

О выражении лингвистических операторов в виде распределённого нейросетевого алгоритма

А. В. Демидовский¹, Э. А. Бабкин²

Национальный исследовательский университет;

Высшая школа экономики

Нижний Новгород, Россия

¹ademidovskij@hse.ru, ²eababkin@hse.ru

Аннотация. Рассматривается важная проблема интеграции символической и суб-символической парадигмы вычислений в системах массовой параллельной обработки данных, таких как Интернет Вещей. Данная проблема исследуется применительно к области лингвистического принятия решений. В работе выдвигается новый подход выражения лингвистических операторов агрегации оценок экспертов в терминах динамики нейронной сети. Данный подход предполагает два последовательных шага: выражение лингвистических операторов в виде манипуляций над специальным образом закодированными рекурсивными структурами и перевод этих действий на нейросетевой уровень. Теоретическим основанием работы является подход Тензорных Представлений, который предоставляет набор правил для построения нейронных сетей, не требующих обучения, результат работы которых эквивалентен работе символических алгоритмов. Обсуждается достижимость реализации предлагаемого подхода, демонстрируются основные строительные блоки и формируются направления дальнейших исследований.

Ключевые слова: искусственные нейронные сети; методы лингвистического принятия решений; тензорные представления

I. ВВЕДЕНИЕ

Многие годы символические и суб-символические вычисления воспринимались как соревнующиеся парадигмы, не имеющие ничего общего и конфликтующие в ключевых понятиях. В последние два десятилетия вопрос построения интегрированных решений вызывает большой интерес. Одной из причин для создания гибридных решений является желание объединить сильные стороны каждой из парадигм: высокую интерпретируемость символических моделей, массивный параллелизм и устойчивость коннективистских, или суб-символических, моделей. Искусственные Нейронные Сети (ИНС) часто воспринимаются как универсальные масштабируемые вычислительные единицы [1], [2], которые внедряются в широкий спектр промышленных задач.

В области создания параллельных моделей распределённых нейронных сетей важной научной задачей является синтез структуры распределённой нейронной сети, выполняющей на суб-символическом уровне

существенные интеллектуальные действия без предварительного обучения [3] в таких системах массовой параллельной обработки данных, как Интернет Вещей (IoT), где каждый компонент играет роль единичного нейрона или небольшой подсети [4].

Первым шагом на пути решения данной задачи является создание ИНС, способной производить точный результат для выбранной прикладной задачи. Многокритериальное принятие решений на основе нечётких лингвистических оценок, предоставляемых экспертами [5] может выступать примером такой прикладной задачи. При этом в рамках данной задачи в качестве символических структур могут выступать объекты разных типов [6]: деревья, формируемые в результате синтаксического и семантического разбора, морфемного анализа, альтернативы, критерии, оценки экспертов и т.д. Существует ряд работ, посвящённых применению суб-символической парадигмы к задачам принятия решений. В [3] представлены общие правила унификации и логического вывода. Однако, отсутствует формулирование арифметических операций в виде эквивалентных структурных операций. Существуют и работы посвященные, в том числе, агрегации нечётких оценок и построению нейронной сети, имитирующей алгоритм принятия решения [7]. Недостатком такого подхода является наличие обязательного шага по тренировке модели, что подразумевает наличие обучающей выборки. Отсутствие репрезентативных наборов данных в задачах принятия решений является открытой проблемой данной научной области. Вторым недостатком данной работы является необходимость осуществлять тренировку нейронной сети для каждой новой задачи. Исходя из этого, можно сделать вывод, что задача выражения процесса принятия решений на основе нечётких оценок на нейросетевом уровне является сложной, актуальной и открытой задачей.

Данная работа посвящена вопросу выражения процесса агрегации лингвистических оценок экспертов на нейросетевом уровне, не требуя при этом тренировки нейронной сети. В данной работе для моделирования лингвистических оценок используется модель 2-tuple и соответствующие операторы агрегации [8], а метод Тензорных Представлений (Tensor Product Representations) [9] – в качестве основополагающего инструмента для

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

кодирования структур в виде тензора и декодирования тензора в вид символической структуры. Цель данной работы заключается в исследовании применимости TPR к выражению лингвистических операторов и определении необходимых базовых блоков для создания такой системы. Гипотеза исследования следующая: если операции над 2-tuple представимы в виде структурных преобразований, таких как объединение деревьев и извлечение структурных элементов, тогда TPR может быть использовано для кодирования 2-tuple и их агрегации на суб-символическом уровне в виде распределённой нейронной сети.

Структура данной работы следующая. В Разделе II даётся краткий обзор методов лингвистического принятия решений (Linguistic Decision Making). Раздел III описывает подход Тензорных Представлений (TPR), правила кодирования и декодирования произвольных рекурсивных структур. Дизайн арифметики, построенной с помощью TPR, как основа для нейросетевой агрегации 2-tuple, представлен в Разделе IV. Выводы и направления дальнейших исследований представлены в Разделе V.

II. НЕЧЁТКИЕ ЛИНГВИСТИЧЕСКИЕ ПЕРЕМЕННЫЕ

Современные методы многокритериального принятия решений, которые учитывают нечёткие оценки экспертов, строятся на основе базовой модели 2-tuple [8]. Считается, что эта модель является более эффективной, чем другие способы сбора нечётких оценок экспертов [7]. Модель 2-tuple строится на основе операции символического перевода [4].

Определение 1. Структура 2-tuple представляет собой пару (s_i, α) где $s_i \in S = \{s_0, \dots, s_g\}$ – лингвистический термин (концепт), α – численное значение или символический перевод, отражающий расстояние до ближайшего концепта $s_i \in S = \{s_0, \dots, s_g\}$.

Определение 2. Правило перевода. Пусть $S = \{s_0, \dots, s_g\}$ – это лингвистическая шкала, где $g = \tau + 1$ обозначает уровень гранулярности S . Если $\beta \in [0, 1]$ – это результат символической агрегации, то существует способ построить по этому значению экземпляр 2-tuple:

$$\begin{aligned} \Delta_g &= [0, 1] \rightarrow S \times [-0.5, 0.5] \\ \Delta_g(\beta) &= (s_i, \alpha) \\ &= \begin{cases} s_i, i = \text{round}(\beta\tau) \\ \alpha = \beta\tau - i, \alpha \in [-0.5, 0.5] \end{cases} \end{aligned} \quad (1)$$

Определение 3. Правило обратного перевода. Пусть $S = \{s_0, \dots, s_g\}$ – это лингвистическая шкала, где $g = \tau + 1$ обозначает уровень гранулярности S . Пусть (s_i, α) – это экземпляр 2-tuple на лингвистической шкале S , где $\alpha \in [-0.5, 0.5]$. Тогда существует способ преобразовать экземпляр 2-tuple в число $\beta \in [0, 1]$:

$$\begin{aligned} \Delta_g^{-1} &= S \times [-0.5, 0.5] \rightarrow [0, 1] \\ \Delta_g^{-1}(s_i, \alpha) &= \frac{i + \alpha}{\tau} \end{aligned} \quad (2)$$

За последние годы активное развитие области лингвистического принятия решений привело к созданию многочисленных алгоритмов, или операторов, агрегации нечёткой информации, выраженной в форме лингвистических оценок: MTWA (Multigranularity 2-tuple Weighted Averaging), MHTWA (Multigranularity Hesitant 2-tuple Weighted Averaging) [10], P2TLWA (Pythagorean 2-tuple Linguistic Weighted Averaging) [11] и т.д. Один из самых простых и в то же время распространённых, MTWA, осуществляет поиск взвешенного среднего на множестве экземпляров 2-tuple.

Определение 4. Оператор MTWA. Пусть (b_i, α_i) – это экземпляр 2-tuple на лингвистической шкале S^{g_i} , $i = 1, 2, \dots, n$. Пусть $w = (w_1, w_2, \dots, w_n)$ – это заданный вектор весов, где w_i обозначает вес для (b_i, α_i) , $i = 1, 2, \dots, n$. Тогда оператор MTWA определяется как:

$$\begin{aligned} MTWA_{S^{g_k}}^w((b_1, \alpha_1), (b_2, \alpha_2), \dots, (b_n, \alpha_n)) \\ = \Delta_{g_k} \left(\sum_{j=1}^n w_j \Delta_{g_j}^{-1}(b_j, \alpha_j) \right) \end{aligned} \quad (3)$$

Однако, модель 2-tuple – это базовая модель, которая в последнее время используется для создания более гибких, с точки зрения качества отображения нечёткой информации, структур: Hesitant Fuzzy Linguistic Term Sets (HFLTTS) [12], Institutional 2-tuple [13], [14], гибридные модели [15] и т.д. Модель 2-tuple и её вариации стали активно внедряться в традиционные методы принятия решений, например в TOPSIS (Technique for Order of Preference by Similarity to Ideal Solution) [16], [17]. Более поздние работы [18], [19] учитывают не только лингвистические оценки, но и уверенность экспертов в даваемых ими оценках, что в целом повышает качество результатов.

III. РАБОТА СО СТРУКТУРАМИ С ПОМОЩЬЮ ТЕНЗОРНЫХ ПРЕДСТАВЛЕНИЙ

Как уже отмечалось ранее, представление знания в виде рекурсивных структур имеет важное прикладное значение. В то же время, структура – это символический элемент. Поэтому, она не может быть использована в качестве входных данных для нейронной сети. Ряд учёных под руководством Поля Смоленского предлагают рассматривать любую структуру как набор пар {филлер, роль} [9] и называют семейство подходов, объединённых этой идеей Тензорными Представлениями, а ключевой метод – Tensor Product Variable Binding позволяет выразить символические операции над рекурсивными структурами в виде операций над тензорами разной размерности [20].

Определение 4. Филлер – конкретный экземпляр данного структурного типа.

Определение 5. Роль – функция, которую филлер играет в структуре.

Определение 6. Тензорное произведение – это операция над тензором a с размерностью x и b с размерностью y , которая производит тензор z с

размерностью $x + y$, такой, что он состоит из попарных произведений элементов x и y .

Определение 7. Тензорное представление структуры. Пусть структура s – это набор пар филлеров $\{f_i\}$ и ролей $\{r_i\}$, тогда её тензорное представление выражается как:

$$\psi = \sum_i f_i \otimes r_i \quad (4)$$

Определение 8. Операция объединения $cons(p, q)$ – это действие, выполняемое над двумя структурами (деревьями) p и q , так что дерево p становится левым сыном нового дерева, а дерево q – правым сыном.

Операция $cons$ для двоичных деревьев обозначается как:

$$\begin{aligned} cons(p, q) &= p \otimes r_0 + p \otimes r_1 \\ cons_0(p) &\equiv cons(p, \emptyset) \\ cons_1(q) &\equiv cons(\emptyset, q) \end{aligned} \quad (5)$$

где r_0 и r_1 – это роли, \emptyset – это пустое дерево.

Доказано, что эта операция может быть выражена в матричной форме [9]:

$$cons(p, q) = W_{cons0}p + W_{cons1}q \quad (6)$$

Определение 9. Операция извлечения $ex_x(p)$ – это такая операция над структурой (деревом) s , что дерево s корнем в $-$ том по порядку потомке корня s становится независимым деревом.

Операция ex_x для двоичных деревьев обозначается как:

$$\begin{aligned} ex_0(s) &= p \\ ex_1(s) &= q \end{aligned} \quad (7)$$

где $s = cons(p, q)$.

Доказано, что эта операция может быть выражена в матричной форме [9]:

$$ex_x(s) = W_{ex_x} s \quad (8)$$

За последние пять лет наблюдается активное использование методов Тензорных Представлений в прикладных задачах. В частности, были предложены современные нейронные архитектуры для кодирования [21] и декодирования [22] структур, осуществление операций $cons$ и ex_x [23]. С другой стороны, идеи Тензорных Представлений применяются и в области глубокого обучения. Например, в задачах генерации подписи к изображениям [24] или при анализе способности нейронной сети вычленивать структурные зависимости во входных данных [25].

IV. ДИЗАЙН АРИФМЕТИКИ, ПОСТРОЕННОЙ НА TPR, КАК ОСНОВА ДЛЯ НЕЙРОСЕТЕВОЙ АГРЕГАЦИИ 2-TUPLE

Агрегация лингвистической информации, выраженной в форме экземпляров 2-tuple предполагает перевод этих оценок в численные представления (1). Поэтому, достаточно выразить базовые арифметические операции в виде структурных манипуляций и тогда станет возможным

агрегация лингвистических оценок на нейросетевом уровне. Для задания арифметики на основе TPR требуется определение аксиоматики, включая базовые значения и операции. Предлагается задавать арифметику по Пеано [26].

С точки зрения данной аксиоматики и основных принципов TPR мы предлагаем рассматривать любое число как дерево, состоящее из комбинации позиционных ролей: r_0 (левый потомок) and r_1 (правый потомок). Структура определяется рекурсивно и произвольной глубины. Как уже отмечалось, определены два примитивных значения. Пусть $\{0\}$ – это структура, состоящая исключительно из нулей (рис. 1a). В то же время, другое примитивное значение $\{1\}$ (рис. 1b) состоит из пары филлера и r_1 . Теперь с помощью предлагаемых базовых значений определим арифметику на основе TPR.

TPR-Inc. Оператор увеличения числа на единицу. Пусть $\{1\}$ обозначает структуру, представляющую 1. Процедура увеличения на единицу принимает структуру a на вход. В результате появляется структура s , представляющая входное число, увеличенное на единицу:

$$inc(a) = cons(a, \{1\}) \quad (9)$$

TPR-Dec. Оператор уменьшения числа на единицу. Пусть $\{0\}$ обозначает структуру, представляющую 0. Процедура уменьшения на единицу принимает структуру a на вход. В результате появляется структура s , представляющая входное число, уменьшенное на единицу:

$$dec(a) = \begin{cases} \{0\}, & \text{if } equal(a, \{0\}) \\ ex_0(a), & \text{otherwise} \end{cases} \quad (10)$$

TPR-Eq. Оператор проверки двух структур на равенство. Пусть a, b обозначают две структуры. Тогда оператор проверки на равенство определяется как:

$$equal(a, b) = \begin{cases} equal(dec(a), dec(b)), & \\ \text{if } a \neq \{0\}, b \neq \{0\} \\ \{0\}, & \text{if } a \neq \{0\}, b = \{0\} \\ \{0\}, & \text{if } a = \{0\}, b \neq \{0\} \\ \{1\}, & \text{if } a = \{0\}, b = \{0\} \end{cases} \quad (11)$$

TPR-Sum. Оператор суммы. Пусть a, b обозначают две структуры. Тогда оператор суммы определяется как:

$$plus(a, b) = \begin{cases} plus(dec(a), inc(b)), & \text{if } a \neq \{0\} \\ b, & \text{if } a = \{0\} \end{cases} \quad (12)$$

TPR-Mult. Оператор умножения. Пусть a, b – две структуры, тогда оператор умножения определяется как:

$$times(a, b) = \begin{cases} plus(times(dec(a), b), b), & \\ \text{if } a \neq \{0\}, b \neq \{0\} \\ \{0\}, & \\ \text{if } a \neq \{0\}, b = \{0\} \\ \{0\}, & \text{if } a = \{0\} \end{cases} \quad (13)$$

Примеры получения новых чисел из примитивных констант $\{0\}$ и $\{1\}$ представлены на рис. 1. Например, число $\{2\}$ представимо как результат выполнения

оператора *inc* над примитивным значением $\{1\}$, что на структурном уровне эквивалентно объединению структуры $\{1\}$ с ролью r_0 и структуры $\{1\}$ с ролью r_1 . Определение базовых операторов позволяет задавать более сложные операторы, такие как TPR-Sum (12) и TPR-Mult (13). Основные строительные блоки, такие как TPR-Inc (9) спроектированы в виде нейронной сети и доступны в проекте с открытым исходным кодом (<https://github.com/demid5111/ldss-tensor-structures>).

V. ЗАКЛЮЧЕНИЕ

В данной работе демонстрируется, что числа могут быть закодированы как рекурсивные структуры по правилам TPR, а целочисленная арифметика может быть выражена в виде операций в рамках эти правил. Это позволяет говорить о достижимости создания нейросетевого метода агрегации нечётких лингвистических оценок. Действительно, как было продемонстрировано в Разделе II, любая лингвистическая оценка может быть переведена в численный формат, а после агрегации, производится обратный перевод в вид 2-tuple. Далее, оператор MTWA (3) по определению состоит из операций суммы и произведения, каждая из которых представима в виде структурных манипуляций (12, 13). Наконец, существуют методы и инструменты, которые позволяют производить такие манипуляции на нейросетевом уровне [19], [20], [21], [22]. В силу того, что изначальная гипотеза подтвердилась, в качестве дальнейших исследований рассматривается построение системы, способной к агрегации лингвистических оценок в виде распределённого нейросетевого алгоритма.

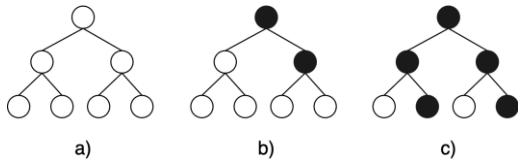


Рис. 1. Представление неотрицательных чисел в виде структур: а) $\{0\}$ б) $\{1\}$ в) $\{2\}$.

СПИСОК ЛИТЕРАТУРЫ

- [1] Rumelhart D. E., Hinton G. E., McClelland, J. L. A general framework for parallel distributed processing. *Parallel distributed processing: Explorations in the microstructure of cognition*, vol. 1, pp. 45-76.
- [2] Pinkas G. Reasoning, nonmonotonicity and learning in connectionist networks that capture propositional knowledge. *Artificial Intelligence*, vol. 77, i. 2, pp. 203-247.
- [3] Pinkas G., Lima P., Cohen S. Representing, binding, retrieving and unifying relational knowledge using pools of neural binders. *Biologically Inspired Cognitive Architectures*, vol. 6, pp. 87-95.
- [4] Yousefpour A., Nguyen B. Q., Devic S., Wang G., Kreidieh A., Lobel H., Jue J. P. Failout: Achieving Failure-Resilient Inference in Distributed Neural Networks. 2020. arXiv preprint arXiv:2002.07386.
- [5] Herrera F., Martínez L. A 2-tuple fuzzy linguistic representation model for computing with words. *IEEE Transactions on fuzzy systems*, vol. 8, i. 6, pp. 746-752.

- [6] Cho P.W., Goldrick M., Smolensky P. Incremental parsing in a continuous dynamical system: Sentence processing in Gradient Symbolic Computation. *Linguistics Vanguard*, vol. 3, i. 1.
- [7] Golmohammadi D. Neural network application for fuzzy multi-criteria decision making problems. *International Journal of Production Economics*, vol. 131, i. 2, pp. 490-504.
- [8] Zhang H., Wu Y., Gao J., Xu C. A Method for Multi-Criteria Group Decision Making with 2-Tuple Linguistic Information Based on Cloud Model. *Information*, vol. 8, i. 2.
- [9] Smolensky P. Tensor product variable binding and the representation of symbolic structures in connectionist systems. *Artificial intelligence*, vol. 46, i. 1-2, pp. 159-216.
- [10] Wei C., Liao H. A multigranularity linguistic group decision-making method based on hesitant 2-tuple sets. *International Journal of Intelligent Systems*, vol. 31, i. 6, pp. 612-634.
- [11] Wei G., Lu M., Alsaadi F.E., Hayat T., Alsaedi A. Pythagorean 2-tuple linguistic aggregation operators in multiple attribute decision making. *Journal of Intelligent & Fuzzy Systems*, vol. 33, i. 2, pp. 1129-1142.
- [12] Rodriguez R.M., Martinez L., Herrera F. Hesitant fuzzy linguistic term sets for decision making. *IEEE Transactions on fuzzy systems*, vol. 20, i. 1, pp. 109-119.
- [13] Liu P., Chen S. M. Multiattribute group decision making based on intuitionistic 2-tuple linguistic information. *Information Sciences*, vol. 430, pp. 599-619.
- [14] Wei G., Alsaadi F.E., Hayat, T., Alsaedi A. Picture 2-tuple linguistic aggregation operators in multiple attribute decision making. *Soft Computing*, vol. 22, i. 3, pp. 989-1002.
- [15] Wang J H., Hao J. A new version of 2-tuple fuzzy linguistic representation model for computing with words. *IEEE transactions on fuzzy systems*, vol. 14, i. 3, pp. 435-445.
- [16] Geng X., Gong X., Chu X. Component oriented remanufacturing decision-making for complex product using DEA and interval 2-tuple linguistic TOPSIS. *International Journal of Computational Intelligence Systems*, vol. 9, i. 5, pp. 984-1000.
- [17] Cheng P., Zhou B., Chen Z., Tan J. The TOPSIS method for decision making with 2-tuple linguistic intuitionistic fuzzy sets. In *2017 IEEE 2nd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*, 2017, pp. 1603-1607.
- [18] Wang J., Wei G., Lu M. TODIM method for multiple attribute group decision making under 2-tuple linguistic neutrosophic environment. *Symmetry*, vol. 10, i. 10, pp. 486.
- [19] Demidovskij A.V., Babkin E.A. Developing a distributed linguistic decision making system. *Biznes-informatika*, vol. 13, i. 1.
- [20] Legendre G., Miyata Y., Smolensky P. Distributed recursive structure processing. In *Advances in Neural Information Processing Systems*, 1991, pp. 591-597.
- [21] Demidovskij A. Implementation Aspects of Tensor Product Variable Binding in Connectionist Systems. In *Proceedings of SAI Intelligent Systems Conference*, 2019, pp. 97-110.
- [22] Demidovskij A., Babkin E., "Robust Encoding and Decoding of Recursive Structures in Connectionist Systems", unpublished.
- [23] Demidovskij A. V. Towards Automatic Manipulation of Arbitrary Structures in Connectivist Paradigm with Tensor Product Variable Binding. In *International Conference on Neuroinformatics*, 2019, pp. 375-383.
- [24] Huang Q., Smolensky P., He X., Deng L., Wu D. Tensor product generation networks for deep NLP modeling. arXiv preprint arXiv:1709.09118.
- [25] Soulos P., McCoy T., Linzen T., Smolensky P. Discovering the compositional structure of vector representations with role learning networks. arXiv preprint arXiv:1910.09113.
- [26] Van Heijenoort, J. From Frege to Gödel: a source book in mathematical logic, 1879-1932. Harvard University Press. 1967.