

УДК 004.896

doi:10.31799/1684-8853-2021-5-40-50

## Адаптация нейронных машин Тьюринга для задачи агрегации лингвистических оценок в нейросимволических системах поддержки принятия решений

А. В. Демидовский<sup>а</sup>, аспирант, старший преподаватель, [orcid.org/0000-0003-3605-6332](https://orcid.org/0000-0003-3605-6332), [ademidovskij@hse.ru](mailto:ademidovskij@hse.ru)

Э. А. Бабкин<sup>а</sup>, канд. техн. наук, заведующий лабораторией, профессор, [orcid.org/0000-0003-2597-9043](https://orcid.org/0000-0003-2597-9043)

<sup>а</sup>Национальный исследовательский университет «Высшая школа экономики», Нижегородский филиал, ул. Большая Печерская, 25/12, Нижний Новгород, 603155, РФ

**Введение:** построение интегрированных нейросимволических систем является актуальной и сложной задачей. Построение нейросимволических систем поддержки принятия решений требует новых подходов к представлению знаний о проблемной ситуации и выражению символических рассуждений на субсимволическом уровне. **Цель:** разработка нейросетевых архитектур и методов для эффективного представления знаний в распределенном виде и субсимволических рассуждений в системах поддержки принятия решений в части алгоритмов агрегирования нечетких оценок экспертов для выбора альтернативных решений. **Методы:** представление нечетких и неопределенных оценок в распределенном виде с помощью тензорных представлений, построение обучаемой нейросетевой архитектуры для субсимволической агрегации лингвистических оценок. **Результаты:** предложены два новых метода представления лингвистических оценок в распределенном виде. Первый подход основывается на возможности перевода произвольной лингвистической оценки в численное представление и заключается в переводе этого численного представления в распределенное за счет перевода самого числа в вид битовой строки и дальнейшего формирования матрицы, хранящей распределенное представление всего выражения для агрегации оценок. Второй подход основывается на представлении лингвистической оценки в виде дерева и кодировании этого дерева с помощью метода тензорных представлений, что позволяет избежать шага перевода лингвистической оценки в численный вид и обеспечивает переход между символическим и субсимволическим представлениями лингвистических оценок без потери информации. Структурные элементы лингвистической оценки рассматриваются как заполнители с соответствующими позиционными ролями. Предложен новый субсимволический метод агрегации лингвистических оценок, который заключается в создании обучаемого нейросетевого модуля в виде нейронной машины Тьюринга. **Практическая значимость:** результаты исследования демонстрируют, как символический алгоритм агрегации лингвистических оценок может быть реализован коннективистскими, или субсимволическими, механизмами, что является необходимым требованием при построении распределенных нейросимволических систем поддержки принятия решений.

**Ключевые слова** – нейронные машины Тьюринга, искусственные нейронные сети, субсимволические системы, гибридные интеллектуальные системы, многокритериальное принятие решений.

**Для цитирования:** Демидовский А. В., Бабкин Э. А. Адаптация нейронных машин Тьюринга для задачи агрегации лингвистических оценок в нейросимволических системах поддержки принятия решений. *Информационно-управляющие системы*, 2021, № 5, с. 40–50. doi:10.31799/1684-8853-2021-5-40-50

**For citation:** Demidovskij A. V., Babkin E. A. Adapting Neural Turing Machines for linguistic assessments aggregation in neural-symbolic decision support systems. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2021, no. 5, pp. 40–50 (In Russian). doi:10.31799/1684-8853-2021-5-40-50

### Введение

Долгое время области искусственного интеллекта и анализа решений развивались параллельно несмотря на то, что обе исследуют формальные модели знаний и опыта человека [1, 2]. Системы поддержки принятия решений (СППР) были разработаны как инструмент для автоматизации сбора и анализа мнений экспертов и заинтересованных сторон в целях оказания помощи лицу, принимающему решения. При построении СППР следует учитывать многочисленные факторы, в том числе сбор неоднозначных экспертных оценок альтернативных вариантов, использование неполных данных и т. п. Такие сложности создали общую основу для междис-

циплинарной коммуникации, и для решения вышеупомянутых вопросов были разработаны различные подходы.

Одним из центральных является вопрос, как представить знания таким образом, чтобы это было эффективно с точки зрения вычислений. То, как знания представляются и моделируются человеческим мозгом, на данный момент определяет два различных течения научной мысли. Многочисленные исследователи мотивированы тем, что когнитивная система человека использует невероятно широкую нейронную сеть в качестве вычислительного двигателя, и предлагают считать, что когнитивная система человека использует распределенное представление информации и обрабатывает ее динамически и осмыс-

ленно. Такой нейронный подход к представлению и обработке информации называют субсимволическим, или коннективистским. Одновременно с этим другие исследователи советуют рассматривать познание человека как метод символических манипуляций [3], а именно, как «вид вычислений, выполняемых в конкретном типе биологической системы» [4]. В этом случае символы выступают в качестве входов в такую систему, а затем с помощью predetermined правил и наборов инструкций преобразуются в другие символы. Такой подход называют символическим.

Субсимволические подходы по своей природе распределены и имеют высокую эффективность и надежность. Кроме того, они включают в себя важный компонент обучения, позволяя обучать субсимволические методы для решения конкретной задачи и постоянно модифицировать их по мере поступления новой информации. В то же время такие подходы сложно интерпретировать, хотя развивающаяся область, называемая объяснимым искусственным интеллектом (Explainable AI, xAI), посвящена разработке методов извлечения смысла из сложных вычислительных моделей. Другим недостатком субсимволических методов является их неспособность учитывать сложные взаимосвязи из-за ограничений современных методов распределенного представления информации [3, 4]. Некоторые достижения в этой области позволили решить ряд проблем распределенного представления, связанных с субсимволическими архитектурами: тензорные представления [5], способные представлять рекурсию в символьных структурах; дальнейшее развитие идей о более компактных распределенных представлениях с помощью голографических сокращенных представлений [6] и других методов выражения символьных вычислений. Как правило, этот класс методов называют векторными символьными архитектурами [7, 8]. С другой стороны, хотя символические подходы можно интерпретировать, они по определению являются последовательными. Важно, что символические подходы предполагают креативность правил и инструкций по обработке информации, что приводит к двум последствиям. Во-первых, информация, закодированная в символьной системе программистом, отражает его собственный опыт и мировоззрение, делая систему предвзятой. Этот вопрос был идентифицирован как проблема символьного основания (symbol grounding problem) [9]. Во-вторых, информация, хранящаяся в символической СППР, быстро теряет свою актуальность.

В результате интеграции символического и субсимволического подходов формируются такие архитектуры и системы, как ACT-R [10], CLARION [11] и SS-RICS [12, 13]. В пользу построения интегрированных систем также гово-

рит и психофизиологическая правдоподобность такой интеграции, которая была показана и установлена путем демонстрации наличия взаимосвязанных архитектур в нервной системе человека и различных биологических архитектур [3]. Построение нейросимволических интегрированных систем является актуальной задачей в области анализа решений [14, 15]. Такие системы будут сочетать субсимволические рассуждения и вычисления на коннективистском уровне с символическими рассуждениями и вычислениями на символическом уровне. Однако постоянно возникают споры о том, что значит для СППР стать нейросимволической или интегрированной [1, 15, 16]. Особое внимание уделяется тому, какие компоненты СППР могут быть делегированы на субсимволический уровень и какую функцию искусственные нейронные сети могут играть в СППР [17, 18].

Данная статья посвящена специальному аспекту СППР — агрегации оценок экспертов. Любая задача, связанная с принятием решения, описывается, в частности, проблемой, альтернативными решениями, требованиями и экспертными оценками. Агрегирование оценок является сложной задачей по целому ряду причин, включая нечеткий характер оценок, таких как лингвистические, большое число альтернативных решений и неравные степени компетентности экспертов. Агрегирование оценок является общим этапом в различных методах принятия решений, включая TOPSIS [19], ELECTRE [20] и ML-LDM [21]. Таким образом, построение нейросимволической СППР может начинаться с выражения агрегирования нечетких оценок на нейросетевом уровне. Целью данной работы является разработка нового субсимволического метода агрегации лингвистических оценок в виде обучаемого нейросетевого модуля.

### Агрегация лингвистических оценок с помощью мультигранулярного оператора агрегации

Современные методы многоаспектного многоуровневого принятия решений используют в качестве базового строительного блока двухкортежную модель представления лингвистической информации — модель 2-tuple. Важной особенностью данной модели является способность выражать как качественные, так и количественные оценки. Модель 2-tuple основана на концепции символического перевода [22].

*Определение 1.* Структура 2-tuple состоит из пары  $(s_i, \alpha)$ , где  $s_i \in S = \{s_0, \dots, s_g\}$  — лингвистический терм (концепт);  $\alpha$  — численное значение, результат символического перевода, значение

функции принадлежности. Данное значение показывает расстояние до ближайшего концепта  $s_i \in S = \{s_0, \dots, s_g\}$ , если функция принадлежности не дает точного ответа ( $s_i$ ).

**Определение 2.** Лингвистической шкалой  $S$  [23] называется конечное упорядоченное множество термов  $S = \{s_i | i = 1 \dots T\}$ , удовлетворяющее следующим условиям:

- 1) если  $i < j$ , то  $s_i$  предшествует  $s_j$  ( $s_i < s_j$ );
- 2) отрицание терма определяется правилом  $Neg(s_i) = s_{T-i+1}$ ;
- 3) пусть  $s_i < s_j$ , тогда объединение (дизъюнкция, связка «ИЛИ») термов определяется правилом  $s_i \vee s_j = \max\{s_i, s_j\} = s_j$ ;
- 4) пусть  $s_i < s_j$ , тогда пересечение (конъюнкция, связка «И») термов определяется правилом  $s_i \wedge s_j = \min\{s_i, s_j\} = s_i$ .

**Определение 3.** Правило перевода. Пусть  $S = \{s_0, \dots, s_g\}$  — лингвистическая шкала, где  $g = \tau + 1$  обозначает уровень гранулярности  $S$ . Если  $\beta \in [0, 1]$  — это результат символической агрегации, то существует однозначный способ построить соответствующий элемент типа 2-tuple:

$$\begin{aligned} \Delta_g &= [0, 1] \rightarrow S \times [-0,5; 0,5), \\ \Delta_g(\beta) &= (s_i, \alpha), \\ i &= \text{round}(\beta\tau), \\ \alpha &= \beta\tau - i, \alpha \in [-0,5; 0,5). \end{aligned} \quad (1)$$

**Определение 4.** Правило обратного перевода. Пусть  $S = \{s_0, \dots, s_g\}$  — лингвистическая шкала, где  $g = \tau + 1$  обозначает уровень гранулярности  $S$ . Пусть  $(s_i, \alpha)$  — элемент типа 2-tuple, определенный на лингвистической шкале  $S$ , где  $\alpha \in [-0,5; 0,5)$ . Тогда есть способ преобразования данного элемента в численное представление  $\beta \in [0, 1]$ :

$$\begin{aligned} \Delta_g^{-1} &= S \times [-0,5; 0,5) \rightarrow [0, 1], \\ \Delta_g^{-1}(s_i, \alpha) &= (i + \alpha)/\tau. \end{aligned} \quad (2)$$

Существует множество способов агрегирования оценок, выраженных в форме 2-tuple, их обычно называют операторами: МТWA (Multigranularity 2-tuple Weighted Averaging), МНТWA (Multigranularity Hesitant 2-tuple Weighted Averaging), P2TLWA (Pythagorean 2-tuple Linguistic Weighted Averaging) [24] и т. д. Один из них, МТА, выполняет вычисление средневзвешенного значения по набору 2-tuple.

**Определение 5.** Оператор МТА (Multigranular 2-tuple Averaging). Пусть  $(b_i, \alpha_i)$  — элемент типа 2-tuple, определенный на шкале  $S_i^g$ ,  $i = 1, 2, \dots, n$ . Тогда оператор МТА определяется как

$$\begin{aligned} \text{MTA}((b_1, \alpha_1), (b_2, \alpha_2), \dots, (b_n, \alpha_n)) &= \\ = \Delta_{gh}(\sum_{j=1}^n (1/n \Delta_{gj}^{-1}(b_j, \alpha_j))). \end{aligned} \quad (3)$$

Несмотря на то, что модель 2-tuple является фундаментальной, в последние годы были предложены многочисленные методы агрегирования нечетких оценок: Hesitant Fuzzy Linguistic Term Sets (HFLTST) [25], Institutional 2-tuple [26], гибридные модели [27] и т. д. Каждый оператор обычно ассоциируется с набором арифметических операций.

Одним из возможных методов построения нейросимволических СППР является динамическое выражение арифметических операций в виде динамики нейронной сети. Настоящая статья является частью серии статей о выстраивании нейросимволической архитектуры СППР [28]. Однако здесь используется подход к проектированию СППР, отличный от рассмотренного в работе [28], где предложено создание таких нейросетевых архитектур, которые не требуют обучения и работают поверх лингвистических оценок, закодированных с помощью тензорных представлений [29]. В данной статье предлагается агрегировать лингвистические оценки в виде обучаемого нейросетевого модуля, построенного по принципам архитектуры нейронной машины Тьюринга (НМТ, Neural Turing Machine — NTM).

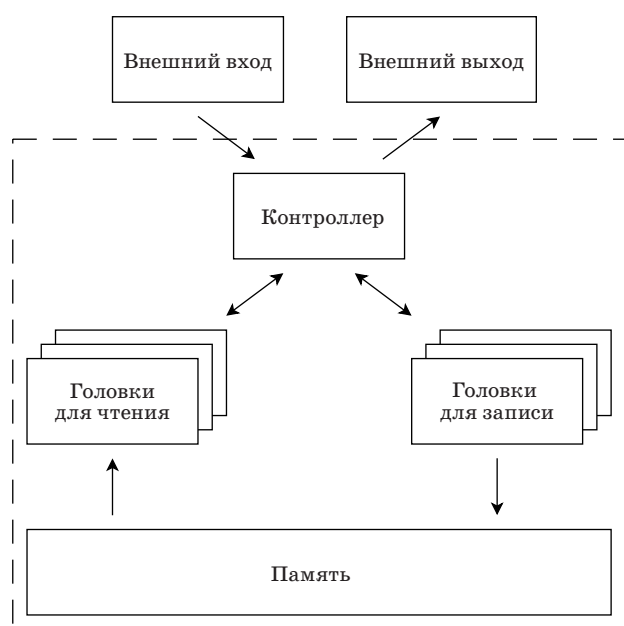
## Нейронные машины Тьюринга

### Архитектура

Нейронные машины Тьюринга впервые были предложены в работах [30, 31] и с тех пор получили широкое распространение в различных задачах, начиная от простых алгоритмических задач и заканчивая обучением с подкреплением [32], последовательными рекомендациями [33], трансдукцией естественного языка [34] и др.

Нейронные машины Тьюринга есть частный пример отдельного класса ИНС с дополненной памятью, которая играет важнейшую роль и является внешней по отношению к внутреннему состоянию нейросети. Для записи в эту память и чтения из нее существуют специальные абстракции, называемые записывающими и считывающими головками соответственно. Контроллер выполняет координацию этих головок для получения результата. Теоретическая архитектура НМТ показана на рис. 1.

Нейронная машина Тьюринга обучается с учителем. На этапе обучения НМТ ожидает набор векторных последовательностей в качестве входа и набор меток, который является еще одной последовательностью векторов и обозначает ожидаемый выход сети. На выходе НМТ выдает пакет векторных последовательностей, таким образом, НМТ может выполнять одну и ту же задачу над несколькими последовательностями одновременно. Доказано, что НМТ может решать мно-



■ *Рис. 1.* Общий дизайн НМТ  
 ■ *Fig. 1.* General design of NTM

жество задач, таких как копирование, повтор, ассоциативный поиск, сложение и умножение двоичных чисел, задачу приоритетной сортировки. В данной работе рассматривается выражение оператора МТА для чисел, представленных в двоичном виде различным количеством бит: 4, 6, 8, 10. Общая схема подачи данных в нейронную сеть показана на рис. 2.

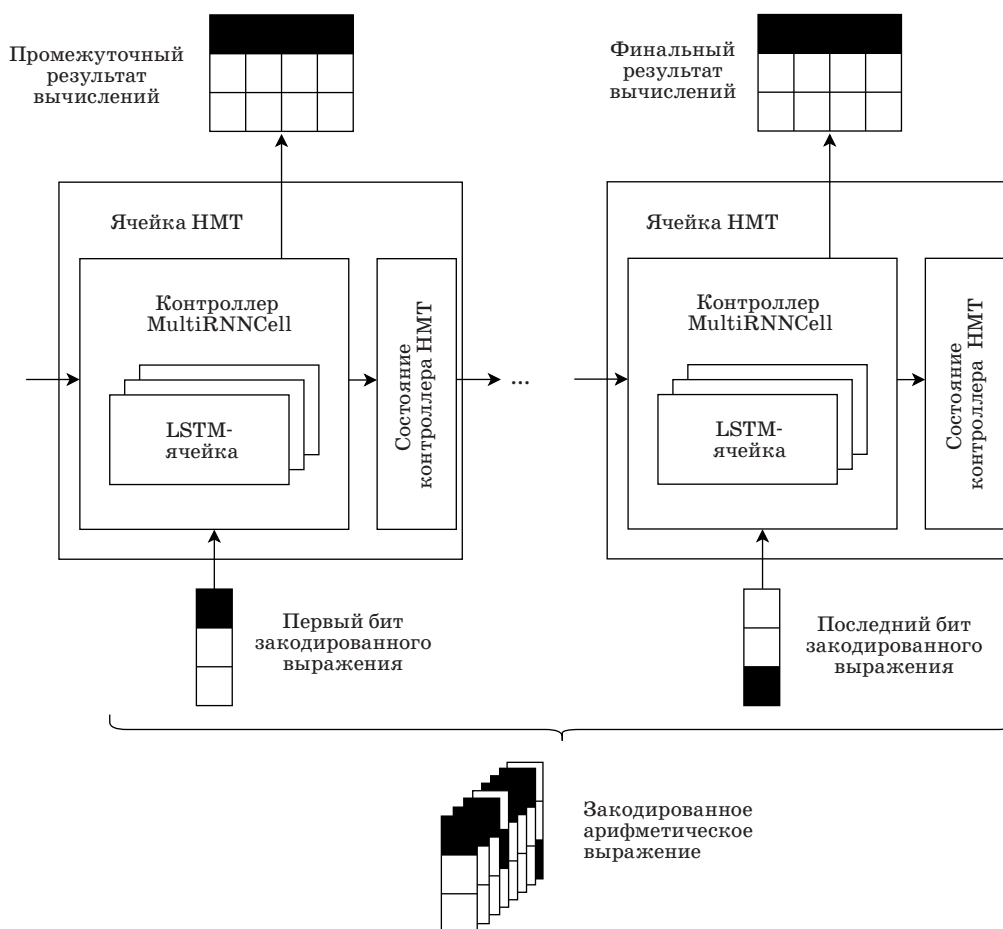
**Аспекты реализации**

Аспекты внедрения НМТ хорошо описаны в работах [35, 36] и в основном охватывают различные подходы к повышению устойчивости обучения за счет внедрения механизма контроля над градиентом. Важно отметить функцию ошибки, которая была использована как критерий для оценки качества обучения модели:

$$\delta_{i \geq 0,5} = \{(1, i \geq 0,5), (0, i < 0,5)\}; \quad (4)$$

$$e = \sum_{b=1}^B \sum_{n=1}^{N+1} \sum_{m=1}^M |\delta_{X(b,n,m) \geq 0,5} - Y_{b,n,m}| \quad (5)$$

где  $B$  — размер партии обучающих данных, поступающих в сеть;  $N$  — количество бит на одно



■ *Рис. 2.* Выполнение закодированного выражения для МТА с помощью НМТ  
 ■ *Fig. 2.* Executing NTM with encoded expression for MTA operator

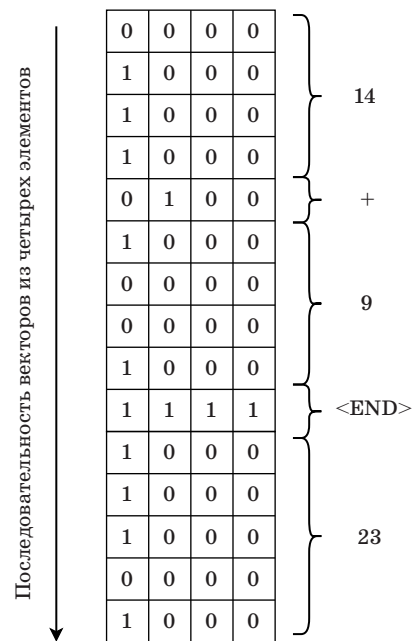
число;  $M$  — длина вектора, содержащего один бит выражения.

Архитектура НМТ предполагает создание одной или нескольких НМТ-ячеек, где каждая включает в себя контроллер, внешнюю память и головки для работы с этой памятью. Более того, НМТ способна к обобщению той задачи, для которой она была сконструирована и обучена. В частности, НМТ может обучаться на векторных последовательностях определенной длины, и после обучения сеть может решить задачу для большего набора последовательностей. Все приведенные ниже результаты могут быть воспроизведены с помощью проекта с открытым исходным кодом. Проект построен на основе текущей реализации системы TensorFlow Neural Turing Machine [22]. Однако были произведены значимые улучшения — осуществлен переход на поддерживаемую версию TensorFlow (2.3.0), так как базовая реализация НМТ основывалась на версии TensorFlow, которая более не поддерживается компанией Google.

### Оценка НМТ как нейросетевого оператора МТА

В данной работе с помощью НМТ мы выражаем оператор МТА, который агрегирует оценки от фиксированного числа экспертов; каждая оценка преобразуется в числовую форму и представляется в виде двоичной строки [36]. Затем численные представления оценок объединяются в одно арифметическое выражение, которое уже кодируется в виде матрицы с определенным форматом. Каждое число представлено в виде матрицы с  $N$  строками и  $M$  столбцами, где  $N$  обозначает количество бит, используемых для кодирования числа в формате Little-Endian. Мы используем множество «каналов» в данных, чтобы помочь сети различать биты, кодирующие число, и биты, кодирующие маркеры. Первый столбец матрицы содержит биты данных, второй — биты для маркеров, а конечный маркер представлен вектором длины  $M$ . Для обучения создан набор данных, в котором каждое выражение закодировано в определенном формате и каждый обучающий набор содержит 32 арифметических выражения. Каждое закодированное выражение содержит либо две, либо три лингвистические оценки, представленные в численном виде и закодированные согласно схеме на рис. 3. В рамках данного исследования проводились эксперименты с различной длиной битовой строки: 4, 6, 8, 10, 16 бит.

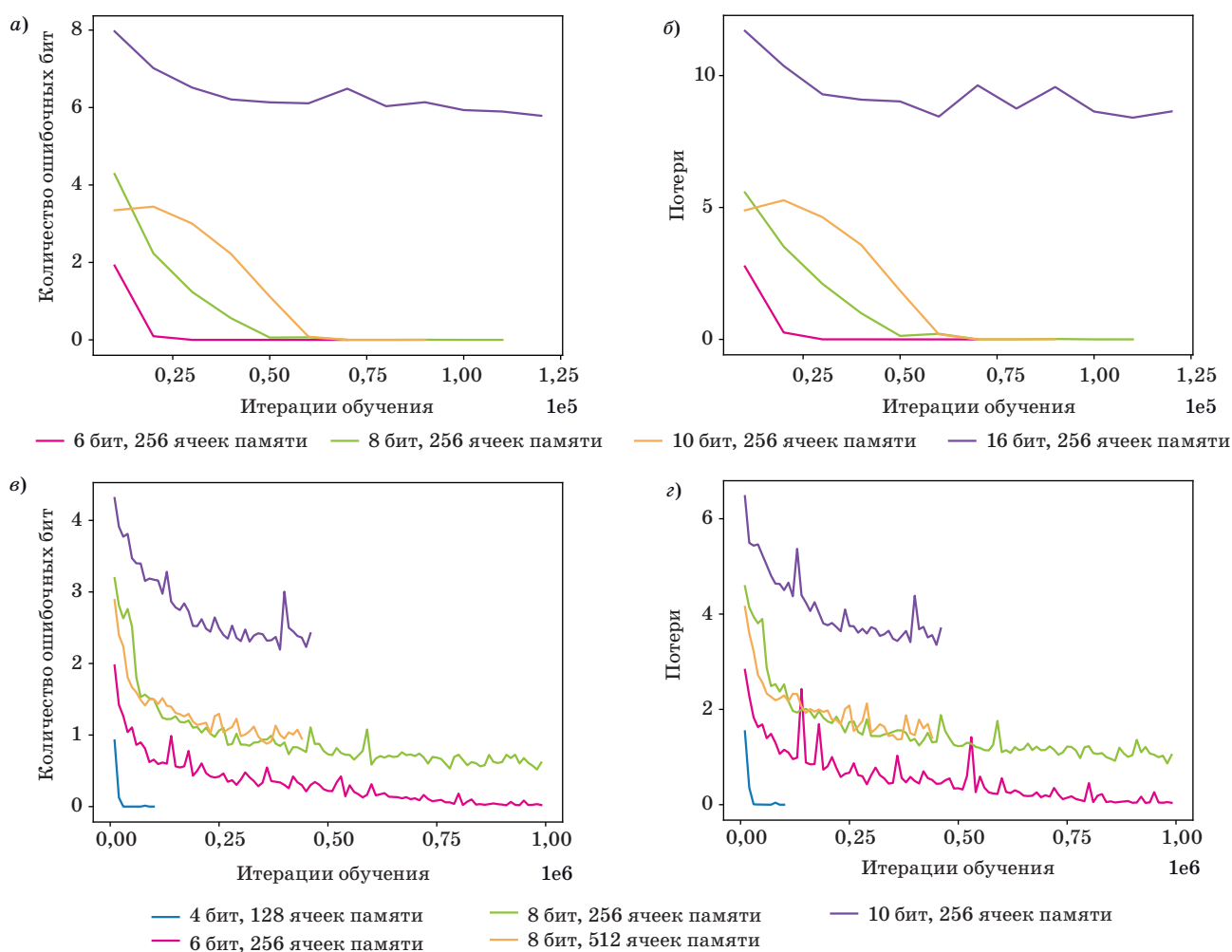
С точки зрения гиперпараметров и внутреннего устройства нейронной сети Тьюринга в рамках данного исследования использовались следующие значения. У НМТ одна считывающая и одна записывающая головка. Контроллер представлен



■ Рис. 3. Закодированное выражение для оператора МТА  
 ■ Fig. 3. Encoded expression for the MTA operator

одной LSTM-ячейкой. Размер LSTM-ячейки равен 100 элементам и не менялся в зависимости от входных данных. Размер внешней памяти составлял 256 ячеек, каждая состояла из 128 элементов. Для каждого эксперимента, который характеризуется количеством экспертов и длиной битовой строки, производилось отдельное обучение нейронной сети и отдельная серия экспериментов после обучения. В случае агрегации двух оценок НМТ смогла добиться нулевой ошибки в экспериментах с 4-, 6-, 8- и 10-битными числами (рис. 4, а). При агрегации трех оценок НМТ смогла добиться нулевой ошибки только для 4-битных чисел (рис. 4, в). Для большей длины битовой строки ошибка варьируется от 0,26 для 6 бит до 2,42 для 10-битных чисел. В то же время НМТ смогла существенно минимизировать значение функции потерь во всех вариантах (рис. 4, б, г). Эксперименты проводились на различных конфигурациях CPU (таблица): Intel(R) Xeon(R) CPU E5-2650 v4 @ 2.20GHz, AMD EPYC 7282 16-Core Processor @ 2.79 GHz, Intel(R) Xeon(R) Gold 6140 CPU @ 2.30 GHz с нефиксированной частотой и 60 GB оперативной памяти и GPU: NVIDIA Corporation TU104GL [Tesla T4]. Результаты экспериментов можно воспроизвести с использованием авторского проекта с открытым исходным кодом (<https://github.com/demid5111/NeuralTuringMachine>).

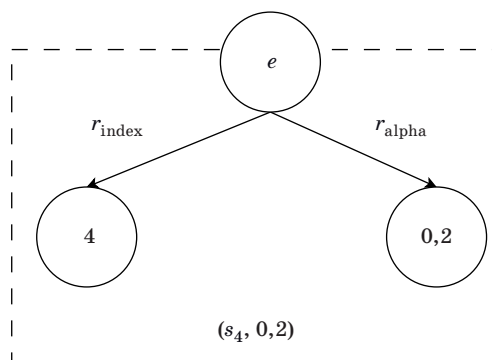
Ограничением подхода, апробированного в первой серии экспериментов, является необходимость промежуточного перевода лингвистических



■ **Рис. 4.** Динамика обучения НМТ для агрегации оценок, представленных в виде битовых строк фиксированной длины. Изменение функции ошибки (а) и потерь (б) НМТ при агрегации двух оценок; изменение функции ошибки (в) и потерь (з) НМТ при агрегации трех оценок

■ **Fig. 4.** NTM training dynamics for bit strings of various length. Change of error per sequence for two assessments aggregation (а), loss function for two assessments aggregation (б), error per sequence for three assessments aggregation (в), loss function for three assessments aggregation (з)

оценок в численное представление перед тем, как передавать их на вход обученному нейросетевому агрегатору оценок. Более того, как уже отмечалось, в рамках задачи построения нейросимволических систем вопрос представления знаний является актуальным. Требуется создание методов для перевода символов в истинно распределенное представление. Поэтому с точки зрения решаемой задачи агрегации лингвистических оценок необходимо построение распределенного представления для каждой такой оценки и обучение нейросетевого агрегатора, который бы мог работать с такими распределенными представлениями. Для этого предлагается рассматривать лингвистическую оценку с точки зрения тензорных представлений [28], а именно как дерево с двумя листьями. Например, для лингвистической оценки, представленной на рис. 5, при заданных  $r_{\text{index}} = [1, 0]$  и  $r_{\text{alpha}} = [0, 1]$ ,



■ **Рис. 5.** Представление лингвистической оценки  $(s_4, 0,2)$  в виде структуры из двух заполнителей, т. е. листьев дерева, и двух позиционных ролей:  $r_{\text{index}}$  и  $r_{\text{alpha}}$

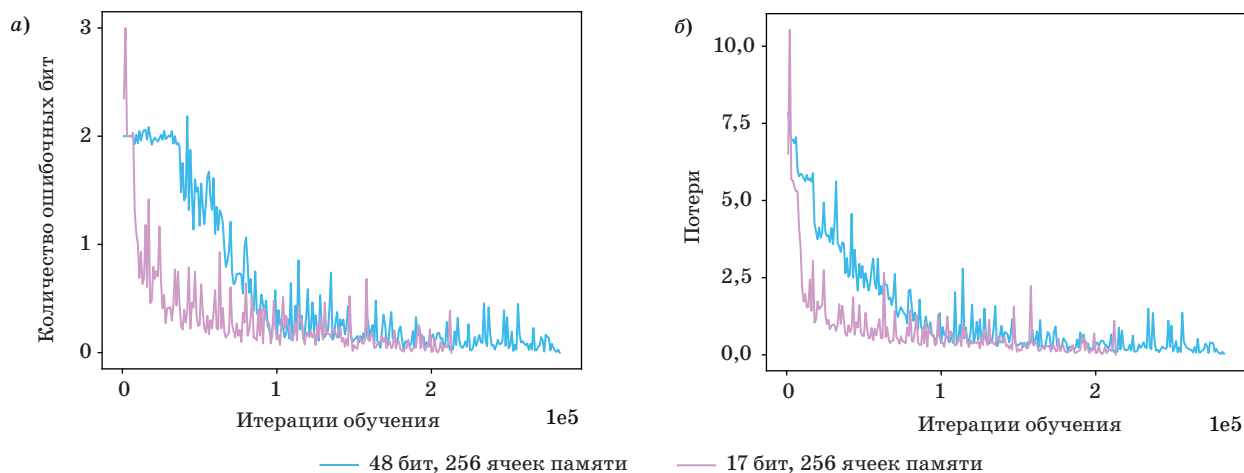
■ **Fig. 5.** Representing linguistic assessment  $(s_4, 0,2)$  as a structure with two fillers as leaves and two positional roles:  $r_{\text{index}}$  and  $r_{\text{alpha}}$

размере заданной лингвистической шкалы, равной пяти, и фиксированной точности для представления проекции лингвистической переменной, равной одному знаку после запятой, получаем следующее распределенное представление:  $\psi = [[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0], [0, 0], [0, 0], [0, 0], [1, 0], [0, 0], [0, 0], [0, 0], [0, 0], [0, 0], [0, 0], [0, 0], [0, 0], [0, 0], [0, 1], [0, 0], [0, 0], [0, 0]]$ . При вытягивании такого распределенного представления в вектор получается тензор ранга 1, состоящий из 48 элементов.

Вторая серия экспериментов была посвящена подготовке данных и обучению нейросетевого агрегатора, который принимает на вход набор из распределенных представлений лингвистических оценок. Иными словами, по сравнению с первой серией экспериментов больше не требовался предварительный шаг по переводу лингвистических оценок в численное представление перед их упаковкой в качестве входных данных для нейросетевого агрегатора. Общая схема подготовки данных для первого и второго экспериментов выглядит аналогично. Во втором эксперименте вместо распределенного представления чисел в виде битовой строки заданной длины используется распределенное представление лингвистических оценок в виде битовой строки фиксированной длины, равной 48. Эксперименты проводились с агрегацией оценок только от двух экспертов. С точки зрения гиперпараметров и внутреннего устройства нейронной сети Тьюринга в рамках второго эксперимента использовались те же настройки, что и в первом эксперименте. Результат обучения нейросетевого агрегатора представлен на рис. 6, а и б. Вычислительные эксперименты проводились на Intel(R) Xeon(R) Gold 6240 CPU @ 2.30 GHz с нефиксированной частотой и 60 GB оперативной памяти.

Таким образом, нейросетевой агрегатор принимает на вход лингвистические оценки в распределенном виде, закодированные согласно правилам тензорных представлений, и выполняет задачу агрегации лингвистических оценок от двух экспертов с нулевой ошибкой.

В экспериментальном исследовании производилась агрегация оценок от фиксированного количества экспертов. Учитывая устройство МТА и фиксированное количество экспертов, данный оператор можно представить в виде единственного нейрона обычной сети прямого распространения (без функции активации). Использование существенно более сложной нейросетевой архитектуры для реализации этого оператора вызвано рядом причин. Во-первых, с точки зрения развития идей распределенных вычислений в СППР существует более широкая задача, которая заключается в разработке методов построения истинно распределенного представления информации о проблемной ситуации, которая, например, включает и информацию об экспертах, альтернативах и критериях. Веса в таком случае не должны быть закодированы в локальном виде, что происходит, например, при использовании единственного нейрона и некоторого количества обучаемых весов, соответствующих конкретному эксперту или критерию. Во-вторых, исследовательский вопрос заключался в попытке избавиться от необходимости создания новых символических операторов агрегации, чему посвящено большое количество работ, и их число с каждым годом только растет. Вместо этого предлагается использовать нейросетевую архитектуру, которая могла бы функционировать как оператор агрегации, при этом правила агрегации подбирались бы нейронной сетью самостоятельно. В-третьих, существу-



■ **Рис. 6.** Динамика функции ошибки (а) и потерь (б) НМТ для агрегации двух экспертных оценок, представленных в распределенном виде с помощью тензорных представлений

■ **Fig. 6.** Change of error per sequence for aggregation of two assessments encoded with tensor representations in a distributed form

- Таблица экспериментов по обучению НМТ
- NTM training experiments details

Номер серии экспериментов	Количество оценок	Размер битовой строки	Количество слотов внешней памяти	Количество ошибочных бит в результате работы НМТ	Аппаратное обеспечение
1	2	4	128	0	Intel(R) Xeon(R) CPU E5-2650 v4 @ 2.20 GHz
1	2	6	256	0,26	AMD EPYC 7282 16-Core Processor @ 2.79 GHz
1	2	8	256	0,62	Intel(R) Xeon(R) CPU E5-2650 v4 @ 2.20 GHz
1	2	8	512	0,26	Intel(R) Xeon(R) Gold 6140 CPU @ 2.30 GHz
1	2	10	256	0,62	Intel(R) Xeon(R) Gold 6140 CPU @ 2.30 GHz
1	3	4	128	0	AMD EPYC 7282 16-Core Processor @ 2.79 GHz
1	3	6	256	0	AMD EPYC 7282 16-Core Processor @ 2.79 GHz
1	3	8	256	0	AMD EPYC 7282 16-Core Processor @ 2.79 GHz
1	3	10	256	0	Intel(R) Xeon(R) Gold 6240 CPU @ 2.60 GHz
1	3	16	256	5,58	NVIDIA Corporation TU104GL [Tesla T4]
2	2	17	256	0	Intel(R) Xeon(R) Gold 6240 CPU @ 2.60 GHz
2	2	48	256	0	Intel(R) Xeon(R) Gold 6240 CPU @ 2.60 GHz

ет задача построения компилятора нейросетевых каскадов для различных конфигураций СППР и для работы с произвольным сценарием. В такой постановке задачи использование простейшего варианта в виде нейрона нецелесообразно, в то время как существенно более сложная нейросетевая архитектура позволяет решать поставленную задачу.

## Заключение

Цель данного исследования заключалась в разработке нейросетевой архитектуры для эффективного представления субсимволических рассуждений в системах поддержки принятия решений в части алгоритмов агрегирования нечетких оценок экспертов для выбора альтернативных решений. Данная работа является частью серии статей о выстраивании нейросимволической архитектуры СППР [28]. В рамках общего исследования рассматривается шаг агрегации оценок при выполнении многокритериального выбора и осуществляется развитие методов к выражению алгоритмов агрегации в полностью коннективистском виде. Было продемонстрировано [28], что задача агрегации оценок может быть осуществлена с помощью нейросетевых архитектур, не требующих обучения. Однако показано, что такое кодирование чисел требует создания древовидных структур большой вложенности, а одним из известных недостатков тензорных представлений является взрывной рост размерности. Обработка таких распределенных представлений

большой размерности требует значительных вычислительных ресурсов и также является затратной по времени [37].

Настоящая работа показывает, что шаг агрегации оценок может быть реализован на основе обучения нейронной сети, а уже обученная нейронная сеть может достаточно быстро и точно агрегировать оценки. При этом мы считаем важным использовать тензорные представления для кодирования проблемной ситуации, а также самих оценок в силу отсутствия потерь информации. Иными словами, вклад данной работы в общее исследование построения нейросимволических СППР заключается в демонстрации возможности создания и практического использования обучаемого нейросетевого модуля на основе нейронных машин Тьюринга для агрегации оценок. Кроме того, предлагается использовать метод тензорных представлений для кодирования лингвистических оценок в распределенный вид. Это позволяет при необходимости переходить с символического уровня на субсимволический и обратно, что создает возможность для интерпретации работы нейросимволической СППР. В качестве дальнейших направлений исследования предлагается рассмотреть возможности НМТ для выражения другого алгоритма агрегации — МТWA оператора, так как он является самым распространенным оператором, используемым при разработке лингвистических СППР. Также интерес представляет исследование возможности создания распределенного представления проблемной ситуации и дальнейшего выражения всех этапов агрегации



экспертных оценок в виде динамики нейронной сети. Разработка таких решений, на наш взгляд, обеспечивает фундаментальные строительные блоки для распределенных гибридных нейросимволических СППР.

## Литература

1. Matzkevich I., Abramson B. Decision analytic networks in artificial intelligence. *Management Science*, 1995, vol. 41, no. 1, pp. 1–22. doi:10.1287/MNSC.41.1.1
2. Silverman B. G. Unifying expert systems and the decision sciences. *Operations Research*, 1994, vol. 42, no. 3, pp. 393–413. doi:10.1287/opre.42.3.393
3. Kelley T. D. Symbolic and sub-symbolic representations in computational models of human cognition: What can be learned from biology? *Theory & Psychology*, 2003, vol. 13, no. 6, pp. 847–860. doi:10.1177/0959354303136005
4. Pylyshyn Z. W. *Computing in cognitive science. Foundations of cognitive science*. Cambridge, MA, MIT Press, 1989. Pp. 51–91.
5. Smolensky P., Legendre G. *The harmonic mind: From neural computation to optimality-theoretic grammar. Vol. 1: Cognitive architecture*. Cambridge, MA, MIT Press, 2006. 590 p.
6. Plate T. A. Holographic reduced representations. *IEEE Transactions on Neural Networks*, 1995, vol. 6, no. 3, pp. 623–641. doi:10.1109/72.377968
7. Kanerva P. Hyperdimensional computing: An introduction to computing in distributed representation with high-dimensional random vectors. *Cognitive Computation*, 2009, no. 1, pp. 139–159. doi:10.1007/s12559-009-9009-8
8. Schlegel K., Neubert P., Protzel P. A comparison of vector symbolic architectures. *arXiv:2001.11797*, 2020.
9. Harnad S. The symbol grounding problem. *arXiv:cs/9906002v1*, 1999.
10. Anderson J. R., Lebiere C. J. *The atomic components of thought*. Psychology Press, 2014. 504 p.
11. Sun R., Peterson T. A hybrid agent architecture for reactive sequential decision making. In: *Connectionist symbolic integration: From unified to hybrid approaches*. New York, Psychology Press, 1997. Pp. 113–138.
12. Avery E., Kelley T., Davani D. Using cognitive architectures to improve robot control: Integrating production systems, semantic networks, and sub-symbolic processing. *Proc. 15th Annual Conference on Behavioral Representation in Modeling and Simulation (BRIMS)*, 2006, pp. 190–198.
13. Kelley T., Avery E., Long L., Dimperio E. A hybrid symbolic and sub-symbolic intelligent system for mobile robots. *Proc. AIAA Infotech@Aerospace Conference*, 2009, p. 1976. doi:10.2514/6.2009-1976
14. Parvar J., Lowe D., Emsley M., Duff R., Akintoye A. Neural networks as a decision support system for the decision to bid process. *Proc. 16th Annual ARCOM Conference*, 2000, pp. 209–217.
15. Delen D., Sharda R. Artificial neural networks in decision support systems. In: *Handbook on decision support systems 1*. Springer, 2008. Pp. 557–580. doi:10.1007/978-3-540-48713-5\_26
16. Power D. J., Sharda R. Model-driven decision support systems: Concepts and research directions. *Decision Support Systems*, 2007, vol. 43, no. 3, pp. 1044–1061. doi:10.1016/j.dss.2005.05.030
17. Power D. J. *Decision support systems: Concepts and resources for managers*. Greenwood Publishing Group, 2002. 284 p.
18. Golmohammadi D. Neural network application for fuzzy multi-criteria decision making problems. *International Journal of Production Economics*, 2011, vol. 131, no. 2, pp. 490–504. doi:10.1016/j.ijpe.2011.01.015
19. Hwang C., Yoon K. *Multiple attribute decision making methods and applications: a state-of-the-art survey*. New York, Springer-Verlag Berlin Heidelberg, Springer Publications, 1981. 259 p. doi:10.1007/978-3-642-48318-9
20. Figueira J., Greco S., Ehrgott M. *Multiple criteria decision analysis: State of the art surveys*. New York, Springer-Verlag Berlin Heidelberg, Springer Publications, 2005. 1267 p. doi:10.1007/978-1-4939-3094-4
21. Demidovskij A. V., Babkin E. A. Developing a distributed linguistic decision making system. *Business Informatics*, 2019, vol. 13, no 1, pp. 18–32. doi:10.17323/1998-0663.2019.1.18.32
22. Herrera F., Martinez L. A 2-tuple fuzzy linguistic representation model for computing with words. *IEEE Transactions on Fuzzy Systems*, 2000, vol. 8, no. 6, pp. 746–752. doi:10.1109/91.890332
23. Pogoyan K. S. The algorithm for generating optimal linguistic scale for group decision-making. *Proceedings of Voronezh State University: Systems Analysis and Information Technologies*, 2015, no. 2, pp. 97–104.
24. Wei G., Gao H. Pythagorean 2-tuple linguistic power aggregation operators in multiple attribute decision making. *Economic Research-Ekonomiska Istraživanja*, 2020, vol. 33, no. 1, pp. 904–933. doi:10.1080/1331677X.2019.1670712
25. Rodriguez R. M., Martinez L., Herrera F. Hesitant fuzzy linguistic term sets for decision making. *IEEE Transactions on Fuzzy Systems*, 2011, vol. 20, no. 1, pp. 109–119. doi:10.1109/TFUZZ.2011.2170076

## Финансовая поддержка

Исследование выполнено при финансовой поддержке РФФИ в рамках научного проекта № 19-37-90058.

26. Liu P., Chen S.-M. Multiattribute group decision making based on intuitionistic 2-tuple linguistic information. *Information Sciences*, 2018, vol. 430, pp. 599–619. doi:10.1016/j.ins.2017.11.059
27. Wang J.-H., Hao J. A new version of 2-tuple fuzzy linguistic representation model for computing with words. *IEEE Transactions on Fuzzy Systems*, 2006, vol. 14, no. 3, pp. 435–445. doi:10.1109/TFUZZ.2006.876337
28. Demidovskij A., Babkin E. Designing a neural network primitive for conditional structural transformations. *Lecture Notes in Computer Science*, 2020, vol. 12412, pp. 117–133. doi:10.1007/978-3-030-59535-7\_9
29. Demidovskij A., Babkin E. Designing arithmetic neural primitive for sub-symbolic aggregation of linguistic assessments. *Journal of Physics: Conference Series*, 2020, vol. 1680, no. 1, article ID 012007. doi:10.1088/1742-6596/1680/1/012007
30. Graves A., Wayne G., Danihelka I. Neural turing machines. *arXiv:1410.5401*, 2014.
31. Graves A., Wayne G., Reynolds M., Harley T., Danihelka I., Grabska-Barwińska A., Colmenarejo S. G., Grefenstette E., Ramalho T., Agapiou J., Badia A. P., Hermann K. M., Zwols Y., Ostrovski G., Cain A., King H., Summerfield C., Blunsom P., Kavukcuoglu K., Hassabis D. Hybrid computing using a neural network with dynamic external memory. *Nature*, 2016, no. 538, pp. 471–476. doi:10.1038/nature20101
32. Zaremba W., Sutskever I. Reinforcement learning neural Turing machines-revised. *arXiv:1505.00521*, 2015.
33. Chen X., Xu H., Zhang Y., Tang J., Cao Y., Qin Z., Zha H. Sequential recommendation with user memory networks. *Proc. Eleventh ACM International Conference on Web Search and Data Mining (WSDM '18)*, 2018, pp. 108–116. doi:10.1145/3159652.3159668
34. Grefenstette E., Hermann K. M., Suleyman M., Blunsom P. Learning to transduce with unbounded memory. *arXiv:1506.02516*, 2015.
35. Collier M., Beel J. Implementing neural Turing machines. *Lecture Notes in Computer Science*, 2018, vol. 11141, pp. 94–104. doi:10.1007/978-3-030-01424-7\_10
36. Castellini J. Learning numeracy: Binary arithmetic with neural Turing machines. *arXiv:1904.02478*, 2019.
37. Demidovskij A. Encoding and decoding of recursive structures in neural-symbolic systems. *Optical Memory and Neural Networks*, 2021, vol. 30, no. 1, pp. 37–50.

UDC 004.896

doi:10.31799/1684-8853-2021-5-40-50

### Adapting Neural Turing Machines for linguistic assessments aggregation in neural-symbolic decision support systems

A. V. Demidovskij<sup>a</sup>, Post-Graduate Student, Senior Lecturer, orcid.org/0000-0003-3605-6332, ademidovskij@hse.ru

E. A. Babkin<sup>a</sup>, PhD, Tech., Acting Head of Labs, Professor, orcid.org/0000-0003-2597-9043

<sup>a</sup>National Research University Higher School of Economics, Nizhegorodsky branch, 25/12, Bolshaya Pecherskaya St., 603155, Nizhny Novgorod, Russian Federation

**Introduction:** The construction of integrated neurosymbolic systems is an urgent and challenging task. Building neurosymbolic decision support systems requires new approaches to represent knowledge about a problem situation and to express symbolic reasoning at the subsymbolic level. **Purpose:** Development of neural network architectures and methods for effective distributed knowledge representation and subsymbolic reasoning in decision support systems in terms of algorithms for aggregation of fuzzy expert evaluations to select alternative solutions. **Methods:** Representation of fuzzy and uncertain estimators in a distributed form using tensor representations; construction of a trainable neural network architecture for subsymbolic aggregation of linguistic estimators. **Results:** The study proposes two new methods of representation of linguistic assessments in a distributed form. The first approach is based on the possibility of converting an arbitrary linguistic assessment into a numerical representation and consists in converting this numerical representation into a distributed one by converting the number itself into a bit string and further forming a matrix storing the distributed representation of the whole expression for aggregating the assessments. The second approach to translating linguistic assessments to a distributed representation is based on representing the linguistic assessment as a tree and coding this tree using the method of tensor representations, thus avoiding the step of translating the linguistic assessment into a numerical form and ensuring the transition between symbolic and subsymbolic representations of linguistic assessments without any loss of information. The structural elements of the linguistic assessment are treated as fillers with their respective positional roles. A new subsymbolic method of aggregation of linguistic assessments is proposed, which consists in creating a trainable neural network module in the form of a Neural Turing Machine. **Practical relevance:** The results of the study demonstrate how a symbolic algorithm for aggregation of linguistic evaluations can be implemented by connectionist (or subsymbolic) mechanisms, which is an essential requirement for building distributed neurosymbolic decision support systems.

**Keywords** — Neural Turing Machine, artificial neural networks, subsymbolic systems, hybrid intelligent systems, multi-attribute decision making.

**For citation:** Demidovskij A. V., Babkin E. A. Adapting Neural Turing Machines for linguistic assessments aggregation in neural-symbolic decision support systems. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2021, no. 5, pp. 40–50 (In Russian). doi:10.31799/1684-8853-2021-5-40-50

#### Financial support

The reported study was funded by RFBR, project number 19-37-90058.

## References

1. Matzkevich I., Abramson B. Decision analytic networks in artificial intelligence. *Management Science*, 1995, vol. 41, no. 1, pp. 1–22. doi:10.1287/MNSC.41.1.1
2. Silverman B. G. Unifying expert systems and the decision sciences. *Operations Research*, 1994, vol. 42, no. 3, pp. 393–413. doi:10.1287/opre.42.3.393
3. Kelley T. D. Symbolic and sub-symbolic representations in computational models of human cognition: What can be learned from biology? *Theory & Psychology*, 2003, vol. 13, no. 6, pp. 847–860. doi:10.1177/0959354303136005
4. Pylyshyn Z. W. *Computing in cognitive science. Foundations of cognitive science*. Cambridge, MA, MIT Press, 1989. Pp. 51–91.
5. Smolensky P., Legendre G. *The harmonic mind: From neural computation to optimality-theoretic grammar. Vol. 1: Cognitive architecture*. Cambridge, MA, MIT Press, 2006. 590 p.
6. Plate T. A. Holographic reduced representations. *IEEE Transactions on Neural Networks*, 1995, vol. 6, no. 3, pp. 623–641. doi:10.1109/72.377968
7. Kanerva P. Hyperdimensional computing: An introduction to computing in distributed representation with high-dimensional random vectors. *Cognitive Computation*, 2009, no. 1, pp. 139–159. doi:10.1007/s12559-009-9009-8
8. Schlegel K., Neubert P., Protzel P. A comparison of vector symbolic architectures. *arXiv:2001.11797*, 2020.
9. Harnad S. The symbol grounding problem. *arXiv:cs/9906002v1*, 1999.
10. Anderson J. R., Lebiere C. J. *The atomic components of thought*. Psychology Press, 2014. 504 p.
11. Sun R., Peterson T. *A hybrid agent architecture for reactive sequential decision making*. In: *Connectionist symbolic integration: From unified to hybrid approaches*. New York, Psychology Press, 1997. Pp. 113–138.
12. Avery E., Kelley T., Davani D. Using cognitive architectures to improve robot control: Integrating production systems, semantic networks, and sub-symbolic processing. *Proc. 15th Annual Conference on Behavioral Representation in Modeling and Simulation (BRIMS)*, 2006, pp. 190–198.
13. Kelley T., Avery E., Long L., Dimperio E. A hybrid symbolic and sub-symbolic intelligent system for mobile robots. *Proc. AIAA Infotech@Aerospace Conference*, 2009, p. 1976. doi:10.2514/6.2009-1976
14. Parvar J., Lowe D., Emsley M., Duff R., Akintoye A. Neural networks as a decision support system for the decision to bid process. *Proc. 16th Annual ARCOM Conference*, 2000, pp. 209–217.
15. Delen D., Sharda R. *Artificial neural networks in decision support systems*. In: *Handbook on decision support systems 1*. Springer, 2008. Pp. 557–580. doi:10.1007/978-3-540-48713-5\_26
16. Power D. J., Sharda R. Model-driven decision support systems: Concepts and research directions. *Decision Support Systems*, 2007, vol. 43, no. 3, pp. 1044–1061. doi:10.1016/j.dss.2005.05.030
17. Power D. J. *Decision support systems: Concepts and resources for managers*. Greenwood Publishing Group, 2002. 284 p.
18. Golmohammadi D. Neural network application for fuzzy multi-criteria decision making problems. *International Journal of Production Economics*, 2011, vol. 131, no. 2, pp. 490–504. doi:10.1016/j.ijpe.2011.01.015
19. Hwang C., Yoon K. *Multiple attribute decision making methods and applications: a state-of-the-art survey*. New York, Springer-Verlag Berlin Heidelberg, Springer Publications, 1981. 259 p. doi:10.1007/978-3-642-48318-9
20. Figueira J., Greco S., Ehrgott M. *Multiple criteria decision analysis: State of the art surveys*. New York, Springer-Verlag Berlin Heidelberg, Springer Publications, 2005. 1267 p. doi:10.1007/978-1-4939-3094-4
21. Demidovskij A. V., Babkin E. A. Developing a distributed linguistic decision making system. *Business Informatics*, 2019, vol. 13, no. 1, pp. 18–32. doi:10.17323/1998-0663.2019.1.18.32
22. Herrera F., Martinez L. A 2-tuple fuzzy linguistic representation model for computing with words. *IEEE Transactions on Fuzzy Systems*, 2000, vol. 8, no. 6, pp. 746–752. doi:10.1109/91.890332
23. Pogosyan K. S. The algorithm for generating optimal linguistic scale for group decision-making. *Proceedings of Voronezh State University: Systems Analysis and Information Technologies*, 2015, no. 2, pp. 97–104.
24. Wei G., Gao H. Pythagorean 2-tuple linguistic power aggregation operators in multiple attribute decision making. *Economic Research-Ekonomika Istraživanja*, 2020, vol. 33, no. 1, pp. 904–933. doi:10.1080/1331677X.2019.1670712
25. Rodriguez R. M., Martinez L., Herrera F. Hesitant fuzzy linguistic term sets for decision making. *IEEE Transactions on Fuzzy Systems*, 2011, vol. 20, no. 1, pp. 109–119. doi:10.1109/TFUZZ.2011.2170076
26. Liu P., Chen S.-M. Multiattribute group decision making based on intuitionistic 2-tuple linguistic information. *Information Sciences*, 2018, vol. 430, pp. 599–619. doi:10.1016/j.ins.2017.11.059
27. Wang J.-H., Hao J. A new version of 2-tuple fuzzy linguistic representation model for computing with words. *IEEE Transactions on Fuzzy Systems*, 2006, vol. 14, no. 3, pp. 435–445. doi:10.1109/TFUZZ.2006.876337
28. Demidovskij A., Babkin E. Designing a neural network primitive for conditional structural transformations. *Lecture Notes in Computer Science*, 2020, vol. 12412, pp. 117–133. doi:10.1007/978-3-030-59535-7\_9
29. Demidovskij A., Babkin E. Designing arithmetic neural primitive for sub-symbolic aggregation of linguistic assessments. *Journal of Physics: Conference Series*, 2020, vol. 1680, no. 1, article ID 012007. doi:10.1088/1742-6596/1680/1/012007
30. Graves A., Wayne G., Danihelka I. Neural Turing machines. *arXiv:1410.5401*, 2014.
31. Graves A., Wayne G., Reynolds M., Harley T., Danihelka I., Grabska-Barwińska A., Colmenarejo S. G., Grefenstette E., Ramalho T., Agapiou J., Badia A. P., Hermann K. M., Zwols Y., Ostrovski G., Cain A., King H., Summerfield C., Blunsom P., Kavukcuoglu K., Hassabis D. Hybrid computing using a neural network with dynamic external memory. *Nature*, 2016, no. 538, pp. 471–476. doi:10.1038/nature20101
32. Zaremba W., Sutskever I. Reinforcement learning neural Turing machines-revised. *arXiv:1505.00521*, 2015.
33. Chen X., Xu H., Zhang Y., Tang J., Cao Y., Qin Z., Zha H. Sequential recommendation with user memory networks. *Proc. Eleventh ACM International Conference on Web Search and Data Mining (WSDM '18)*, 2018, pp. 108–116. doi:10.1145/3159652.3159668
34. Grefenstette E., Hermann K. M., Suleyman M., Blunsom P. Learning to transduce with unbounded memory. *arXiv:1506.02516*, 2015.
35. Collier M., Beel J. Implementing neural Turing machines. *Lecture Notes in Computer Science*, 2018, vol. 11141, pp. 94–104. doi:10.1007/978-3-030-01424-7\_10
36. Castellini J. Learning numeracy: Binary arithmetic with neural Turing machines. *arXiv:1904.02478*, 2019.
37. Demidovskij A. Encoding and decoding of recursive structures in neural-symbolic systems. *Optical Memory and Neural Networks*, 2021, vol. 30, no. 1, pp. 37–50.