

Сергей Соболенко

**Искусственный интеллект. Строки,
контекст и волны на Паскале**

«Издательские решения»

Соболенко С.

Искусственный интеллект. Строки, контекст и волны на Паскале /
С. Соболенко — «Издательские решения»,

ISBN 978-5-44-963281-4

Эта книга подарит путешествие в увлекательный мир строковых данных, открывающий новые возможности для исследования искусственного интеллекта. В ней вы найдете множество примеров и иллюстраций, а кроме того, идей, до этого момента неопубликованных. Она будет полезна и интересна как новичку в теме программирования, так и искушенному читателю, интересующемуся наукой и технологиями.

ISBN 978-5-44-963281-4

© Соболенко С.
© Издательские решения

Содержание

Предисловие, которое рекомендуется прочитать	6
Глава 1. Строковые процедуры и функции на практике	15
Подготовка к работе в TMT-Pascal	15
Строковые данные	16
Длинные строки	17
Конец ознакомительного фрагмента.	18

Искусственный интеллект. Строки, контекст и волны на Паскале

Сергей Соболенко

«Обучение путем подачи хорошего примера часто – самый эффективный, а иногда – и единственно возможный метод.»

Никлаус Вирт, создатель языка Паскаль.

«Меня не интересует создание мощного компьютерного интеллекта. Мне вполне хватило бы самого посредственного – примерно как у президента Американской телефонной и телеграфной компании.»

Приписывается Алану Тьюрингу

© Сергей Соболенко, 2019

ISBN 978-5-4496-3281-4

Создано в интеллектуальной издательской системе Ridero

Предисловие, которое рекомендуется прочитать

Учебные заведения России и мира продолжают обучать программированию на Паскале. И это происходит не из-за лени преподавателей к переизданию методических пособий по информатике. Такое обучение позволяет понять гибкую внутреннюю логику организации программ, ведь Паскаль обладает свойствами универсальности и широкими возможностями, при своей компактности и легкости понимания. С помощью этого языка программирования можно адаптировать, подготовить восприятие школьника или студента для самоорганизации и в других языковых средах.

Поэтому в этой книге автор изложил полезные и практические аспекты накопленного многолетнего опыта, приемы и методы работы с неформализованными строковыми данными на Паскале; а также сведения о новой технологии, являющейся плодом этих практических изысканий. Модель описанной технологии искусственного интеллекта, в качестве исследовательского проекта также реализуется на этом языке. При этом, значительная часть приведенных алгоритмов и программных процедур *публикуется впервые*, что придает изданию особую ценность.

Почему эта книга актуальна?

Вся беда в том, что профессиональным приемам подробной, прикладной работы с данными строкового типа занимается очень ограниченное число специалистов. Отсюда вытекает явный недостаток практических пособий по данной тематике,¹ и как следствие – неявный прикладной функционал в современных языковых компиляторах.² Также этому есть и другое вполне логичное объяснение. Тема парсинга строковых данных много десятилетий являлась своего рода табу, дурным тоном программирования. Традиционно устоялся стереотип, что строковые переменные – это «рабы интерфейса» и баз данных, а неформализованные сведения – зло, с которым нужно бороться. Результат – налицо: отсутствие приличной школы: как учителей, разработчиков, так и их учеников. Отсутствие предложения за много десятилетий сформировало зеркальное отсутствие спроса. А затем появились и искусственные нейронные сети, которые худо-бедно начали с этими данными работать; но вот разработчикам от этого не стало легче. По существу, искусственные нейросети «подарили» программистам «черный ящик», действия которого бывают необъяснимы.³

Этот пробел явно требует компенсации; тем более, очевидно неустраиваемой в современной парадигме когнитивных систем оказалась ниша *символьного искусственного интеллекта*.⁴

¹ В данном случае рассматривались десятки русскоязычных учебных пособий, самоучителей, методических рекомендаций и задачиков по Паскалю, C++, Java и PHP, изданных за последние 20—25 лет.

² Хотя современные наборы процедур и функций для работы со строками, например в PHP, Object Pascal и C++ довольно похвальны, (а в Python, – интуитивно удобны и компактны), но на практике очень редки сколько-нибудь интересные алгоритмы, использующие их где-то, кроме баз данных и текстовых процессоров. Мастодонты странного программирования, как Lisp и Prolog автором не рассматривались принципиально, – они напугали его еще в ранней юности.

³ Разработчик системы на основе ИНС далеко не всегда может отвечать за определенные решения программы и ему также сложно утверждать, не является ли определенный ответ эффектом «переобучения», иначе говоря, случайным, «мусорным» ответом.

⁴ **Символический** (символьный) **искусственный интеллект** – это собирательное название для всех методов исследования искусственного интеллекта, основанных на высокоуровневом «символическом» (человекочитаемом) представлении задач, логики и поиска. Символический ИИ лёг в основу доминирующей парадигмы исследований ИИ с середины 1950-х до конца 1980-х. Наиболее успешная форма символического ИИ – это экспертные системы, использующие сеть продукционных правил. Продукционные правила объединяют символы в отношения, похожие на оператор «если-то». Экспертная система, обрабатывая эти правила, делает логические выводы и определяет, какая дополнительная информация ей необходима, то есть какие следует задать вопросы, используя человекочитаемые символы. *Определение Wiki*.

В книге приводятся необычные, но достаточно простые решения по проекту интеллектуальной символьной диалоговой системы. Тем не менее, описанные подходы к ее реализации категорически отличаются от классического представления об объектно-ориентированном программировании, традиционных символьных экспертных систем и от современной методологии машинного обучения. Также весьма существенным моментом в этой работе является отказ от прямого использования символьных конструкций на естественном языке. Тем не менее, практические результаты показывают, что этот комплекс методик может решать задачи, по сей день считавшиеся серьезной проблемой.

Чем мы занимаемся?

«Мы» в данном случае – это небольшое сообщество единомышленников, не входящих в какое-либо формальное объединение, но захваченных идеей реализации искусственного интеллекта на базе технологии MSM.

Сверхзадачей исследований в данном направлении не являлось и не является создание *сильного искусственного интеллекта*⁵ в понимании современных физиков и инженеров; (такого, который будет многократно превосходить умственные способности людей). Для начала, следовало бы создать «средний» интеллект. Такой, каким обладает например, ваш сосед, Михаил Ефремович, учитель географии средних лет. Особо ничем не выдающийся, иногда не очень внимательный или рассеянный, иногда циничный, иногда смешной, но... понимающий. Способный сравнивать, оценивать, делать выводы, формулировать суждения. Решать повседневные задачи. Планировать, следовать определенным целям. И главное, способный действовать так, как этого требуют обстоятельства. Вы скажете, такие машины уже есть? Хорошо, если вы так считаете, не будем спорить. Но, автор считает, что нет.

Это учебник?

Книгу можно использовать как учебное пособие по разделу строковых данных на Паскале. Но было бы неверно воспринимать показанные решения, как нечто неоспоримое, – любая приведенная процедура может быть усовершенствована. Сама природа строк приближает нас к другой форме подхода к методикам реализации. Здесь, в этой новой реальности открывается то, что называется нечеткостью, субъективной оценкой и многогранностью связей элементов конструктора, который мы можем выстраивать.

Книга затронет и некоторые философские, психологические, лингвистические аспекты сущности человеческого восприятия. Но она не станет тешить человеческое самолюбие или рисовать утопические миры. На этом поле и без того работают слишком много мастеров слова, чтобы стать только очередным из них. Автору это неинтересно.

Гораздо интереснее работа, имеющая под собой практическое основание, а не метафорические рассуждения. «Практика есть критерий истины» – известное выражение, приписываемое Карлу Марксу, но авторство в данном случае не имеет значения.

Эта книга пробует осветить шаг за шагом движение в новом, малоизученном направлении работы с новой технологией. Страница за страницей практического освоения вы имеете шанс испытать чувства первооткрывателя, какие вероятно испытывал Левенгук, настраивающий первый оптический микроскоп. И пусть спустя сотни лет был построен и микроскоп элек-

⁵ **Сильный искусственный интеллект** – гипотеза в философии искусственного интеллекта, согласно которой некоторые формы искусственного интеллекта могут действительно обосновывать и решать проблемы. Теория сильного искусственного интеллекта предполагает, что компьютеры могут приобрести способность мыслить и осознавать себя как отдельную личность (в частности, понимать собственные мысли), хотя и не обязательно, что их мыслительный процесс будет подобен человеческому. *Определение Wiki.*

тронный, но микроскоп конструкции Левенгука был самым важным, первым шагом в нужном направлении.

С другой стороны, кто-то мог бы и сказать «*Да обернитесь вокруг: в нашем мире data science,⁶ блокчейна⁷ и искусственных нейросетей нет места для того, чтобы заниматься какими-то символами...*» Но вещь, кажущаяся сегодня слишком простой или неочевидной может стать завтра тем самым электронным микроскопом. История знает массу таких примеров.

Например, концепция первого *перцептрона*,⁸ на основании которой родилось все изобилие современных искусственных нейросетей и машинного обучения более чем на 30 лет была подвергнута остракизму во время так называемой «Зимы искусственного интеллекта». И что бы сказали бывшие критики перцептрона теперь, когда каждая домохозяйка, школьник или пенсионер, имея некоторый интерес и немного свободного времени может настроить искусственную нейросеть и сортировать огурцы по 12 критериям в собственном гараже.⁹

Можно освоить вещи и поинтереснее огурцов, например машинный перевод, зрение сквозь стены или толщи океанских вод... Но автор в данном случае равнодушен по отношению к искусственным нейросетям, к биг-дате или к попыткам создавать коннектоны. Безусловно, это в своем роде интересные, нужные и незаменимые вещи, представляющие немаловажный научный и коммерческий интерес. Плохая новость состоит в том, что все это имеет мало общего с настоящим интеллектом, который основан на процессе понимания смысла. Да, искусственные нейросети хорошо работают. Выписывают штрафы. Рисуют картины. Консультируют по ряду известных проблем. Играют в шахматы, Го или водят машины. Они это все умеют или будут уметь; и ваша сетка будет уметь гораздо лучше сортировать огурцы, чем человек, – с помощью нескольких строк кода, небольшой настройки переменных и нескольких тысяч примеров выборки. Это – сложная, но популярная и вполне осуществимая задача в рамках существующей парадигмы машинного обучения. Кроме того – уже изданы сотни прикладных учебных пособий по этой теме, существуют простые видеоинструкции по работе с кодом на Питоне и в свободном доступе находятся нужные приложения и наборы данных.

То, о чем говорит эта книга – трудно. Требуется умственного труда, фиксации внимания, сосредоточенности. Эта книга – для тех, кто привык использовать критическое мышление, ставить все под сомнение, моделировать собственную реальность, а не идти вперед за общественным мнением.

Кого-то эта книга может развлечь, кого-то заставит задуматься или по-новому взглянуть на информационные технологии. Возможно – кто-то сделает интересный учебный проект, доклад, презентацию или реферат, за который может получить хорошую оценку. А кто-то, (и на это хочется надеяться), использует описанные методы, чтобы участвовать в приближении лучшего будущего с машинами, имеющими сознание. При этом, не будет иметь значения язык реализации, – все описанные методики могут быть переведены на любой язык програм-

⁶ «**Наука о данных**» (англ.) – раздел информатики, изучающий методы статистической обработки больших объемов данных в приложениях искусственного интеллекта и проблемы представления данных в цифровом формате для машинного обучения.

⁷ **Blockchain** (англ.) – технология распределенных баз данных для хранения информации на различных компьютерах в сети; преимущественно используется для хранения и передачи информации о финансовых транзакциях. Используется с 2008 года для работы с криптовалютами.

⁸ **Перцептрон**, или персептрон (англ. perceptron от лат. perceptio – восприятие; нем. Perzeptron) – кибернетическая модель мозга, предложенная Фрэнком Розенблаттом в 1957 году и впервые реализованная в виде электронной машины «Марк-1» в 1960 году. Перцептрон стал одной из первых моделей нейросетей, а «Марк-1» – первым в мире нейрокомпьютером.

⁹ *Пример с огурцами – не выдумка. Японец Макото Коике в 2016 году создал систему, которая с помощью технологии машинного обучения TensorFlow от Google сортирует огурцы на ферме своих пожилых родителей. Для этого Макото использовал одноплатный компьютер Raspberry Pi 3, оснащенный камерой. Непосредственной сортировкой в его системе был занят микроконтроллер Arduino, получающий команды после обработки в сенсорной сети TensorFlow и отправляющий огурцы в разные ведра.*

мирования для достижения лучших показателей в скорости, адаптивности или коммерческой ценности.

Что здесь интересного?

Во-первых, эта книга является практическим руководством по работе со строковыми данными и может выполнять роль «решебника» для разного типа задач; во-вторых, она рассказывает о новой технологии, (в которую автор оказался замешан практически поневоле); и в-третьих – она затрагивает различные аспекты и неординарные подходы к проблеме *сильного искусственного интеллекта*.

При этом автор не станет говорить о чем-то невозможном. Любые части повествования будут относиться только к тем вещам, которые действительно существуют или к тем, которые могут быть созданы.

Вначале мы поговорим о представлении строковых данных на Паскале и о том, какие необычные процедуры и функции можно использовать, чтобы решать неординарные задачи, из разряда олимпиадных задач по информатике. Вероятно, эта часть особенно понравится школьникам и студентам, ведь здесь будет описано множество подпрограмм, имеющих прикладное применение.

Также мы вспомним о том, что такое альтернативная классификация данных в многомерном строчном множестве и каковы могут быть приемы работы с ними.

Далее, мы сосредоточимся на том, чтобы немного разобраться в действующих диалоговых интеллектуальных системах, постараемся понять их сильные и слабые места, разберем их «по винтикам» и порассуждаем, «как это работает?»

Во второй половине книги будут приведены новые методы работы с полисемией и контекстом, в чем нам поможет волна предложений, адаптированный толковый словарь и еще немного мозгового усердия.

И наконец, в завершении мы постараемся сконструировать систему, (пока только в теории, но со значительной степенью детализации) претендующую на звание «сильной логической машины». В частности, здесь мы поговорим о гештальт-цикличности, теориях физики мышления и связи логики, языка и интеллекта.

Кстати, в книге вы не увидите отсылки в приложения (чего так не любят многие читатели): все процедуры и функции описываются в разделе соответствующей тематики прямо по ходу повествования с приведением иллюстраций, показывающих, что происходит на экране компьютера.

О чем речь?

Это повествование является не столько логическим продолжением первой книги автора, сколько изложением «обратной стороны» практического программирования, комплекса приемов, демонстрирующих структуру программы обработки строковых данных «с нуля».

Книга «Искусственный интеллект: начала MSM», вышла в начале 2018 года.¹⁰ Она рассказывает об основах перспективной технологии в построении искусственного интеллекта – многомерном строчном (строковом) множестве (или «*Multidimensional Strings Multiplicity*», англ.). Простая программная среда, построенная на основе специфического классификатора всего за год с небольшим своего существования показала замечательные результаты, в связи с чем автор посчитал важным продолжить публикацию данных, а также методов и технологий, реализуемых в этом проекте.

¹⁰ Соболенко С. В. «Искусственный интеллект: начала MSM: Сингулярность неизбежна: Издательские решения, 2018. – 272 с.

В новой книге автор расскажет и о технологических приемах, использующихся в программной модели для решения наиболее сложных проблем «понимания» естественного языка. В первую очередь это касается вопросов полисемии и контекста, которые чаще всего, (а может быть и по привычке), считаются в ЭВМ непреодолимыми.

Что потребуется?

Все приведенные в качестве примера процедуры и функции реализованы на языке Pascal, а точнее, в бесплатном компиляторе TMT Pascal lite.¹¹ Выбор автора оказался на стороне этого компилятора отнюдь не случаен. Это один из немногих компиляторов Паскаля, позволяющих использовать во внутренних процедурах и функциях входные и выходные массивы переменных строчного, логического и целочисленного типа.¹² Кроме того, TMT Pascal известен, доступен и понятен настолько, что строить простые программные модели, обрабатывающие строковые массивы данных не доставит большого труда.

Споры об MSM

Технология MSM – вещь своенравная. Первые публикации о ней в сети в начале 2018 года были восприняты с подозрением, а иногда и с недоумением. Под сомнение ставилась как сама технология, так и квалификация автора, мол «Куда ты, блогер, сунулся в калашный ряд?»¹³ Но были среди критических замечаний и вполне корректные, достойные рассмотрения.

Особенно часто оппонентам автора вспоминался Курт Фридрих Гедель,¹⁴ австрийский математик и философ с его теорией о неполноте, а также интерпретация этой теории физиком и математиком Роджером Пенроузом¹⁵ в контексте принципиального различия между человеческим мозгом и компьютером. Мы не будем сильно беспокоить Геделя, поскольку систему MSM, едва ли можно назвать математически формальной.

Основой MSM является иерархическая онтология, которая строится в соответствии с интуитивной, смысловой, неформализованной «нагрузкой» понятий, их синонимов, весов, отношений и правил. Иерархия имеет комплементарный код каждого элемента, не имеющий фиксированной числовой разрядности, который построен по принципу бесконечности и многомерности. В словарях, организованных по принципу MSM программная модель оперирует не конкретными словами, а их смысловым кодом, и действует на основе подбора правил, выбранных в соответствии с конкретными ситуациями. Также в логике MSM реализован механизм, который позволяет уйти от математической, компьютерной точности в сторону нечеткости, адаптивности к конкретным условиям. Этот механизм помогает определять близкие понятия, находить родство и различия слов-множеств на основании онтологических связей, фактически вшитых в код понятий, в противовес четкому математическому сходству формального написания слов на естественном языке.

В книге «Искусственный интеллект: начала MSM» приводился простой пример, когда эталонное знание **«Кошачьи имеют красивую шерсть»** автоматически порождает новые знания о том, что **«У киски есть красивая шерстка»** или **«У льва есть прекрасная**

¹¹ MS-DOS Edition Version 3.90 (Copyright © 1995, 2002 by TMT Development Corporation).

¹² Для сравнения, популярный в России ABC Pascal – не позволяет производить данные операции; а поздние версии Delphi и Object Pascal используют более сложный операционный инструментарий, чтобы сосредоточиться на нем на страницах небольшой по объему книги.

¹³ Однако, что более интересно, – реакция зарубежного форума OpenAI, где демонстрация проекта произвела сильное впечатление, вызвала интерес и получила явное одобрение. Прим. автора.

¹⁴ Курт Фридрих Гедель (нем. Kurt Friedrich Gödel; 1906—1978) – австрийский логик, математик и философ математики. Наиболее известен сформулированными и доказанными им теоремами о неполноте, которые оказали огромное влияние на представление об основаниях математики. Считается одним из наиболее выдающихся мыслителей XX века.

¹⁵ Сэр Роджер Пенроуз (англ. Roger Penrose, род. 8.08.1931 года, г. Колчестер, Англия) – английский физик и математик, работающий в различных областях математики, общей теории относительности и квантовой теории.

грива», и еще сотен подобных. Этот процесс совсем не похож на математичный. Первоначально введенные данные, формально, математически, совершенно отличаются от проверочных, и тем не менее программа находит их идентичными, поскольку связи объектов, вшитые в коде, и правила обработки позволяют это сделать.

Конечно же, подобный результат можно было бы получить и посредством обычного объектно-ориентированного программирования (ООП),¹⁶ однако относительно энергозатрат в данном случае MSM оказывается в безусловном выигрыше за счет комплементарности кода понятий и общей простоты организации системы. На следующем примере также видно принципиальное отличие в построении классификатора традиционной семантической сети (рис. 1) и MSM (рис. 2).

В отличие от традиционного решения семантической сети и вариантов объектного программирования, в которых сеть связей элементов описывается с помощью дополнительных описательных блоков, не обладающих комплементарными зависимостями, элементы в MSM входят в единую структуру, где каждый из них обладает «памятью предков», а кроме того «отягощен» свойствами и «отношениями». Эта структура обладает безграничной глубиной связей, «защитой» в самом коде элементов. На рисунке – показан лишь микроскопический фрагмент, примерно 1/10 000 доли реального объема структуры, которая кодируется с помощью этих простых правил.

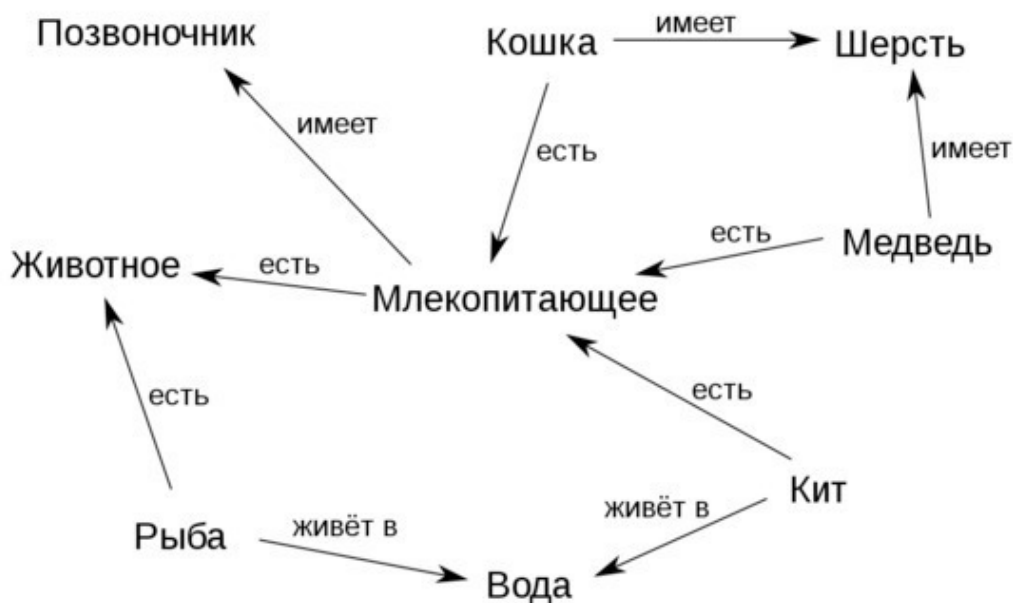


Рис. 1. Элементы семантической сети представлены в виде ориентированного графа, вершины которого обозначают объекты предметной области, а ребра между ними формируют отношения между ними.

¹⁶ **Объектно-ориентированное программирование** (ООП) – методология программирования, основанная на представлении программы в виде совокупности объектов, каждый из которых является экземпляром определенного класса, а классы образуют иерархию наследования. *Определение Wiki.*

Кроме того, во взаимоотношениях элементов MSM играют немаловажную роль дополнительные словари зависимостей, весов и свойств, позволяющие с помощью минимальных средств строить сложные логические конструкции, обобщающиеся на целые классы значений.

Приведенный фрагмент классификатора занимает минимальный объем памяти, составляющий менее (!) 1 килобайта. За счет подобной реализации в перспективе можно достигнуть возможностей мощной логической машины, имеющей ничтожный объем в десятки или сотни мегабайт. Так приложение на основе MSM сможет работать даже в обычной бытовой технике вроде холодильника или микроволновки.

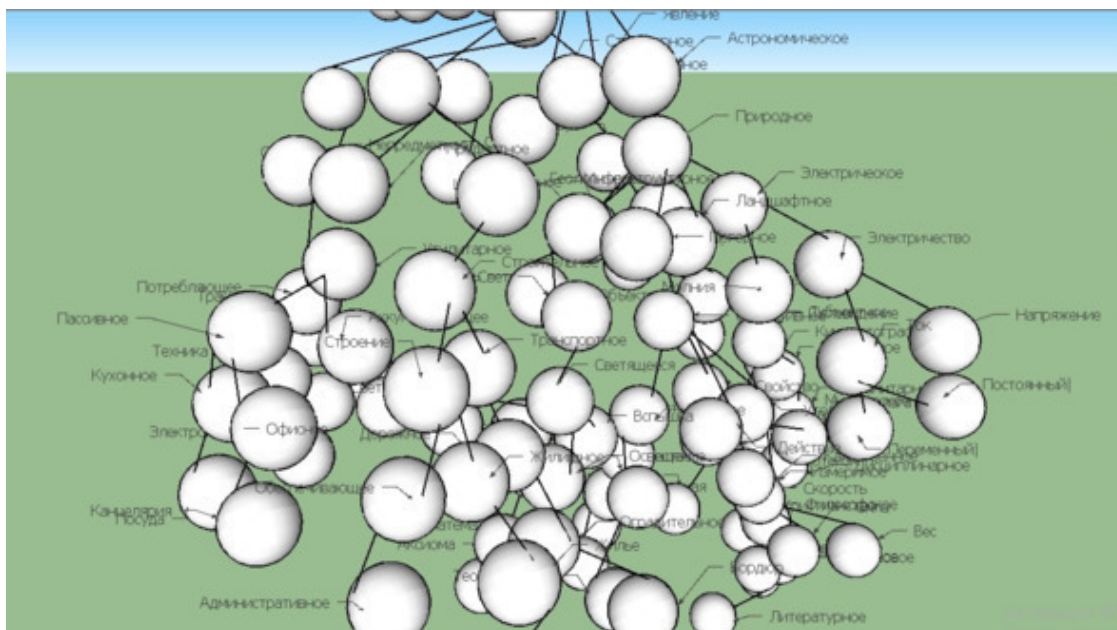


Рис. 2. Фрагмент визуализации дерева элементов MSM в 3D.

(Правда, насколько для потребителя может быть актуальна «понимающая» смысл языка микроволновая печь – это большой вопрос, но почему бы и нет?)

Но возвращаясь к теории Геделя, а точнее, к ее интерпретации, сам Роджер Пенроуз отчасти соглашается с вероятным сценарием построения сильного искусственного интеллекта: «Наше непонимание человеческого сознания говорит о наличие большого пробела в научных знаниях и этот пробел, возможно, находится там, где теории нужно выйти за пределы вычислительной системы».¹⁷

Вот еще цитата из его книги «Тени разума»:¹⁸

«В современной научной картине мира отсутствует один очень важный ингредиент. Этот недостающий ингредиент совершенно необходим, если мы намерены хоть сколько-нибудь успешно уместить центральные проблемы мыслительных процессов человека в рамки логически последовательного научного мировоззрения. Я утверждаю, что сам по себе этот ингредиент не находится за пределами, доступными науке, хотя в данном случае нам, несомненно, придется в некоторой степени расширить наш научный кругозор...»

Не является ли MSM тем самым недостающим звеном, пробелом в науке, выходящим за рамки математических вычислений? Ведь зависимости, образующиеся в этой технологии в большей степени функциональны только по причине организованных «вручную» связей, основанных на смысловой «нагрузке» языка, а также их отношений? Нематематические

¹⁷ «Роджер Пенроуз: Квантовая природа сознания», видео канала «2045 Initiative», 2013 г., на 3 мин. 19 сек.

¹⁸ Роджер Пенроуз «Тени разума – в поисках науки о сознании», М. 2005. с. 27—28.

«вычисления» следуют контексту естественного языка, который выступает носителем смыслов, весов, свойств и отношений между элементами; и система MSM в достаточной степени удовлетворяет полноте отражения этих отношений.

Возвращаясь к рассуждениям в книге «Тени разума», можно сделать вывод, что Пенроуз в своем видении вычислительной модели опирается на математизм подхода к рассмотрению идеи воплощения искусственного интеллекта. Суть же метода MSM состоит в *парсинге* сходства и различия объектов и использовании правил обработки не только относительно конкретных элементов, но и в отношении классов объектов и фраз, предложений, составленных из классов моделей. Похожие процессы происходят и в искусственных нейросетях, которые также основаны на принципе поиска сходства и различий, однако диалоговые нейросети используют для своих операций слова, фразы и предложения естественного языка, тогда как MSM оперирует их смысловым кодом, заимствованным из понимания естественного языка.

Но другой вопрос состоит в том, является ли сам человек эталоном смысла и интеллекта? И возможно ли, анализируя смысловую нагрузку языка, достигнуть более высоких интеллектуальных результатов, чем у среднестатистического человека, как это видится многим нашим современникам?

Еще одна цитата Пенроуза в этом месте кажется весьма уместной: *«Мы уверены, что являем собой апофеоз интеллекта в царстве животных, однако этот интеллект, по всей видимости, оказывается самым жалким образом не способен справиться с множеством проблем, которые продолжает ставить перед нами наше собственное общество»*.¹⁹

Таким образом Пенроуз понимает, что человеческий интеллект несовершенен. И как вы помните, наша задача в конструировании MSM видится скорее в воссоздании модели человеческой логики в программной модели, нежели в создании сверхинтеллекта. Объективное человеческое несовершенство является частью человеческой сущности; в связи с чем рано или поздно нам не избежать его воплощения, частичного или полного в электронно-вычислительной машине.

Почему этот вывод становится нам очевидным? Напомню, что посыл реализации диалоговой модели на основе MSM состоит в том, что человеческое мышление и символичный язык по существу неотъемлемы, как утверждает, в частности, Ноам Хомски.²⁰ И хотя этот тезис неоднозначен, но он хорошо аргументирован, ведь мышление без языка – это набор рефлексивных зависимостей, бихевиористический подход, способный решать лишь узкий набор задач, отвечающих за реакцию на положительное подкрепление. Находить выход в лабиринте, избежать столкновения, выиграть в шахматы или Го, – это все замечательные примеры реализации статистических, а иначе говоря, математических методов в построении искусственного интеллекта. И даже те современные системы, которые направлены на работу с естественным языком (например, переводчики или боты-консультанты) по существу не оперируют *смыслом* языка, а производят лишь внешнюю, автоматную, формальную обработку статистических зависимостей того или иного написания слов. Однако они не «понимают» принципиальной разницы между кукурузой и кузнечиком, не могут сказать, кто быстрее или умнее, не различают позитивных и отрицательных факторов; а ведь эти вопросы по существу является сутью мышления.

В пользу этой точки зрения стоит привести несколько фрагментов из статьи *«Ноам Хомский: где искусственный интеллект пошел не туда?»*:²¹

¹⁹ См. там же.

²⁰ **Аврам Ноам (Наум) Хомски** (также часто транскрибируется как Хомский или Чомски, англ. Avram Noam Chomsky; род. 7 декабря 1928, Филадельфия, штат Пенсильвания, США) – американский лингвист, политический публицист, философ и теоретик. Профессор лингвистики Массачусетского технологического института. Также он известен своей открытой позицией относительно связи мышления, языка и интеллекта.

²¹ Автор статьи – молодой ученый-когнитивист Ярден Кац, Ссылка на публикацию: <https://habr.com/ru/post/432846/>

«... Ноам Хомский и его коллеги работали над тем, что впоследствии стало называться когнитивной наукой – открытие ментальных представлений и правил, которые лежат в основе наших познавательных и умственных способностей. Хомский и его коллеги опрокинули доминирующую в тот момент парадигму бихевиоризма, возглавляемую гарвардским психологом Б. Ф. Скиннером, в которой поведение животных было сведено к простому набору ассоциаций между действием и его следствием в виде поощрения или наказания.

...В мае 2011 года, в честь 150-летней годовщины Массачусетского технологического института, состоялся симпозиум «Brains, Minds and Machines» («Мозги, умы и машины»), на котором ведущие ученые-информатики, психологи и специалисты в области нейронаук собрались для обсуждения прошлого и будущего искусственного интеллекта и его связь с нейронауками. Подразумевалось, что собрание вдохновит всех междисциплинарным энтузиазмом по поводу возрождения того научного вопроса, из которого и выросла вся сфера искусственного интеллекта: Как работает разум? Как наш мозг создал наши когнитивные способности, и можно ли это когда-либо воплотить в машине?

Ноам Хомский, выступая на симпозиуме, не был преисполнен энтузиазма. Хомский раскритиковал сферу ИИ за принятие подхода, похожего на бихевиоризм, только в более современной, вычислительно- сложной форме. Хомский заявил, что опора на статистические техники для поиска закономерностей в больших объемах данных маловероятно даст нам объяснительные догадки, которых мы ждем от науки. Для Хомского новый ИИ – сфокусированный на использовании техник статистического обучения для лучшей обработки данных и выработки предсказаний на их основе – вряд ли даст нам общие выводы о природе разумных существ или о том, как устроено мышление. Эта критика вызвала подробный ответ Хомскому со стороны директора по исследованиям корпорации Google и известного исследователя в области ИИ, Питера Норвига, который защищал использование статистических моделей и спорил о том, что новые методы ИИ и само определение прогресса не так уж далеки от того, что происходит и в других науках. Хомский ответил, что статистический подход может иметь практическую ценность, например, для полезной поисковой системы, и он возможен при наличии быстрых компьютеров, способных обрабатывать большие объемы данных. Но с научной точки зрения, считает Хомский, данный подход неадекватен, или, говоря более жестко, поверхностен.

«Мы не научили компьютер понимать, что означает фраза „физик сэр Исаак Ньютон“, даже если мы можем построить поисковую систему, которая возвращает правдоподобные результаты пользователям, вводящим туда эту фразу...»

Еще раз уточним: задачей-минимумом нашего сегодняшнего исследования остается реализация диалоговой системы, близкой к человеческой модели мышления. И есть надежда, что технология MSM является принципиально новым подходом, способным дать нам новую пищу, иной способ исследования; вдохнуть новую жизнь в парадигму искусственного интеллекта как символической системы, (но – не системы на естественном символическом языке).²²

²² Это очень важная деталь, ведь письменный естественный язык по существу является лишь системой письменного оформления звуков, а не классификатором смыслов. (Прим. авт).

Глава 1. Строковые процедуры и функции на практике

Подготовка к работе в TMT-Pascal

Все, что нам первоначально потребуется для работы – скачать и установить программу компилятора TMT-Pascal.

Это можно сделать бесплатно, например, отсюда: <http://pascal.sources.ru/tmt/download.htm>.

Здесь можно выбрать, например, версию [TMT Pascal Lite v.3.90](#), которая подходит для реализации наших задач.

Далее, для работы с русским языком в DOS-приложениях, нам понадобится произвести некоторые предварительные настройки. **На 32-битной Windows:**

В Windows XP:

В командной строке меню «Пуск» набираем команду «Regedit». Далее, вносим изменения: HKEY_LOCAL_MACHINE -> SYSTEM -> CurrentControlSet -> Control -> KeyboardLayout

Здесь нужно изменить значение ключа 00000409 в папке DosKeybCodes на «ru».

Также, в файле autoexec. nt, расположенном в Windows\system32, надо добавить строку «lh %SystemRoot%\system32\kb16.com ru». После перезагрузки переключение раскладки клавиатуры в DOS-приложениях будет производиться комбинациями Ctrl+Left Shift (английская) и Ctrl+Right Shift (русская). Упомянутый метод нормально работает в том случае, если Language settings for the system в качестве Default стоит Cyrillic.

В Windows 7:

В файле Windows\system32\autoexec. nt вписать строки:

```
«lh %SystemRoot%\system32\dosx
lh %SystemRoot%\system32\kb16.com ru
set clipper=F80»
```

В файле Windows\system32\config. nt вписать:

```
«files=80; buffers=99».
```

В командной строке в меню «Пуск» через команду «regedit» редактируем реестр:

HKEY_LOCAL_MACHINE -> SYSTEM -> CurrentControlSet -> Control -> Keyboard Layout

DosKeybCodes изменить значение параметра 0000409 на ru

Раскладка будет переключаться сочетаниями клавиш:

русский – правые Shift+Alt; английский – левые Shift+Alt.

В Windows 10:

Не требует изменений в реестре. В файле Windows\system32\autoexec. nt добавляем строчку – "%SystemRoot%\system32\kb16 ru».

В случае если используется **64-разрядный Windows**, требуется использование приложения-эмулятора для DOS-программ «DOS-box».

Строковые данные

Работе со строковыми данными всегда уделяют слишком мало внимания. Считается, что это нечто, само-собой разумеющееся и слишком понятное, чтобы делать на них акцент. В учебниках, самоучителях любой толщины и направленности нам сильно повезет, если им уделяется хотя бы 2 страницы.

Попробуем же восполнить известные пробелы, для чего начнем с самых азов работы со строками на Паскале.

Итак, строчные переменные и константы в Паскале описываются так же, как и любые другие, например строка **S: String** означает что переменная S будет хранить некий набор символов, строку (в количестве от 1 до 255).²³

Соответственно, декларация **S: String [100]** в блоке переменных обозначает, что переменная может хранить в себе последовательность элементов до 100 символов включительно. Декларировать количество символов может быть удобно, когда мы работаем с формализованными данными, например, с серией и номером документа с определенным числом символов. Но при этом следует учесть следующее: при попытке присвоения такой переменной более длинного ряда символов, не вошедшие символы будут «обрублены».

Например, описав переменную: S: String [8] следующая операция

```
S:='Мама мыла раму'
```

даст в результате S=«Мама мыл» и соответственно, потерю оставшихся символов.

Также, если в одной процедуре входные или выходные переменные отличались по своему числу символов от переменных в другой процедуре или функции, компилятор вернет ошибку и укажет на разность типов переменных. В таком случае нам потребуется, чтобы число декларированных в переменных символов совпадало и не превышало 255. Если последнее условие представляет затруднения, но представляет крайнюю необходимость, помогут методы обработки длинных строк.

²³ По существу, строка является массивом символьных переменных типа **Char**, описываемого по умолчанию как содержимое [1..255], вызов к индексу которого аналогичен операции с традиционным массивом (array). При этом, нулевой индекс [0] содержит информацию о текущей длине массива символов.

Длинные строки

В Паскале возможна работа с длинными строками,²⁴ длина которых определяется не числовым значением первого байта строки, а замыкающим символом «#0». Ограничение в них определяется лишь объемом оперативной памяти или ее схемой адресации. Максимально возможная строка в таком случае ограничена числом в 65 535 символов; еще такие строки называют ASCIIZ-строками.

²⁴ Методы, процедуры и функции обработки длинных строк и обычных в ТМТ Pascal и в других языках программирования существенно отличаются, – с длинными строками объективно работать сложнее и они имеют ряд критических недостатков. В дальнейшем повествовании они не будут использоваться. Поэтому вы можете достаточно безболезненно пропустить этот раздел, или только бегло с ним ознакомиться.

Конец ознакомительного фрагмента.

Текст предоставлен ООО «ЛитРес».

Прочитайте эту книгу целиком, [купив полную легальную версию](#) на ЛитРес.

Безопасно оплатить книгу можно банковской картой Visa, MasterCard, Maestro, со счета мобильного телефона, с платежного терминала, в салоне МТС или Связной, через PayPal, WebMoney, Яндекс.Деньги, QIWI Кошелек, бонусными картами или другим удобным Вам способом.