал данного блока является адаптированным материалом ГОСТа 16600-72: Передача речи по трактам радиотелефонной связи [4] и ГОСТа Р 50840-95: Передача речи по трактам связи. Методы оценки качества, разборчивости и узнаваемости [8]. Из предложенных в ГОСТах списков предложений были выбраны предложения, содержащие нейтральную лексику, и далее скомпонованы так, чтобы получились сложноподчинённые и сложносочинённые предложения. Следует отметить, что в получившихся предложениях встречаются все вышеперечисленные звукосочетания. Ср.:

Солнце ещё находится в зените. Фильм снимают целый год.

The room was remarkably groom and grouse.

Мальчик разбил санки о скамейку, а ведь они были заново выкрашены красной краской.

My telephone was too old, and I took a new one.

Этап был завершён, когда был собран киноаппарат.

Before you brought this book, my bookcase was not so heavy.

При тестировании систем синтеза речи с помощью первого блока участникам эксперимента предлагается оценить разборчивость и естественность синтезированной речи по пятибалльной шкале и, если была отмечена ошибка синтеза, указать, какая ошибка была допущена синтезатором, зафиксировав кириллицей или латиницей, используя специальные обозначения (таблица 1), вариант произношения, предложенный системой.

Таблица 1

Обозначение	Пример	Значение
Буква в верхнем регистре	парАметрический	Смещение ударения в слове
Замена буквы на 0	пар0метрический	Пропуск звука
Вставка звуков	пара0метрический	Посторонние звуки
Знак → перед словом (в предложении)	Мы поехали на → станцию	Ускорение темпа на слове
Знак ← перед словом	Мы поехали на ← станцию	Замедление темпа на слове
Знак ↑ перед/после/в середине слова	Мы поеха ↑ ли на станцию	Восходящая интонация
Знак ↓ перед/после/в середине слова	Мы поеха ↓ ли на станцию	Нисходящая интонация
Знак ~ перед/после слова	Мы поехали на ~ станцию	Прерывистая интонация
Знак ~ в середине слова	Мы поехали на ста~нцию.	Посторонние шумы/невнятность
Знак	Мы поехали на станцию	Лишняя пауза
Знак X	Мы X поехали на станцию	Отсутствие необходимой паузы

На втором этапе осуществляется проверка плавности воспроизведения предложений, расстановки фразовых ударений, а также просодические характеристики синтезируемой речи. Участники тестирования, помимо вышеописанных критериев, используют для оценки такие критерии как скорость, паузы и интонация. Интонация оценивается по двум шкалам: «верная-неверная», «ровная-прерывистая», расстановка пауз – по шкале «верно-неверно».

После этого для каждого блока производится подсчёт доли верно синтезированных слов/предложений; слов/предложений, воспроизведённых с минимальными искажениями; и слов/предложений, сильно искажённых при синтезе, от общего числа слов/предложений. Естественно, оценка является субъективной и опирается на восприятие синтезированных слов и предложений участниками тестирования. На завершающем этапе тестирования проводится сбор и анализ полученных результатов, после чего составляется сводная таблица, в которой указываются системы TTS, средний процент точного

воспроизведения, незначительно искажённого, сильно искажённого и основные ошибки синтеза как для слов, так и для предложений.

Все вышеперечисленное было проделано для четырех систем автоматического синтеза речи.

Основными типами ошибок оказались:

- смещение ударения (например, Анаст^Асия вместо Анастас^Ия);
- выпадение звука (например, Euro0 вместо Euroe);
- замена звука (например, ириспруденция вместо юриспруденция);
- неровная интонация (например, оператор стирает ↑ старые записи).

Процент этих ошибок от общего числа тестовых единиц для каждой системы TTS представлен на рис. 2.

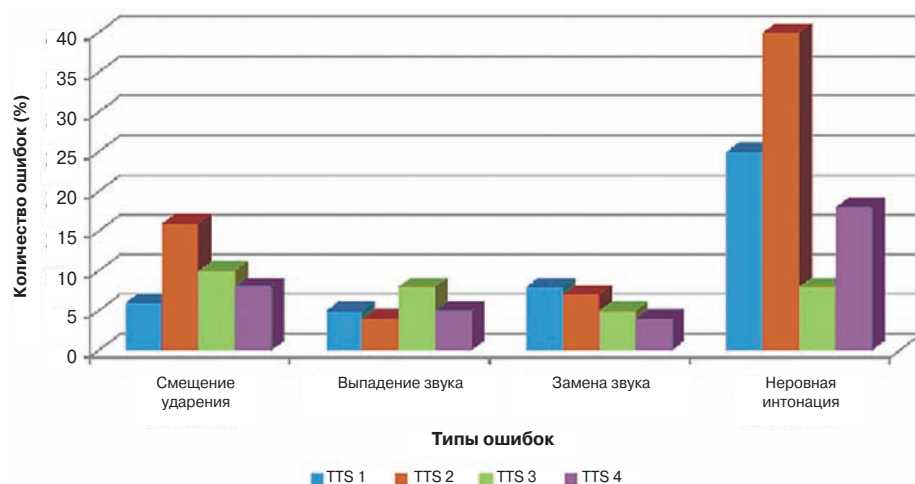


Рис. 2

Кроме того, было выявлено, что количество ошибок для разных тестовых групп различается.

Например, самыми сложными для синтеза оказались сложные слова и аббревиатуры, тогда как числа синтезировались сравнительно хорошо: у двух из четырех систем ошибок не зафиксировано, о чем свидетельствует рис. 3.

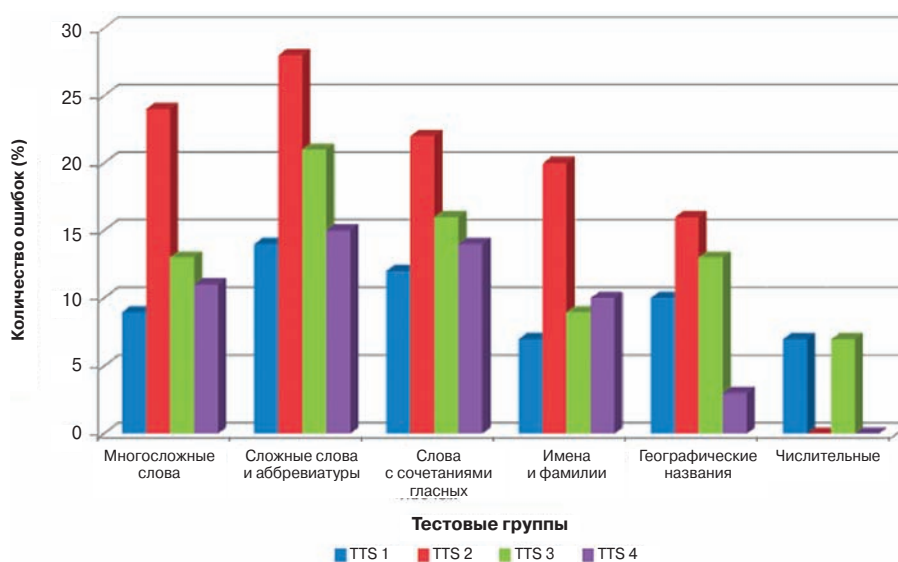


Рис. 3

Выводы

Разработанная методика тестирования систем автоматического синтеза и распознавания речи позволяет выяснить, какими недостатками и достоинствами обладают существующие системы ASR и TTS, сравнить их, а также определить коммерческую целесообразность их использования. Кроме того, после проведения такого тестирования можно сделать общие выводы об основных проблемах, связанных с автоматическим синтезом и распознаванием речи, и способах их преодоления. Важно отметить, что данная методика может иметь успешное коммерческое применение, так как она позволяет сделать выбор в пользу той или иной системы ASR и TTS для ее внедрения в конкретное решение. Результатом разработки данной методики стали зарегистрированные базы данных для тестирования систем ASR и TTS [9].

Список литературы

1. *Грамматчиков А.* Поговорить с компьютером. [Электронный ресурс]. Режим доступа: http://expert.ru/expert/2007/44/pogovorit_s_komputerom/
2. *Русанова О.А.* Исследование и разработка методов анализа и оценки качества синтезированной устной речи: дисс... канд. техн. наук / Краснояр. гос. техн. ун-т. Красноярск. 2004. 107 с.
3. *Нгуен Минь Туан.* Разработка алгоритмов построения оценок достоверности для систем распознавания речи: дисс... канд. техн. наук / Вычисл. центр РАН. М.: 2008. 102 с.
4. ГОСТ. 16600-72. Передача речи по трактам радиотелефонной связи. Требования к разборчивости речи и методы артикуляционных измерений. Переизд. 1973. Взамен ГОСТ 16600-71; Введ. 27.09.72. М.: Госстандарт России. 1973. 75 с.
5. Word Error Rate. [Электронный ресурс]. Режим доступа: http://en.wikipedia.org/wiki/Word_error_rate
6. *Strik H., Cucchiaroni C., Kessens J.M.* Comparing the recognition performance of CSRs: in search of an adequate metric and statistical significance test, Proc. ICSLP-2000, Beijing, 2000. P. 740–743
7. *Lemmetty S.* Review of Speech Synthesis Technology, Master's Thesis, Helsinki University of Technology. 1999. 104 p.
8. ГОСТ. Р 50840-95. Передача речи по трактам связи. Методы оценки качества, разборчивости и узнаваемости. Введ. 01.01.97. М.: Госстандарт России. 1996. 230 с.
9. Свидетельство о государственной регистрации базы данных № 2012620510. «База данных материала для тестирования систем синтеза и распознавания речи для оценки качества». Правообладатель: ЗАО «ЭсТюЭс Нэкст». Зарегистрировано в Реестре баз данных 5 июня 2012 года.

Сведения об авторах

Корсакова Н.С. —

лингвист, ЗАО «ЭсТюЭс Нэкст». info@s2snext.com

Засыпкина К.А. —

директор по проектам, ЗАО «ЭсТюЭс Нэкст». info@s2snext.com