

Трансфертное обучение и доменная адаптация на основе моделирования социально-экономических систем

О.Д. Казаков 

E-mail: kod8383@mail.ru

О.В. Михеенко 

E-mail: miheenkoov@mail.ru

Брянский государственный инженерно-технологический университет
Адрес: 241037, г. Брянск, пр. Станке Димитрова, д. 3

Аннотация

В статье рассматриваются вопросы применения методов трансфертного обучения (transfer learning) и доменной адаптации (domain adaptation) в рекуррентной нейронной сети, построенной по архитектуре долгой краткосрочной памяти (long short-term memory, LSTM), для повышения эффективности управленческих решений и государственной экономической политики. Обзор существующих в данной области подходов позволяет сделать вывод о необходимости решения ряда практических вопросов повышения качества предиктивной аналитики для задач разработки прогнозов развития социально-экономических систем. В частности, в контексте применения алгоритмов машинного обучения одной из проблем представляется ограниченное количество размеченных данных. Авторами реализовано обучение исходной рекуррентной нейронной сети на синтетических данных, полученных в результате имитационного моделирования, с последующим трансфертным обучением и доменной адаптацией. Для реализации этой цели на основе комбинирования нотаций системной динамики с агентным моделированием в системе AnyLogic разработана имитационная модель, позволяющая исследовать влияние совокупности факторов на ключевые параметры эффективности социально-экономической системы. Обучение исходной LSTM осуществлялось с помощью открытой программной библиотеки для машинного обучения TensorFlow. Предложенный подход позволяет расширить возможности комплексного применения методов имитационного моделирования для построения нейронной сети в целях обоснования параметров развития социально-экономической системы и позволяет получить информацию о ее перспективном состоянии.

Ключевые слова: трансфертное обучение; доменная адаптация; имитационное моделирование; системы поддержки принятия решений; социально-экономическое развитие регионов.

Цитирование: Казаков О.Д., Михеенко О.В. Трансфертное обучение и доменная адаптация на основе моделирования социально-экономических систем // Бизнес-информатика. 2020. Т. 14. № 2. С. 7–20.

DOI: [10.17323/2587-814X.2020.2.7.20](https://doi.org/10.17323/2587-814X.2020.2.7.20)

Введение

Управление развитием социальных-экономических систем в своей основе опирается на документы, содержащие плановые значения индикаторов по той или иной тематике (стратегия, концепция, прогноз и т.п.). На сегодняшний день управление регионом осуществляется посредством мониторинга путем корректировки плановых значений в соответствии с фактически достигнутыми [1]. Это означает, что в основе будущего развития по большей части заложены показатели прошлых периодов, полученные с существенным опозданием, если принимать во внимание реальную ситуацию с публикацией официальных статистических данных. В связи с этим разработка инструментария для обоснования значений прогнозных экономических параметров, позволяющего с высокой степенью надежности достигать плановых контрольных цифр, является важной научной задачей. Эта задача в конечном счете выступает в качестве объективного условия реализации эффективной экономической политики.

Основу всей совокупности методов социально-экономического прогнозирования традиционно составляют статистические методы, применяемые для построения адекватных моделей временных рядов [2, 3]. Среди наиболее распространенных методов анализа временных рядов можно выделить следующие [4]: регрессионные модели прогнозирования (множественная и нелинейная регрессия), модели экспоненциального сглаживания (ES), модель по выборке максимального подобия (MMSP), модель на цепях Маркова (Markov chains), модель на классификационно-регрессионных деревьях (CART), модель на основе генетического алгоритма (GA), модель на опорных векторах (SVM). Широчайшим и наиболее применимым из классов моделей являются авторегрессионные модели прогнозирования (ARIMAX, GARCH, ARDLM).

В последнее время доказывают свою эффективность методы глубокого машинного обучения, метрики качества которых значительно лучше по сравнению с классическими методами. Однако применение подобных моделей требует наличия огромного объема размеченных данных, получить которые в реальных условиях не всегда представляется возможным. В то же время при прогнозировании большинства показателей, характеризующих социально-экономические системы и процессы, используются статистические данные за один десяток лет и, в лучшем случае, в разрезе месяца. Иначе говоря,

на входе имеется всего сотня размеченных записей. Проблема тем или иным образом могла бы решиться в случае применения обучения без учителя, что, к сожалению, на данном этапе развития серийных вычислительных систем не может быть реализовано на практике.

Для решения данной проблемы предлагается использовать рекуррентную нейронную сеть, построенную по архитектуре долгой краткосрочной памяти (long short-term memory, LSTM) и обученную на синтетических данных, полученных в результате имитационного моделирования с последующим трансфертным обучением (transfer learning) и доменной адаптацией (domain adaption). Это позволит, имея реальную статистику за несколько десятков лет, с высокой степенью точности прогнозировать значения экономических параметров с учетом современных векторов развития. Системы поддержки принятия решений, построенные на данных алгоритмах, позволяют наиболее точно обосновывать экономические планы и прогнозы развития территорий и обеспечивать достижение стратегических ориентиров развития.

1. Методы

1.1. Трансфертное обучение и доменная адаптация в LSTM

Основная идея трансфертного обучения состоит в решении поставленной проблемы на основе «готовых данных», полученных в результате решения аналогичных задач. Это означает, что сначала можно обучить нейросеть на большом объеме данных, а впоследствии дообучить ее на конкретном целевом наборе. В этой связи выделяют два основных преимущества использования трансфертного обучения [5]:

- ◆ значительное снижение времени и затрат в контексте использования соответствующей инфраструктуры для обучения, за счет обучения только определенной части конечной модели;
- ◆ повышение эффективности конечной модели за счет использования моделей, обученных на доступных данных.

Результаты данного исследования тесно связаны со вторым из перечисленных преимуществ, поскольку в предиктивной аналитике социально-экономических систем это является определяющим фактором.

В качестве доступных данных использовались синтетические данные, полученные в результате

имитационного моделирования. Имитационное моделирование представляет собой экспериментальный способ изучения реальности с помощью компьютерной модели [6]. В имитационных моделях реальные экономические процессы описаны так, как если бы они происходили в действительности [7]. Таким образом, имитационные модели могут применяться для исследования реальных социально-экономических систем с условием, что экономические объекты и процессы заменяются совокупностью математических зависимостей, которые определяют в какое состояние перейдет система из изначально заданного [8].

Веса из обученной на синтетических данных модели, полученные в результате имитационного моделирования, переносятся на новую модель. Для этого авторами использовалась открытая программная библиотека для машинного обучения TensorFlow.

Методы трансфертного обучения и доменной адаптации, как правило, зависят от алгоритмов машинного обучения, используемых для решения поставленных задач [9]. Одним из наиболее эффективных инструментов предиктивной аналитики социально-экономических систем являются рекуррентные нейронные сети с долгой краткосрочной памятью (LSTM-сети). В частности, модели, построенные по архитектуре LSTM, являются очень эффективными для прогнозирования временного ряда – одной из самых распространенных задач в управлении социально-экономическими системами [10]. Следует отметить, что эта эффективность не снижается при прогнозировании нескольких шагов.

Базовая архитектура рекуррентной сети, разработанная еще в 1980-е годы, строится из узлов, каждый из которых соединен со всеми другими узлами.

Для обучения с учителем с дискретным временем на входные узлы при каждом очередном шаге подаются данные. При этом прочие узлы (выходные и скрытые) завершают свою активацию и выходные сигналы готовятся для передачи нейронам следующего уровня [11]. Таким образом, рекуррентная сеть с долгосрочной памятью позволяет использовать полученную в прошлом информацию для решения текущих задач. В частности, она дает возможность прогнозирования значений временного ряда, поскольку не использует функцию активации внутри своих рекуррентных компонентов, и хранимое значение не размывается во времени (рисунки 1) [12, 13].

Модуль LSTM имеет пять основных компонентов, которые позволяют ему моделировать как долгосрочные, так и краткосрочные данные [13]:

$$\begin{cases} f_t = \sigma(W_f[h_{t-1}, x_t] + b_f) \\ i_t = \sigma(W_i[h_{t-1}, x_t] + b_i) \\ o_t = \sigma(W_o[h_{t-1}, x_t] + b_o) \\ \tilde{c}_t = \tanh(W_c[h_{t-1}, x_t] + b_c) \\ c_t = f_t \odot c_{t-1} + i_t \odot c_t \\ h_t = o_t \odot \tanh(c_t) \end{cases}, \quad (1)$$

где c_t – «состояние ячейки», представляющее ее внутреннюю память, которая хранит как кратковременную, так и долговременную информацию;

h_t – «скрытое состояние»: такая информация о состоянии вывода, которая рассчитана по текущему входу, предыдущему скрытому состоянию и текущему входу ячейки, которые будут использоваться для прогнозирования того или иного временного ряда. Скрытое состояние может принять решение извлечь кратковременную или долгосрочную, либо оба типа информации из сохраненной в c_t ;

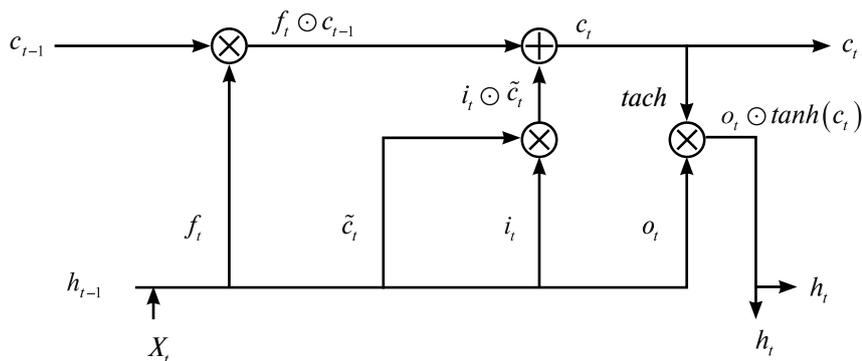


Рис. 1. Архитектура LSTM [12, 13]

```

▶ my_new_model = tf.keras.models.load_model('Kazakov_LSTM.h5')

▶ for i in zip(my_new_model.layers[0].trainable_weights,
              my_new_model.layers[0].get_weights()):
    print('Parametr %s:\n%s' % (i[0], i[1]))
    
```

Рис. 2. Листинг «Вывод параметров LSTM»

i_t – «входные ворота»: определяют объем информации поступающей из текущего ввода в c_t ;

f_t – «переходные ворота»: определяют объем информации, перетекающей из текущего и предыдущего c_{t-1} вводов в текущий c_t ;

o_t – «выходные ворота»: определяют объем информации, попадающей из текущего c_t в скрытое состояние.

Допустим, имеется хорошо работающая модель для предсказательной аналитики временных рядов “Kazakov_LSTM.h5” (процесс ее обучения представлен в следующей разделе статьи). Тогда для просмотра параметров данной модели можно воспользоваться следующими инструкциями (рисунок 2).

Таким образом, получим следующий вывод:

```

Parametr lstm_3_W_i:
[[ 0.00075, ...]]
Parametr lstm_3_U_i:
[[ 1.090001, ...]]
Parametr lstm_3_b_i:
[ 0., ...]
Parametr lstm_3_V_c:
[[-1.17770085, ...]]
    
```

где W -матрицы – матрицы, которые преобразуют входные данные;

U -матрицы – матрицы, которые преобразуют предыдущее скрытое состояние в другое внутреннее значение;

b -векторы – смещение для каждого блока;

V – вектор, определяющий значения, которые следует выводить из нового внутреннего состояния.

Понятие доменной адаптации тесно связано с трансфертным обучением. Суть этой адаптации заключается в обучении модели на данных из домена-источника так, чтобы она показывала сравнимое качество на целевом домене [14]. Домен-источник может представлять собой синтетические данные, которые можно просто сгенерировать в результате прогона соответствующей имитационной модели, а целевой домен – это временной ряд, отражающий

динамику тех или иных ключевых показателей социально-экономической системы. Тогда задача доменной адаптации заключается в тренировке модели на синтетических данных, которая будет хорошо работать с реальными объектами.

Этап адаптации домена сводится к замораживанию весов в модели “Kazakov_LSTM.h5” в их предварительно подготовленном состоянии. Веса уровня адаптации домена обучаются на целевом наборе данных. Для этой цели в модель после LSTM добавим полносвязанные (dense) слои.

1.2. Системно-динамическое моделирование индикаторов инновационного развития социально-экономических систем

В целях формирования набора данных в рамках домена-источника построим системно-динамическую модель, позволяющую определить параметры социально-экономической системы, в частности, произвести оценку значений индикаторов инновационного развития регионов. Основным документом, задающим стратегические ориентиры государственной политики в сфере инновационного развития в целях противостояния современным глобальным вызовам и угрозам, является Стратегия инновационного развития Российской Федерации на период до 2020 года [15]. Стратегия определяет долгосрочные приоритеты развития всех субъектов инновационной деятельности, а также устанавливает ряд целевых индикаторов, которые, в соответствии с установкой Правительства страны, должны учитываться при разработке концепций и программ социально-экономического развития России и ее регионов.

Стратегией определены значения целевых индикаторов на 2020 год. При этом 2010 год закреплен как базовый, а 2013 и 2016 годы являются промежуточными контрольными точками. Анализ фактических значений большинства целевых индикаторов за 2016 год выявил общую тенденцию отставания от запланированного уровня.

Поскольку статические службы готовят аналитические данные за отчетный период с существенным временным лагом, серьезной проблемой является то обстоятельство, что государственные власти, ответственные за реализацию Стратегии, пытаются выработать управленческие решения, ориентируясь на неактуальные результаты деятельности. Назвать такой процесс эффективным управлением крайне сложно.

На наш взгляд, наиболее действенным подходом будет обратное движение, когда значение конкретного целевого показателя на определенную дату дифференцируется по субъектам федерации и доводится до региональных властей заблаговременно, в виде рекомендуемых прогнозных значений. В этом случае государственные службы на местах станут непосредственными участниками процесса реализации национальных стратегических инициатив, в том числе, с учетом определенной ответственности за недостижение целевых показателей. Кроме того, появится возможность управлять на основе актуальной карты, отражающей инновационное развитие по долгосрочным задачам Стратегии в разрезе регионов.

Основными компонентами системной модели, позволяющей определить инновационное развитие региона в соответствии с государственной стратегией, являются стратегические задачи по ключевым направлениям. Связь между итоговым показателем инновационного развития и данными компонентами (субиндексами) может быть описана следующим образом:

$$I = \sum_{j=1}^m w_j \cdot I_j, \quad (3)$$

где I_j – значение j -го субиндекса;

m – число субиндексов;

w_j – коэффициент весомости j -го субиндекса.

Субиндексы представляют собой сводные показатели, отражающие формирование первичных индикаторов в рамках решения конкретной стратегической задачи по приоритетным направлениям инновационной деятельности. Число первичных индикаторов по направлениям варьируется от 2 до 12, при этом каждый из них может быть рассмотрен как самостоятельная сложная системно-динамическая модель. Рассмотрим модель формирования частного индикатора первого порядка «Коэффициент изобретательской активности», который определяется в рамках приоритетной стратегической задачи «Инно-

вационный бизнес». Модели остальных индикаторов могут быть сформированы аналогичным образом и не представлены в рамках данной статьи ввиду значительного масштаба проведенного исследования.

Системно-динамическая модель уровня изобретательской активности представлена на рисунке 3. Модель основана на определении отношения числа патентных заявок, поданных отечественными изобретателями, к общей численности населения. Количество разработанных и поданных патентных заявок зависит от числа организаций, осуществляющих деятельность в области НИОКР, численности персонала, занимающегося исследованиями и разработками, а также объема внутренних затрат организаций на НИОКР.

Соответствующая математическая модель может быть представлена следующим образом:

$$\begin{aligned} \frac{d(\text{Изобретения_разработано})}{dt} &= - \text{patent} \\ \frac{d(\text{Полезные_модели_разработано})}{dt} &= - \text{Полезные_модели_подано} \\ \frac{d(\text{patent})}{dt} &= \text{Изобретения_подано} + \\ &+ \text{Полезные_модели_подано} \\ \frac{d(\text{population})}{dt} &= \text{Рождение} - \text{Смерть} + \\ &+ \text{Миграция} - \text{Эмиграция} \end{aligned} \quad (4)$$

$$\begin{aligned} \text{Изобретения (Полезные_модели)_разработано} &= \\ &= \text{corp_research} \times \text{person_research} \\ &\times \text{Внутренние затраты НИОКР} \end{aligned}$$

$$\begin{aligned} \text{Смерть} &= \text{Смерть_трудосп_возраст} + \\ &+ \text{Смерть_младенческая} + \text{прочее} \end{aligned}$$

$$\begin{aligned} \text{Миграция} &= \text{Беженцы} + \text{Временное_убежище} + \\ &+ \text{Вынужденны_переселенцы} + \text{Прибывшие} \end{aligned}$$

$$\text{coeff_inv_activity} = \frac{\text{patent}}{\text{population}} \times 10\,000.$$

При первом рассмотрении возникает вопрос целесообразности применения имитационного моделирования для оценки уровня изобретательской активности, поскольку каждая из составляющих его компонент поддается прогнозированию (например, в рамках моделей ARIMA или GARCH, хорошо за-

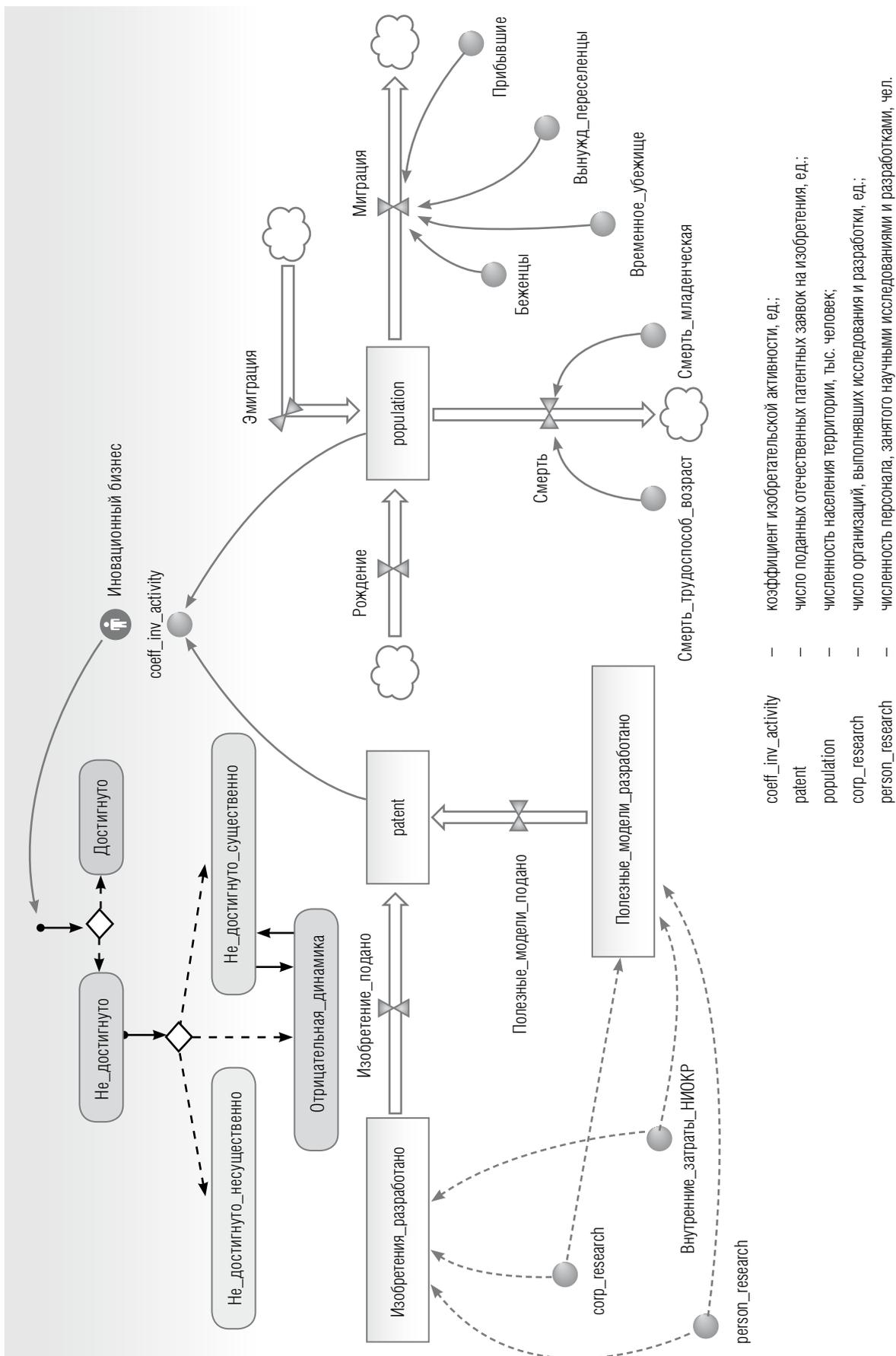


Рис. 3. Системно-динамическая модель уровня изобретательской активности

рекомендовавших себя в сфере прогнозирования демографических и социально-экономических показателей). В действительности зависимость изобретательской активности от многих показателей является стохастической, к тому же на практике по большинству их них собрать достаточный по объему набор значений не представляется возможным. Поэтому имитационное моделирование в данном контексте рассматривается как способ построения модели социально-экономической системы, описывающей сложное поведение объектов и процессов, связанных с управлением инновациями на уровне региона. Данную модель возможно реализовать любое число раз. В этом случае результаты будут обусловлены случайным характером процессов [16]. Используя такие результаты, можно получить устойчивую синтетическую статистику уровня изобретательской активности, которая впоследствии используется для обучения нейронной сети.

2. Эксперимент

2.1. Генерация синтетических данных с помощью системно-динамической модели

Имитационное моделирование было реализовано посредством набора математических инструментальных средств и специального программного обеспечения AnyLogic, позволивших провести целенаправленное моделирование в режиме «имитации» структуры исследуемого индикатора, а также оптимизацию некоторых его параметров [17]. В соответствии с результатами исследования, проведенного группой ученых Казанского технического университета [18], достоверность системы AnyLogic признана удовлетворительной, а в рейтинге аналогичного программного обеспечения данная система входит в тройку лидеров.

Конфигурационные настройки модели, графически описывающей поставленную пользователем проблему в терминах языка AnyLogic, задаются с помощью экспериментов. Дискретно-событийное моделирование реализует возможность аппроксимации реальных процессов дискретными событиями, рассматривающими наиболее важные моменты жизни моделируемой системы [19].

В системе AnyLogic был проведен эксперимент «Варьирование параметров», суть которого состояла в многократном запуске построенной имитационной модели. Для эксперимента была определена доверительная вероятность 0,95 и точность 0,01. Число прогонов модели, рассчитанное по функции Лапласа, составило 9604 [20]. Варьируя разные значения параметров, модель выдавала значение метки в пределах от 1,4725 до 2,1105. Математическое ожидание составило 1,8114, а рассеивание значений относительно математического ожидания – 0,1539, что является допустимым и позволяет сделать вывод об успешной валидации предложенной имитационной модели. Результаты эксперимента представлены в таблице 1.

В таблице 2 представлены синтетические данные, полученные в результате эксперимента «Варьирование параметров» в системе AnyLogic и используемые для обучения первичной нейронной сети.

Исходный набор данных содержит пять функций, изменение во времени которых представлено на рисунке 4.

На рисунке видно, что все временные ряды обладают свойством сезонности, но в явном виде этот фактор мы не будем учитывать при дальнейшем обучении исходной сети.

Таблица 1.

Результаты эксперимента «Варьирование параметров» системно-динамической модели уровня изобретательской активности в системе AnyLogic

| Наименование | population | patent | corp_research | person_research | coeff_inv_activity |
|-------------------------|-------------------|-----------------|---------------|-------------------|--------------------|
| Изменение параметра | [135600...154235] | [21627...30732] | [3317...4384] | [672493...932115] | [1,4725...2,1105] |
| Математическое ожидание | 144705,3752 | 26204,7906 | 3782,2631 | 778989,9941 | 1,8114 |
| Дисперсия | 6347086,0729 | 4467795,1931 | 46155,5052 | 3873708843 | 0,0237 |
| Стандартное отклонение | 2519,3424 | 2113,7160 | 214,8383 | 62239,1263 | 0,1539 |

Таблица 2.

Синтетические данные для обучения исходной нейронной сети

| № | Признаки | | | | Метка |
|------|------------|--------|---------------|-----------------|--------------------|
| | population | patent | corp_research | person_research | coeff_inv_activity |
| 1 | 146890 | 28688 | 4099 | 887729 | 1,950000 |
| 2 | 146841 | 28362 | 4098 | 887553 | 1,931477 |
| 3 | 146792 | 28036 | 4097 | 887377 | 1,909913 |
| 4 | 146743 | 27710 | 4096 | 887201 | 1,888335 |
| 5 | 146694 | 27384 | 4095 | 887025 | 1,886743 |
| ... | ... | ... | ... | ... | ... |
| 9601 | 143267 | 24072 | 3604 | 732274 | 1,732500 |
| 9602 | 146545 | 29269 | 4175 | 738857 | 2,100000 |
| 9603 | 146804 | 26795 | 4032 | 722291 | 1,921500 |
| 9604 | 146880 | 22765 | 3944 | 707887 | 1,627500 |

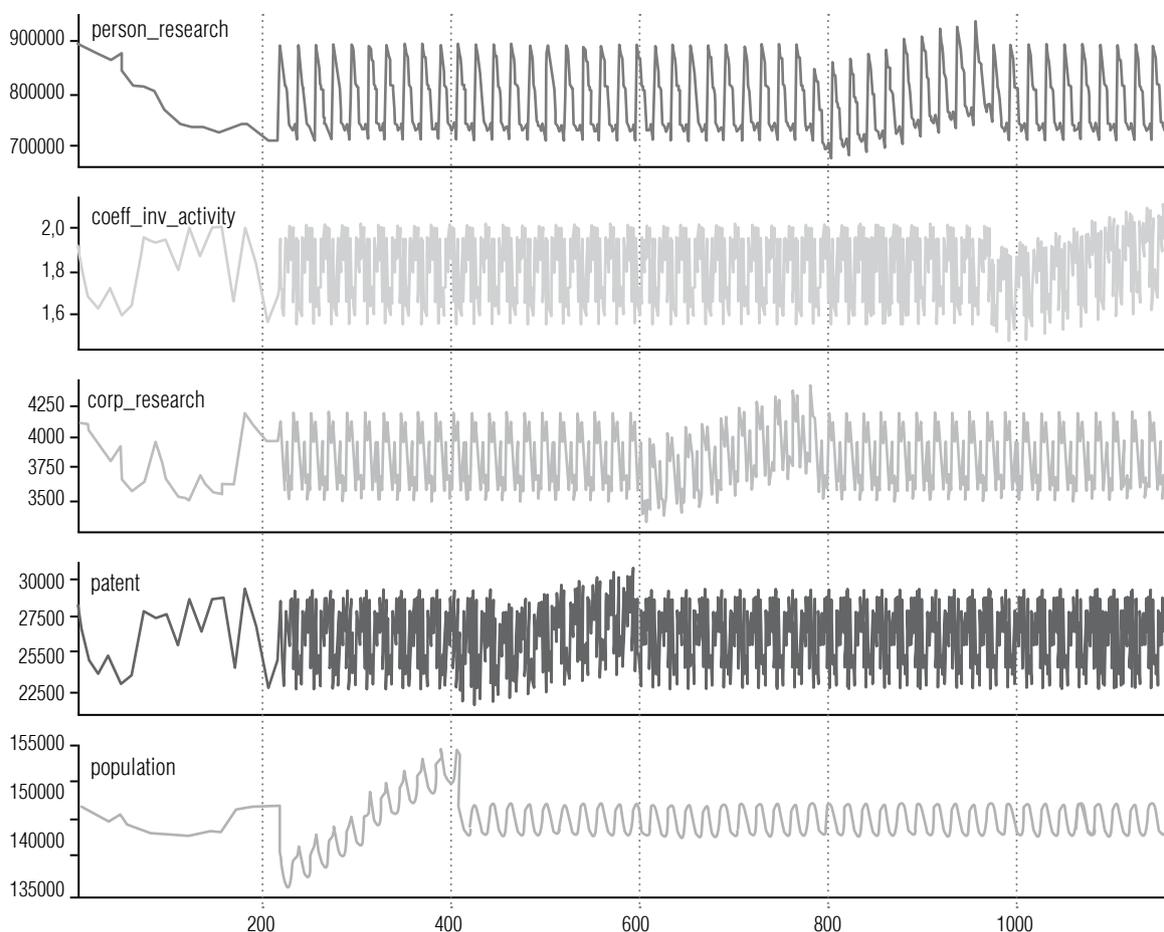


Рис. 4. Изменение во времени исходных функций

2.2. Обучение исходной LSTM и перенос обучения

Как было отмечено выше, обучение исходной LSTM осуществляется на синтетических данных, полученных в результате имитационного моделирования. Для этого используется открытая программная библиотека для машинного обучения TensorFlow, которая предоставляет хороший вспомогательный интерфейс прикладного программирования (RNN API) для реализации предсказательных моделей временных рядов.

Прежде всего, для выполнения процесса обучения загрузим синтетические данные и выполним стандартизацию набора данных, используя функции `mean()` и `std()`.

Далее задача сводится к прогнозированию многомерного временного ряда на основе некоторой предоставленной истории. Сформируем обучающие и валидационные данные и выполним непосредственное обучение исходной LSTM.

Функция `multivariate_data` выполняет задачу управления окнами. Она выбирает прошлые наблюдения на основе заданного размера шага (рисунки 5). Далее фиксируем веса предварительно обученной нейронной сети (рисунки 6). Затем создаем составную нейронную сеть на основе «Kazakov.h5» и компилируем ее (рисунки 7).

Для подбора наилучших гиперпараметров нейронной сети использовался оптимизатор Keras Tuner, разработанный командой Google и входящий в открытую библиотеку Keras. В качестве основного типом Keras Tuner был определен RandomSearch. Листинг выбора наилучшей модели с помощью Keras Tuner выглядит следующим образом (рисунки 8).

Для создания нейронной сети с перебором основных значений гиперпараметров использовалась следующая функция (рисунки 9).

Для двух полносвязанных слоев в качестве функции активации Keras Tuner определил `relu` – выпрямитель (`rectifier`) и оптимизатор `adam` – метод

```
x_train_single, y_train_single = multivariate_data(dataset, dataset[:, 1], 0,
                                                    TRAIN_SPLIT, past_history,
                                                    future_target, STEP,
                                                    single_step=True)
x_val_single, y_val_single = multivariate_data(dataset, dataset[:, 1],
                                                TRAIN_SPLIT, None, past_history,
                                                future_target, STEP,
                                                single_step=True)

single_step_model = tf.keras.models.Sequential()
single_step_model.add(tf.keras.layers.LSTM(32,
                                           return_sequences=True,
                                           input_shape=x_train_multi.shape[-2:]))
single_step_model.add(tf.keras.layers.LSTM(16, activation='relu'))
single_step_model.add(tf.keras.layers.Dense(1))

single_step_model.compile(optimizer=tf.keras.optimizers.RMSprop(clipvalue=1.0),
                          loss='mae')
```

Рис. 5. Листинг «Обучение исходной LSTM»

```
my_new_model = tf.keras.models.load_model('Kazakov_LSTM.h5')
my_new_model.trainable = False
```

Рис. 6. Листинг «Назначение весов из предварительно обученной нейронной сети»

```

▶ single_step_model = tf.keras.models.Sequential()
  single_step_model.add(my_new_model)
  single_step_model.add(tf.keras.layers.Dense(14))
  single_step_model.add(tf.keras.layers.Dense(1))

```

Рис. 7. Листинг «Формирование архитектуры составной нейронной сети»

```

[ ] tuner = RandomSearch(build_model)
  tuner.search(x_train,
              y_train,
              epochs=20,
              verbose = 1)
  models = tuner.get_best_models(num_models=1)

```

Рис. 8. Листинг «Подбор гиперпараметров нейронной сети»

```

▶ def build_model(hp):
  model = Sequential()
  activation_choice = hp.Choice('activation', values=['relu', 'sigmoid', 'tanh'])
  model.add(Dense(units=hp.Int('units_input',
                              min_value=7,
                              max_value=30),
                  activation=activation_choice))
  model.add(Dense(1, activation=activation_choice))
  model.compile(
    optimizer=hp.Choice('adam', values=['adam', 'rmsprop', 'SGD']),
    loss='mse',
    metrics=['mae'])
  return model

```

Рис. 9. Листинг «Функция подбора гиперпараметров нейронной сети»

стохастического градиентного спуска, основанный на адаптивной оценке моментов первого и второго порядка. В качестве функции потерь представлена среднеквадратичная ошибка (mse), в качестве метрики качества – средняя абсолютная ошибка (mae). На последней эпохе обучения данные параметры приняли значения 0,3995 и 0,1739 соответственно.

Таким образом, были добавлены два полностью связанных слоя для реализации доменной адаптации на основе фактических данных, полученных из официальной статистики и представленных в *таблице 3*.

Составлено на основании данных официального сайт Федеральной службы государственной статистики (<https://www.gks.ru/>)

Предполагается, что прогнозирование динамики изобретательской активности будет осуществляться на один шаг, поэтому на выходе последнего слоя сети останется один нейрон.

3. Обсуждение результатов

Разработанная имитационная модель динамики изобретательской активности позволяет формировать потенциально неограниченное количество записей для обучения исходной сети. Исследуемые методики трансфертного обучения и доменной адаптации в LSTM позволили использовать предобученную исходную сеть в новой смешанной архитектуре. Таким образом, несмотря на имеющийся критически малый набор фактических данных для

Таблица 3.

Фактические данные для реализации доменной адаптации

| № п/п | Год | Признаки | | | | Метка |
|-------|------|------------|---------|---------------|-----------------|--------------------|
| | | population | patent | corp_research | person_research | coeff_inv_activity |
| 0 | 2001 | 146304 | 24777,0 | 4037 | 885568 | 1,69 |
| 1 | 2002 | 145649 | 23712,0 | 3906 | 870878 | 1,63 |
| 2 | 2003 | 144964 | 24969,0 | 3797 | 858470 | 1,72 |
| ... | ... | ... | ... | ... | ... | ... |
| 15 | 2016 | 146804 | 26795,0 | 4032 | 722291 | 1,83 |
| 16 | 2017 | 146880 | 22765,0 | 3944 | 707887 | 1,55 |
| 17 | 2018 | 146781 | 24952,8 | 3944 | 707887 | 1,70 |

обучения нейронной сети, получена возможность осуществлять прогноз экономических показателей.

С помощью обученной нейронной сети визуализируем прогнозные значения коэффициента инновационной активности на 2012 год (на основе данных для валидационной выборки) и на 2018 год (на основе данных для тестовой выборки) (рисунки 10).

Данные, полученные на 2012 год, показали значение метки 1,91 при фактически зафиксирован-

ном в данный период значения показателя 2,00 (рисунки 10а). Коэффициент изобретательской активности, определенный сетью на 2018 год, равен 1,73 против фактического 1,70 (рисунки 10б). Таким образом, отклонение составило 4,5% в 2012 году и 1,8% в 2018 году соответственно. Полученные результаты можно считать удовлетворительными, что позволяет транслировать данный метод на перспективу.

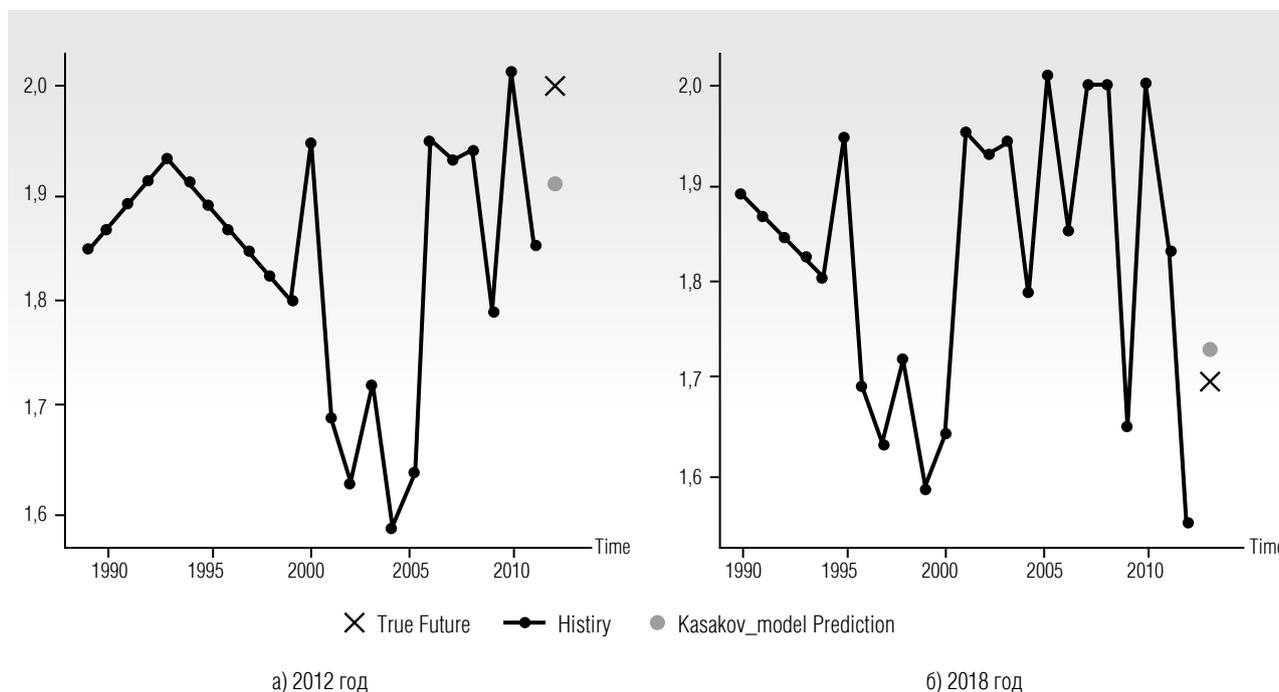


Рис. 10. Определение значения целевой переменной с помощью обученной нейронной сети

Заключение

Представленный в исследовании подход, базирующийся на построении системно-динамической модели и рекуррентной нейронной сети может быть адаптирован к другим социально-экономическим системам и процессам в части решения задач предиктивной аналитики. Авторский подход к обучению и использованию LSTM-сетей в социально-экономических системах позволит существенно повысить эффективность управленческих решений. Несомненным преимуществом применения данной методики, на наш взгляд, является возможность раннего определения трендов в процессах даже в условиях ограниченного набора данных.

Предложенный подход может стать универсальным инструментом LSTM предиктивной аналитики, поскольку исследуемые методики трансфертного обучения и доменной адаптации в LSTM позволили использовать исходную сеть, обученную на синтетических данных, и с высокой степенью

точности прогнозировать значение целевой переменной. Практическая значимость исследования состоит в расширении возможностей комплексного применения методов имитационного моделирования для построения нейронной сети. Вместе с тем разработанный подход может быть использован органами государственной власти для обоснования параметров развития социально-экономической системы и позволяет получить информацию о ее перспективном состоянии. ■

Благодарности

Исследование выполнено при финансовой поддержке РФФИ в рамках научного проекта №18-41-320003 «Математическое моделирование социально-экономического развития региона в системах поддержки принятия решений с использованием адаптивных методов машинного обучения и имитационного моделирования в условиях неопределенности».

Литература

1. Региональное и муниципальное управление социально-экономическим развитием в Сибирском федеральном округе / под ред. А.С. Новоселова. Новосибирск: ИЭОПП СО РАН, 2014.
2. Канторович Г.Г. Анализ временных рядов // Экономический журнал. 2002. № 1. С. 87–110.
3. Андерсон Т. Статистический анализ временных рядов. Москва: Мир, 1976.
4. Обзор моделей прогнозирования временных рядов: проба пера / Сообщество IT-специалистов. [Электронный ресурс]: <https://habr.com/ru/post/180409/> (дата обращения: 15.03.2020).
5. Pan S.J., Yang Q. A survey on transfer learning // IEEE Transactions on Knowledge and Data Engineering. 2010. Vol. 22. No 10. P. 1345–1359. DOI: 10.1109/TKDE.2009.191.
6. Царегородцев Е.И., Баркалова Т.Г. Имитационное моделирование в прогнозировании социально-экономических систем // Вестник ТИСБИ. 2017. № 3. С. 126–134.
7. Манкаев Н.В. Исследование и моделирование процесса управления социально-экономическими системами // Мягкие измерения и вычисления. 2019. № 1 (14). С. 21–30.
8. Звягин Л.С. Практические приемы моделирования экономических систем // Материалы IV Международной научной конференции «Проблемы современной экономики». Челябинск, 20–23 февраля 2015 г. С. 14–19.
9. Domain adaptation with latent semantic association for named entity recognition / H. Guo [et al.] // Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics (HLT-NAACL – 2009). Boulder, Colorado, USA, 31 May – 5 June 2009. P. 281–289. DOI: 10.3115/1620754.1620795.
10. Cortes C., Mohri M. Domain adaptation and sample bias correction theory and algorithm for regression // Theoretical Computer Science. 2014. No 519. P. 103–126. DOI: 10.1016/j.tcs.2013.09.027.
11. Гафаров Ф.М., Галимянов А.Ф. Искусственные нейронные сети и приложения. Казань: Казанский университет, 2018.
12. Olah C. Understanding LSTM networks // GITHUB blog. 27 August 2015. [Электронный ресурс]: <https://colah.github.io/posts/2015-08-Understanding-LSTMs/> (дата обращения: 12.03.2020).
13. Ganegedara T. Stock market predictions with LSTM in Python // GITHUB blog. 3 May 2018. [Электронный ресурс]: <https://www.data-camp.com/community/tutorials/lstm-python-stock-market> (дата обращения: 15.03.2020).
14. Кондрашова Д.А., Насыров Р.В. Сравнение эффективности методов автоматической классификации текстов // Труды VII Всероссийской научной конференции «Информационные технологии интеллектуальной поддержки принятия решений». Уфа, 28–30 мая 2019 г. С. 146–149.
15. О Стратегии инновационного развития РФ на период до 2020 г. Распоряжение Правительства РФ от 8 декабря 2011 г. № 2227-р. [Электронный ресурс]: <https://www.garant.ru/products/ipo/prime/doc/70006124/#review> (дата обращения: 27.11.2019).
16. Создание эксперимента Монте-Карло. [Электронный ресурс]: https://studme.org/286158/informatika/sozdanie_eksperimenta_monte_karlo (дата обращения: 27.11.2019).
17. Звягин Л.С. Ключевые аспекты имитационного моделирования сложных систем // Молодой ученый. 2016. №12. С. 19–23.
18. Дровяников В.И., Хаймович И.Н. Имитационное моделирование управления социальным кластером в системе AnyLogic // Фундаментальные исследования. 2015. № 8–2. С. 361–366.

19. Якимов И.М., Кирпичников А.П., Исаева Ю.Г., Аляутдинова Г.Р. Сравнение результатов имитационного моделирования вероятностных объектов в системах: AnyLogic, Arena, Bizagi modeler, GPSS W // Вестник технологического университета. 2015. Т. 18. №. 16. С. 260–264.
20. Géron A. Hands-on machine learning with Scikit-Learn and TensorFlow: concepts, tools, and techniques to build intelligent systems. Sebastopol, CA: O'Reilly Media, 2017.

Об авторах

Казаков Олег Дмитриевич

кандидат экономических наук, доцент;
заведующий кафедрой информационных технологий,
Брянский государственный инженерно-технологический университет,
241037, г. Брянск, пр. Станке Димитрова, д.3;
E-mail: kod8383@mail.ru;
ORCID: 0000-0001-9665-8138

Михеенко Ольга Валерьевна

кандидат экономических наук;
доцент кафедры государственного управления, экономической и информационной безопасности,
Брянский государственный инженерно-технологический университет,
241037, г. Брянск, пр. Станке Димитрова, д.3;
E-mail: miheenkoov@mail.ru;
ORCID: 0000-0003-0917-8406

Transfer learning and domain adaptation based on modeling of socio-economic systems

Oleg D. Kazakov

E-mail: kod8383@mail.ru

Olga V. Mikheenko

E-mail: miheenkoov@mail.ru

Bryansk State Technological University of Engineering
Address: 3, Stanke Dimitrov Avenue, Bryansk 241037, Russia

Abstract

This article deals with the application of transfer learning methods and domain adaptation in a recurrent neural network based on the long short-term memory architecture (LSTM) to improve the efficiency of management decisions and state economic policy. Review of existing approaches in this area allows us to draw a conclusion about the need to solve a number of practical issues of improving the quality of predictive analytics for preparing forecasts of the development of socio-economic systems. In particular, in the context of applying machine learning algorithms, one of the problems is the limited number of marked data. The authors have implemented training of the original recurrent neural network on synthetic data obtained as a result of simulation, followed by transfer training and domain adaptation. To achieve this goal, a simulation model was developed by combining notations of system dynamics with agent-based modeling in the AnyLogic system, which allows us to investigate the influence of a combination of factors on the key parameters of the efficiency of the socio-economic system. The original LSTM training was realized with the help of TensorFlow, an open source software library for machine learning. The suggested approach makes it possible to expand the possibilities of complex application of simulation methods for building a neural network in order to justify the parameters of the development of the socio-economic system and allows us to get information about its future state.

Key words: transfer learning; domain adaptation, simulation modeling; decision support systems; socio-economic development of regions.

Citation: Kazakov O.D., Mikheenko O.V. (2020) Transfer learning and domain adaptation based on modeling of socio-economic systems. *Business Informatics*, vol. 14, no 2, pp. 7–20. DOI: 10.17323/2587-814X.2020.2.7.20

References

- Novoselov A.S., ed. (2014) *Regional and municipal management of socio-economic development in the Siberian Federal district*. Novosibirsk: IEIE SB RAS (in Russian).
- Kantorovich G.G. (2002) Time series analysis. *Economic Journal*, no 1, pp. 87–110 (in Russian).
- Anderson T. (1976) *Statistical analysis of time series*. Moscow: Mir (in Russian).
- Community of IT specialists (2013) *Overview of time series forecasting models*. Available at: <https://habr.com/ru/post/180409/> (accessed: 15 March 2020) (in Russian).
- Pan S.J., Yang Q. (2010) A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no 10, pp. 1345–1359. DOI: 10.1109/TKDE.2009.191.
- Tsaregorodtsev E.I., Barkalova T.G. (2017) Simulation modeling in forecasting of socio-economic systems. *Herald of TISBI*, no 3, pp. 126–134 (in Russian).
- Mankaev N.V. (2019) Research and modeling of the process of socio-economic systems management. *Soft Measurements and Computing*, no 1 (14), pp. 21–30 (in Russian).
- Zvyagin L.S. (2015) Practical methods of modeling economic systems. Proceedings of the *IV International Scientific Conference on Problems of the Modern Economy. Chelyabinsk, 20–23 February 2015*, pp. 14–19 (in Russian).
- Guo H., Zhu H., Guo Z., Zhang X., Wu X., Su Z. (2009) Domain adaptation with latent semantic association for named entity recognition. Proceedings of the *Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics (HLT-NAACL – 2009). Boulder, Colorado, USA, 31 May – 5 June 2009*, pp. 281–289. DOI: 10.3115/1620754.1620795.
- Cortes C., Mohri M. (2014) Domain adaptation and sample bias correction theory and algorithm for regression. *Theoretical Computer Science*, no 519, pp. 103–126. DOI: 10.1016/j.tcs.2013.09.027.
- Gafarov F.M., Galimyanov A.F. (2018) *Artificial neural networks and applications*. Kazan: Kazan University (in Russian).
- Olah C. (2015) *Understanding LSTM networks*. GITHUB blog. Available at: <https://colah.github.io/posts/2015-08-Understanding-LSTMs/> (accessed 12 March 2020).
- Ganegedara T. (2018) *Stock market predictions with LSTM in Python*. GITHUB blog. Available at: <https://www.datacamp.com/community/tutorials/lstm-python-stock-market> (accessed 12 March 2020).
- Kondrashova D.A., Nasyrov R.V. (2019) Comparison of the effectiveness of automatic text classification methods. Proceedings of the *VII All-Russian Scientific Conference on Information Technologies of Intellectual Decision Support. Ufa, 28–30 May 2019*, pp. 146–149 (in Russian).
- Decree of the Government of the Russian Federation No 2227-R of 8 December 2011. *About the strategy of innovative development of the Russian Federation for the period up to 2020*. Available at: <https://www.garant.ru/products/ipo/prime/doc/70006124/#review> (accessed: 27 November 2019) (in Russian).
- Creating the Monte Carlo experiment*. Available at: https://studme.org/286158/informatika/sozdanie_eksperimenta_monte_karlo (accessed: 27 November 2019) (in Russian).
- Zvyagin L.S. (2016) Key aspects of complex systems simulation. *Young Scientist*, no 12, pp. 19–23 (in Russian).
- Drovyannikov V.I., Khaimovich I.N. (2015) Simulation of social cluster management in the AnyLogic system. *Fundamental Study*, no 8–2, pp. 361–366 (in Russian).
- Yakimov I.M., Kirpichnikov A.P., Isaeva Yu.G., Alyautdinova G.R. (2015) Comparison of simulation results for probabilistic objects in the systems: AnyLogic, Arena, Bizagi modeler, GPSS W. *Bulletin of the Technological University*, vol. 18, no 16, pp. 260–264 (in Russian).
- Géron A. (2017) *Hands-on machine learning with Scikit-Learn and TensorFlow: concepts, tools, and techniques to build intelligent systems*. Sebastopol, CA: O'Reilly Media.

About the authors

Oleg D. Kazakov

Cand. Sci. (Econ.), Associate Professor;

Head of Department of Information Technology, Bryansk State Technological University of Engineering,
3, Stanke Dimitrov Avenue, Bryansk 241037, Russia;

E-mail: kod8383@mail.ru;

ORCID: 0000-0001-9665-8138

Olga V. Mikheenko

Cand. Sci. (Econ.);

Associate Professor, Department of Public Administration, Economic and Information Security,
Bryansk State Technological University of Engineering,
3, Stanke Dimitrov Avenue, Bryansk 241037, Russia;

E-mail: miheenkoov@mail.ru;

ORCID: 0000-0003-0917-8406