

УДК 519.7:007.52

В.Б. РЕПКА, В.Ю. БОРДЮГ

НЕЙРОСЕТЕВАЯ МОДЕЛЬ ГЕНЕРАЦИИ СТРУКТУРЫ ПРЕДЛОЖЕНИЯ ДЛЯ АВТОМАТИЧЕСКОГО ФОРМИРОВАНИЯ ТЕСТОВЫХ ЗАДАНИЙ В ДИСТАНЦИОННОМ ОБУЧЕНИИ

Введение.

На современном этапе развития образования все более весомое место среди различных способов получения знаний и закрепления изученного материала занимает дистанционное обучение – логичная и естественная форма обучения. Поэтому, вслед за возрастанием тенденций к переходу на электронные средства контроля знаний, возникла потребность в упрощении и автоматизации формирования материала для подачи и проверки знаний. Одним из наиболее распространенных способов проверки знаний является тестирование. Тесты применимы практически в любой отрасли науки от лингвистики до программирования. Удобнее всего формулировать вопрос по определению, поскольку в определении обычно не встречается неоднозначности и расплывчатости – оно формулируется четко и ясно. Кроме того, большинство тестов имеют ограничения во времени для ответов. Следовательно, они содержат общие вопросы, которые требуют мгновенного ответа и базируются исключительно на знании определений. Формулировка вопросов по определениям вручную не упрощает процесса подготовки материала, частично обесценивая электронный способ проверки знаний.

Постановка задачи.

Часто в русском языке можно проследить закономерность построения общих вопросов на основе определения, но встречаются случаи, когда существует несколько вариантов построения общих вопросов по одному и тому же определению. Здесь вступает в роль не только логический и лингвистический подходы к формированию вопроса, но и так называемый человеческий фактор [1]. Согласно мнению ученых, данный фактор во многом основывается на процессах, происходящих в нейронах человеческого мозга. С точки зрения классификации знаний, эти процессы и результаты их протекания следует отнести к нечетким знаниям. Одним из наиболее мощных и интересных аппаратов, работающих с нечеткими знаниями, являются нейронные сети. Возникает закономерный вопрос: а можно ли – и насколько успешно – обработать нечеткие знания данного примера с помощью искусственных нейронных сетей [2].

Если проанализировать многообразие форм определения, существующих в русском языке, то можно выделить некоторые шаблоны, под которые любое определение можно скорректировать. Одно из основных правил такой корректировки – построение определений, в которых присутствует словосочетание «– это», нет сложных оборотов, более двух подчиненных предложений и т.п. Согласно правилам русского языка, для построения общего вопроса нужно изменить порядок слов в предложении. Таким образом, имея определение, построенное по всем правилам языка, можно автоматизировать генерацию специального вопроса. Процесс генерации будет заключаться в определении порядка слов в предложении общего вопроса.

Задача, которая ставится перед нейронной сетью, может быть сведена к задачам классификации определенных конструкций определения по соответствующим позициям в вопросе. На входы нейронной сети подаются три или две конструкции, которые необходимо рассортировать правильным образом, не нарушив логическую структуру предложения. Такая постановка вопроса несколько отличается от обычной постановки задач классификации [3,4], когда на входы нейронной сети поступает один цельный объект с сово-

купностью признаков, по которым его необходимо отнести к определенному классу [3]. В рассматриваемой задаче существенен единственный признак – позиция конструкции в определении. Существует еще признак, отвечающий за смысловую нагрузку определения, но он не поддается описанию, поскольку и представляет собой результат сложнейшей организации нервной системы человека.

Не удивительно, что вслед за всем вышеизложенным встает вопрос: как сможет нейронная сеть уловить смысловую нагрузку предложения, представляющую собой нечеткие знания? Очевидно, что для решения описываемой задачи лучше подойдет сеть, которая будет обучаться с учителем [3]. При таком способе обучения, она получит входные данные и точные правильные ответы, не содержащие нечеткости, что позволит ей правильно настроить и скорректировать результаты своей работы, максимально приблизив их к верным. Среди многообразия архитектур нейронных сетей, решающих задачи классификации, для рассматриваемой задачи была выбрана сеть многослойный перцептрон как одна из гибко и наиболее результативно обучаемых с учителем сетей [3].

Анализ входных данных.

Входные данные для данной задачи представляют собой определение некоторого понятия или понятий на русском языке. Как синтаксическая конструкция, правильно сформулированное определение имеет следующие элементы:

- определяемую конструкцию;
- конструкцию «– это»;
- определяющую конструкцию;
- союзную конструкцию;
- дополнение определяющей конструкции.

Определяемая конструкция, с точки зрения языка, представляет собой слово или словосочетание, смысл или значение которого поясняется в определяющей конструкции. Наличие этого элемента обязательно для предложения, идентифицирующегося как определение. Конструкция «– это» является основным признаком определения.

Определяющая конструкция – это слово, чаще словосочетание, которое поясняет смысл определяемой конструкции и также необходимо в определении. Союзная конструкция представляет собой союз с дополняющими словами. Она связывает определяющую конструкцию с дополнением определяющей конструкции. Дополнение определяющей конструкции расширяет и уточняет поясняющие понятия. Дополнения может не быть в определении.

В соответствии с вышеописанным составом определения на рис. 1, 2 показан порядок расположения составляющих частей определения.

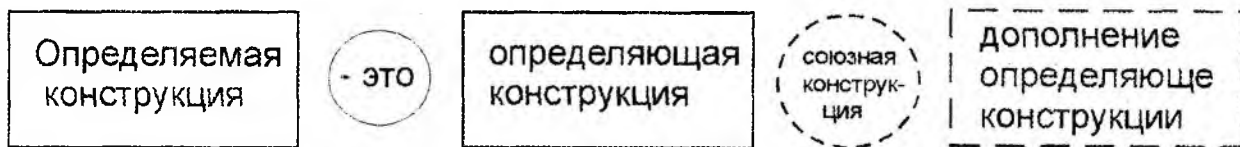


Рис. 1. Сложная структура определения.

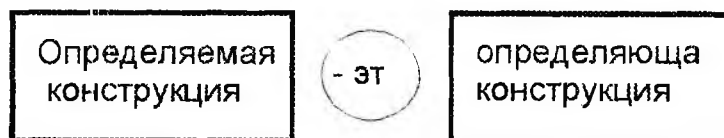


Рис. 2. Простая структура определения.

Таким образом, определяемая конструкция занимает первую позицию в определении. Определяющая конструкция – вторую, а дополнение определяющей конструкции (если оно присутствует) – последнее.

В зависимости от наличия дополнения определяющей конструкции, возможны два типа исходных данных-определений: сложное определение (дополнение определяющей конструкции присутствует, рис. 1) или простое определение (дополнение определяющей конструкции отсутствует, рис. 2).

По правилам языка общий вопрос формируется изменением порядка слов в определении и опусканием некоторых слов. Основное свойство определения – конструкция «– это» - не выполняет больше никаких функций и опускается из рассмотрения, как и союзная конструкция.

Определяемая конструкция всегда находится на последнем месте. Если в определении нет дополнения определяющей конструкции, то на первом месте всегда стоит определяющая конструкция. Такой вопрос будет иметь простую структуру (рис. 3а).

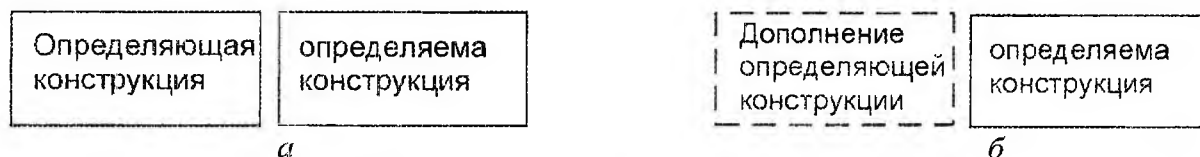


Рис. 3. Простые структуры вопроса.

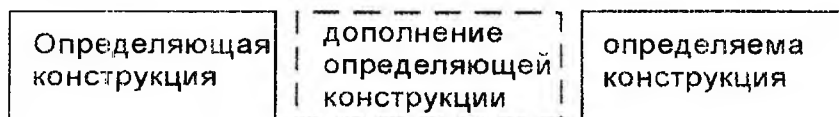


Рис. 4. Сложная структура вопроса.

Если определение сложное, то возможен как сложный, так и простой ответ. В данном случае результат зависит от содержания определения. Иногда определяющую конструкцию можно опустить. Тогда первое место в вопросе занимает дополнение и на выходной вопрос является простым (рис. 3б). Когда определяющую конструкцию опустить нельзя, то первое место занимает определяющая конструкция, второе – ее дополнение, а третье – определяемая конструкция. Такой вопрос имеет сложную конструкцию (рис. 4). Оба варианта допустимы и верны. Возможна также схема вопроса, приведенная на рис. 3а, при сложном определении. Очевидно, что простое определение не вызывает особых трудностей и вопросов, т. к. оно всегда порождает только простые общие вопросы. Трудности возникают при наличии сложного определения, т.к. результат зависит от смысловой нагрузки. Конечно, в этом случае можно обойтись сложным вопросом, но гораздо интереснее получить простой, повышая, таким образом, разнообразие вопросов.

Поскольку нейронные сети работают только с числовыми данными [4, 5], обозначим наличие соответствующей определенному месту в определении конструкции единицей, а отсутствие – нулем. Тогда общий вид определения в числовой форме представляется вектором (1, 1, 1) или (1, 1, 0). Определяемая конструкция вычленяется как часть определения сначала и до конструкции «– это», определяющая конструкция - от конструкции «– это» до союзной конструкции, дополнение определяющей конструкции – от союзной конструкции до конца предложения. Если дополнение определяющей конструкции отсутствует, то определяющая конструкция заканчивается концом определения. Для ускорения процесса обучения и корректной работы сети начальные значения должны быть нормированы. Причем важно не допустить перенасыщения нейронов входного слоя. Такая ситуация возможна, когда нейроны имеют большие значения [5]. В рассматриваемой постановке

задачи значения нейронов входного слоя заранее получились нормированными и не требуют дополнительной обработки.

В числовом виде выходной вопрос можно представить как вектор, значения которого выражают позицию соответствующих конструкций в вопросе. Индекс соответствует позиции конструкции в определении. Если данная конструкция отсутствует, координата имеет значение нуль. Поскольку значения функции активации для многослойного перцептрона лежат в интервале $[-1; 1]$ [5], то обозначим первую позицию в вопросе числом 1.0, вторую – 0.5, а третью – 0.1. Таким образом, задача сети заключается в определении позиции каждой конструкции в вопросе. В данном случае сеть должна определить число, выражающее позицию, из трех возможных или нуль, т.е. отнести определенную конструкцию определения к определенной позиции в вопросе. Получив на выходе для каждой из конструкций определенное число, обозначающее ее место в вопросе, необходимо вернуться от числовых данных к строковым и поставить строковые конструкции на вычисленные сетью места или опустить, если получено соответственное такому исходу значение.

Согласно проведенному выше анализу входных и выходных данных на входе и выходе сети имеется по три значения. Следовательно, количество входных и выходных нейронов многослойного перцептрона равняется трем. Поскольку количество входных и выходных нейронов совпадает, то для определения числа нейронов во внутренних слоях воспользуемся формулой:

$$n_{hidden} = \log_2 n_{in} = \log_2 3 \approx 2, \quad (1)$$

где n_{hidden} – количество нейронов в скрытом слое;

$n_{in}=3$ – количество нейронов во входном слое [3].

Разработка нейросетевой модели.

Поскольку задача сети заключается в перенастройке порядка следования входных данных, воспользуемся одним скрытым слоем. Топология многослойного перцептрона, решающего задачу генерации общих вопросов по известным определениям, представлена на рис. 5

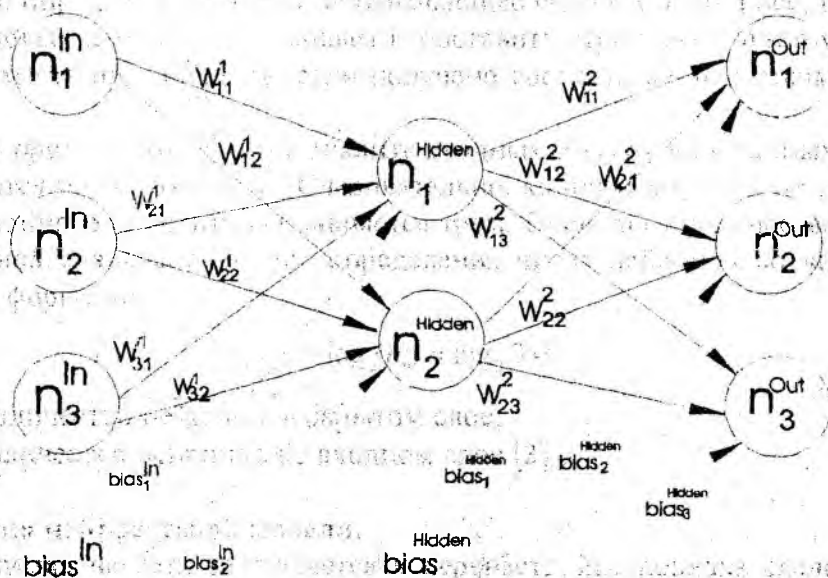


Рис. 5. Многослойные перцептрон для решения задачи генерации вопросов по известным определениям.

Одной из важных частей проектирования многослойного перцептрона является начальная разметка весов [6] – определение значений двух матриц:

w_{ij}^1 – матрица весов между входным и внутренним слоем, где $i = 1..n_{in}$, а $j = 1..n_{hidden}$,

w_{ij}^2 – матрица весов между входным и внутренним слоем, где $j = 1..n_{out}$, а $i = 1..n_{hidden}$.

Значения весов многослойного перцептрона должны задаваться случайным образом, но подчиняться нормальному закону распределения $N(0, 1)$ и лежать в диапазоне $[-1, 1]$ [6, 7]. Поскольку стандартный генератор случайных чисел генерирует значения по равномерному закону распределения, то необходимо провести преобразования по формулам (2)-(3).

$$r = (2 * random1 - 1)^2 + (2 * random2 - 1)^2, \quad (2)$$

$$sq = \sqrt{(2 * \ln r) / r}, \quad (3)$$

$$w_{ij}^k = 0.0 + 1.0 * (2 * random1 - 1) * sq, \quad (4)$$

где $random1$ и $random2$ – последовательно сгенерированные случайные числа.

Согласно (2)-(4) соответствующим синапсам (рис. 5) были назначены веса следующим образом:

$$w_{ij}^1 = \begin{pmatrix} -0.2903 & 0.1513 \\ 0.2086 & 0.5104 \\ 0.8098 & -0.1286 \end{pmatrix}, \quad w_{ij}^2 = \begin{pmatrix} -0.6984 & -0.6300 & 0.5250 \\ -0.2368 & 0.1852 & 0.1109 \end{pmatrix}$$

При проектировании многослойного перцептрона важно правильно задать смещения $bias^k$ и их веса, что значительно упростит и ускорит процесс обучения сети [7]. Поскольку, согласно проведенному анализу входных и выходных данных, значение первого выходного нейрона определяется всегда однозначно, $n_i^{out} = 0.1$, а второго и третьего может быть равным 1.0; 0.5 или 0.0, то вектор для смещения $bias^{hidden}$ зададим следующий: $(-0.5, 1.0, 1.0)$. Для смещения $bias^{out}$ выберем такой же вектор. Назначим смещение $bias^{hidden} = bias^{out} = 1.5$.

Линейная составляющая S_i^{hidden} для нейронов ассоциативного слоя вычисляется по формуле (5):

$$S_i^{hidden} = bias^{hidden} * bias_i^{hidden} + \sum_{j=1}^{n_{in}} w_{ji}^{hidden} * n_j^{in}, \quad (5)$$

где $bias_i^{hidden}$ – i -й синапс для смещения $bias^{hidden}$,

n_j^{in} – значение j -го нейрона входного слоя.

Линейная составляющая S_i^{out} для нейронов ассоциативного слоя вычисляется по формуле:

$$S_i^{out} = bias^{out} * bias_i^{out} + \sum_{j=1}^{n_{hidden}} w_{ij}^{out} * n_j^{hidden}, \quad (6)$$

где n_j^{hidden} – значение j -го нейрона ассоциативного слоя.

В роли функции активации для многослойного перцептрона выступает сигмоида [3, 7]

$$n_i^k = 1 / (1 + e^{-\alpha * S_i^k}), \quad (7)$$

где n_i^k – значение i -го нейрона в слое k ;

α – коэффициент, определяющий степень приближения графика сигмоиды к единице;

S_i^k – значение линейной составляющей i -го нейрона в слое k .

Для описываемого многослойного перцептрона было принято значение $\alpha = -0.5$.

Полученные выходные значения n_i^{out} нуждаются в проверке на совпадение с ожидаемыми d_i^{out} . При этом возможен вариант, когда

$$n_i^{out} = d_i^{out} \quad (8)$$

Это означает, что сеть обучена правильно и готова к тестированию [7]. Если условие (8) не выполнено, тогда вычисляется ошибка сети:

$$errorNet_{step} = \sum_{i=1}^{n_{out}} (d_i^{out} - n_i^{out}), \quad (9)$$

где $errorNet_{step}$ – это ошибка сети на шаге обучения $step$.

Если $errorNet_{step} = errorNet_{step-1}$, то сеть считается обученной [4, 7]. Но фактически, из-за использования чисел с двойной точностью, такое равенство никогда не будет достигнуто, поэтому ограничимся точностью вычислений до четырех знаков после запятой. Тогда условие окончания обучения сети выражается формулой (10):

$$errorNet_{step} - errorNet_{step-1} \leq 0.00006 \quad (10)$$

Таким образом, неточность составляет 0.006%.

Если ни одно из условий окончания обучения сети не выполнено, то сеть итеративно перенастраивает веса по алгоритму «back propagation». Веса перенастраиваются с последнего слоя out по формуле

$$w_{ij}^k(step + 1) = w_{ij}^k(step) + \eta * \delta_i^k * n_i^{k-1}, \quad (11)$$

где $w_{ij}^k(step + 1)$ – значения соответствующих весов между i -м нейроном слоя k и j -м нейроном слоя $k-1$ для следующего шага обучения;

$w_{ij}^k(step)$ – текущее значение весов между i -м нейроном слоя k и j -м нейроном слоя $k-1$;

η – параметр скорости обучения, который определяется итеративно по формуле (14),

δ_i^k – ошибка слоя k , которая вычисляется по формуле (12) для последнего слоя out, и по формуле (13) для скрытого слоя hidden;

n_i^{k-1} – значение нейрона предыдущего слоя: hidden или in.

$$\delta_i^{out} = (d_i^{out} - n_i^{out}) * n_i^{out} * (1 - n_i^{out}), \quad (12)$$

$$\delta_i^{hidden} = n_i^{hidden} * (1 - n_i^{hidden}) * \sum_{j=1}^{n_{out}} \delta_j^{out} * n_j^{out}, \quad (13)$$

$$\eta = 0.9 - (1/10^{step}) * step. \quad (14)$$

В формуле (12) организовано постепенное уменьшение значения параметра скорости обучения в зависимости от шага обучения.

Когда обучение заканчивается, необходимо проверить качество обучения. Для этого при каждой итерации обучения производится определение вычисления средней относительной ошибки по формуле:

$$\sigma = 1/(step * n_{out}) * \sum_{step=1}^{\infty} \sum_j^{n_{out}} ((|d_j^{step} - n_j^{out;step}| + 1) / (|d_j^{step}| + 1) - 1) * 100\%. \quad (15)$$

Качество обучения составляет $100\% - \sigma$ [8].

Разработанная модель нейронной сети стала основой программной системы, написанной на языке Java, которая осуществляет автоматизацию генерации общих вопросов по известным определениям.

На начальном этапе работы сеть принимает веса, полученные единожды и распределенные по нормальному закону распределения с параметрами 0 и 1. Далее происходит обучение сети и настройка весов на обучающей выборке из файла обучающей выборки и выходных данных из выходного файла. Содержания файла можно изменить, предложив сети новую обучающую выборку. Важно правильно разметить определения в файле: каждое определение начинать с новой строки, в конце ставить точку, перевести каретку и вставить разделитель «\n-----\n». Эталонные значения выхода следует просто задать подряд в столбец.

Для корректной настройки весов нейронов и смещений в описываемом случае использовалась выборка из 50 определений, взятых из предметной области «Прикладной анализ случайных данных». В ходе нескольких запусков сети многослойного перцептрона, описанные выше значения смещений $bias^k$ и $bias_j^k$ были определены экспериментально. На сороковом примере из обучающей выборки сеть обучилась. Средняя относительная ошибка составила 3,4%, а качество обучения многослойного перцептрона – 96,6%. Таким образом, делаем вывод, что данная выборка оптимальна для обучения и сеть наиболее обучена в таком состоянии [8].

Для тестового примера определения могут набираться вручную, или загружаться из тестового файла. На этапе первого тестирования в виде тестовой выборки использовались оставшиеся незадействованные в обучении определения из обучающей выборки.

Выводы.

В результате проведенного эксперимента по использованию нейронной сети для генерации общих вопросов по известным определениям и анализа результатов обучения рассматриваемой сети многослойного перцептрона был получен высокий процент качества обучения. Следовательно, аппарат нейронных сетей подходит для использования в работе с нечеткими знаниями, связанными с естественным языком, в области дистанционного обучения. Кроме того, поскольку в нашей стране система дистанционного обучения находится на стадии эксперимента [9], предполагается продолжить разработки и исследования в данной области применения нейронных сетей, расширить спектр генерируемых вопросов вопросами с возможностью ввода ответов, специальных вопросов.

Список литературы: 1. Попов Э.В. Общение с ЭВМ на естественном языке. М.: Наука, 1982. 360 с. 2. Смирнов Ю.М., Андреев А.М., Березкин Д.В., Брик А.В. Об одном способе построения синтаксического анализатора текстов на естественном языке // Изв. вузов. Приборостроение, 1997. Т. 40, № 5. С. 34—42. 3. Уоссерман Ф. Нейрокомпьютерная техника. М. Мир, 1992. 240 с. 4. Руденко О.Г., Бодянский Е.В. Основы теории искусственных нейронных сетей. Харьков: ТЕЛТЕХ, 2002. 317 с. 5. Лесная Н. С., Репка В. Б., Шатовская Т. Б. Интеллектуальный анализ данных: Научное пособие. Харьков: ХНУРЭ, 2003. 112 с. 6. Осовский С. Нейронные сети для обработки информации. М.: ФиС, 2002. 344 с. 7. Круглов В.В., Борисов В.В. Искусственные нейронные сети. Теория и практика. М.: Горячая линия – Телеком, 2001. 382 с. 8. Горбань А. Н., Россиев Д. Нейронные сети на персональном компьютере. Новосибирск: Наука, 1996. 276 с. 9. Черников А. Очевидные успехи дистанционного образования // Компьютерное обозрение. Тема недели. 2003. № 31-32. С. 40-41.

Поступила в редакцию 10.09.2003