

А.И. ТИТОВ
**ИСПОЛЬЗОВАНИЕ НЕЙРОСЕТЕВОЙ АППРОКСИМАЦИИ
ПРИ ОЦЕНКЕ ТРУДОЕМКОСТИ РАЗРАБОТКИ
ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**

Титов А.И. Использование нейросетевой аппроксимации при оценке трудоемкости разработки программного обеспечения.

Аннотация. В статье рассматриваются основные модели оценки трудоемкости разработки ПО. Подробно рассмотрена одна из самых распространенных методик оценки — модель СОСОМО. Предложен подход для повышения уровня точности оценки по модели СОСОМО на основе нейросетевой аппроксимации. Обосновывается выбор нейросети с обратным распространением ошибки в качестве аппроксиматора. Приводятся численные результаты обучения нейросети, использующей параметры модели СОСОМО в качестве входных параметров.

Ключевые слова: оценка трудоемкости разработки ПО, модель СОСОМО, нейронная сеть.

Titov A.I. Software Development Effort Estimation Using a Neuro Network Approximation Approach.

Abstract. The article describes the main effort estimation models for software development. It is spoken in detail about the most widely used software effort estimation model, the Constructive Cost Model (COCOMO). An approach to improve the accuracy of estimates of COCOMO model based on neural network approximation is proposed. It deals with the choice of a neural network with back-propagation errors as an approximator. Data are given about numerical results of neural network learning using COCOMO model parameters as input.

Keywords: software development effort estimation; constructive cost model; artificial neural network.

1. Введение. При управлении проектами в сфере информационных технологий важную роль играет создание обоснованного плана выполнения проекта. По мнению многих специалистов в области управления ИТ-проектами, ошибки в оценке (предсказании ключевых параметров будущих работ) являются одной из наиболее распространенных причин возникновения проблем в проектах по разработке ПО. Существует ряд исследований, согласно которым в среднем реальные трудозатраты превышают исходную оценку трудоемкости на 21% [1-2].

Недооценка объема функциональности, трудозатрат и сроков исполнения проекта может нарушить задуманный порядок работ и привести к увеличению бюджета проекта, а также снизить качество выполняемых работ. При этом слишком высокая оценка проекта также несет в себе определенные риски. Переоценка трудоемкости может повлиять на скорость выполнения работ, поскольку проектная команда будет ориентироваться на сроки выполнения, полученные исходя из переоцененного плана. Помимо этого, завышенная оценка также озна-

чает завышенный бюджет проекта и расход ресурсов, а следовательно, снижение конкурентоспособность исполнителя.

Исходя из этих рисков, при построении планов работ необходимо дать оценку трудоемкости с достаточно высоким уровнем точности. Для того чтобы оценка трудоемкости обладала необходимой точностью и могла быть пересмотрена в процессе разработки ПО она должна удовлетворять следующим условиям [3]:

1. Оценка трудоемкости создания программного компонента должна опираться на предшествующий опыт разработки подобных компонентов.
2. Необходимо оценивать трудоемкость вновь создаваемого компонента, так и работу по внесению изменений в уже реализованные компоненты.
3. Оценка должна учитывать квалификацию участников проектной команды.
4. Оценка должна учитывать расходы на весь жизненный цикл ПО, включая руководство проектом.

Существует большое число методов для оценки трудозатрат на разработку программного обеспечения, такие как PERT, COSOMO и COSOMO II, Functional Points, Use Case Points и многие другие [4-9]. Основным недостатком этих методов в том, что результаты анализа проекта, на которых строится оценка, не всегда могут быть применимы в равной степени к другим проектам. Из-за сложности адаптации методов к конкретному проекту точность расчета может значительно снизиться.

Исходя из этого, при оценке трудоемкости возникает задача внесения изменений в методику расчета с учетом данных, полученных от оценок предыдущих проектов или отдельных компонентов. Одним из возможных подходов к решению этой задачи является использование нейросетевой аппроксимации оценки трудоемкости. При обучении нейронной сети в качестве исходных данных используются базовые (полученные в результате оценки) характеристики программного проекта и его фактические трудозатраты. Если имеющихся данных для обучения нейронной сети недостаточно, то дополнительные наборы параметров трудоемкости и выходных характеристик трудозатрат разработки программного проекта можно получить, к примеру, с помощью линейной или сплайн аппроксимации.

В качестве базовой модели расчета трудоемкости выбрана модель COSOMO, поскольку она обладает следующими преимуществами:

- широкая распространенность в ИТ-отрасли;
- использование большого числа параметров, за счет чего достигается полнота модели проекта;

- возможность расчета для разных уровней детализации (базового, промежуточного и детального);

2. Описание выбранного подхода. Оценка трудоемкости по модели СОСОМО имеет ряд недостатков:

1. Отсутствие возможности корректировки оценки по данным, полученным с предыдущих проектов.

2. Использование в качестве метрики объема проекта количество строк кода (KSLOC). В усовершенствованном варианте методики СОСОМО II предусмотрена возможность использования функциональных точек вместо количества строк кода, но для итогового расчета функциональные точки необходимо перевести в строки кода, что влечет за собой снижение точности оценки.

3. При необходимости добавления новых параметров требуется проведение регрессионного анализа для калибровки модели.

Эти недостатки могут сильно снизить точность вычисления, а также повысить трудозатраты при необходимости скорректировать модель под требования определенной проектной команды. Для устранения этих недостатков предлагается подход на основе нейросетевой аппроксимации. Общая схема предлагаемого подхода представлена на рисунке 1.



Рис.1. Общая схема подхода к использованию нейросетевой аппроксимации в оценке трудоемкости

Основная идея данного подхода заключается в том, что между входными параметрами проекта и его выходной трудоемкостью устанавливается функциональная зависимость, реализуемая с помощью нейронной сети. При обучении нейронной сети в качестве исходных данных используются известные параметры проекта, выбранные в за-

зависимости от используемой модели оценки, а также выходные характеристики трудоемкости. Если имеющихся данных для обучения нейронной сети недостаточно, то дополнительные наборы параметров и выходных характеристик можно получить, к примеру, с помощью линейной или сплайн аппроксимации. После того как нейронная сеть обучена на данных существующих проектов, возможно ее использование для прогнозирования трудозатрат последующих проектов.

Главным преимуществом этого подхода является универсальность. Прогнозирование трудоемкости с помощью нейронной сети может быть применено к различным моделям оценки. Кроме того, возможно использование в условиях изменения состава входных параметров модели. Например, при использовании модели СОСОМО II возможен расчет трудоемкости без перевода функциональных точек в строки кода, что позволяет повысить точность оценки и получить предварительный прогноз на ранних этапах проектирования программного обеспечения.

Основной минус использования нейронной сети заключается в том, что для получения приемлемого уровня точности требуется обучение на реальных примерах. В ситуациях, когда отсутствуют выполненные проекты или их статистика, следует использовать методы оценки без нейросетевой аппроксимации. По мере появления реальных данных по разрабатываемым и внедряемым проектам обучение нейронной сети станет возможным, а с ростом числа обучающих примеров точность оценки будет расти.

3. Вычислительный эксперимент. Для того чтобы оценить возможности повышения точности оценки с помощью нейросетевой аппроксимации, были проведены эксперименты по моделированию и обучению нейросети в пакете Matlab Network Neural.

В качестве архитектуры нейросети выбрана сеть с прямым распространением сигнала и обратным распространением ошибки. Для входных параметров были выбраны 15 атрибутов стоимости CD_k из модели СОСОМО и объем программного средства в тысячах строках кода (KSLOC).

Архитектура выбранной нейросети, смоделированной в пакете Matlab Network Neural представлена на рисунке 2.

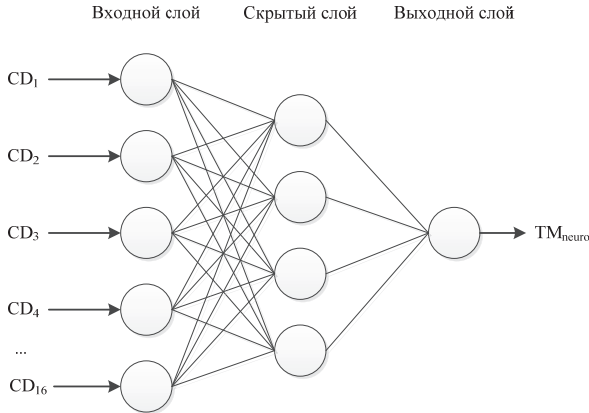


Рис.2. Архитектура выбранной нейросети в Matlab Network Neural

В качестве источника данных были взяты проекты, использованные при разработке модели COSOMO [10]. Из исходных 63 проектов случайным образом были выделены 5, которые использовались как тестовые для оценки точности моделей. На первом этапе обучения нейросети был введен обучающий набор данных по 20 проектам, выбранным из исходных. На втором этапе в обучающий набор были добавлены еще 20 проектов, а на третьем этапе еще 16. После обучения нейросети на обучающем наборе данных, на каждом этапе моделирования было проведено тестирование обученной нейросети.

Для сравнения точности нейросетевого подхода с точностью расчета по методу COSOMO были проведены вычисления величины времени разработки в календарных месяцах (ТМ).

$$TM = c_i \times (PM)^{d_i},$$

где PM (People × Month) — трудоемкость (чел. мес.);

Формула для расчета трудоемкости промежуточного уровня модели имеет вид:

$$PM = EAF \times a_i \times (SIZE)^{b_i},$$

где SIZE — объем программного продукта в тысячах строк исходного текста (KiloofSourceLineofCode —KSLOC);

EAF (Effort Adjustment Factor) — произведение выбранных атрибутов (коэффициентов) стоимости (Cost Drivers) CD_k :

$$EAF = \prod_{k=1}^{15} CD_k.$$

Значения коэффициентов a_i, b_i, c_i, d_i были выбраны для органического типа проекта (см. таблицу 1), который характеризуется тем, что проект выполняется небольшой группой специалистов, имеющих необходимый опыт разработки.

Таблица 1. Значения коэффициентов базового уровня модели

Тип проекта	a_b	b_b	c_b	d_b
Органический	2.4	1.05	2.5	0.38
Полуразделенный	3.0	1.12	2.5	0.35
Встроенный	3.6	1.20	2.5	0.32

В таблице 2 приведена часть значений атрибутов (коэффициентов) стоимости CD_k на примере фактора «характеристики продукта» (ProductAttributes).

Таблица 2. Атрибуты (коэффициенты) трудоемкости среднего уровня модели COSOMO [10]

Факторы стоимости	Рейтинг					
	очень низкий	низкий	средний	высокий	очень высокий	критический
Сложность продукта	0,70	0,85	1,00	1,15	1,30	1,65
Требуемая надежность ПО	0,75	0,88	1,00	1,15	1,40	-
Размер БД приложения	-	0,94	1,00	1,08	1,1	-

В качестве критерия точности оценки выбрана величина среднего отклонения, которая вычислялась для каждого тестового проекта по следующей формуле:

$$X = \frac{|TM_{\text{actual}} - TM|}{TM_{\text{actual}}} \times 100,$$

где TM_{actual} — реальные трудозатраты;

TM — предполагаемые трудозатраты, вычисленные по одному из методов оценки.

Также для оценки рассеивания отклонений, вычисленных по методу COSOMO и при помощи нейросети, для каждого метода была определена остаточная дисперсия:

$$\sigma^2 = \frac{\sum_{n=1}^n (TM - TM_{\text{actual}})^2}{n},$$

где n — количество тестовых проектов. Результаты вычислений приведены в таблице 3.

Таблица 3. Результаты вычисления времени разработки по методике COSOMO и при помощи нейросети

Номер проекта	ТМ _{актуа} , мес.	ТМ _{cosomo} , мес.	ТМ _{нейро} , мес.		
			20 обучающих проектов	40 обучающих проектов	56 обучающих проектов
1	18,2	15,53	16,69	19,63	14,29
2	11,3	10,46	20,69	15,00	13,15
3	16,4	14,65	15,98	17,50	15,21
4	15,3	9,49	12,83	11,18	16,02
5	11,6	8,38	15,99	12,28	13,21
Среднее значение отклонения X, %		19,7	29,6	16,0	12,7
Остаточная дисперсия σ^2		11,01	23,20	6,97	4,65

Как можно увидеть из результатов вычислений, среднее отклонение по трудоемкостям проектов, вычисленное методом COSOMO, составило 19,7%. Среднее отклонение, вычисленное смоделированной нейросетью, составило 12,7%. При этом уже после второго этапа обучения нейросети получены более точные результаты, чем по методу COSOMO. Оценки рассеивания отклонений результатов моделирования времени разработки от фактических данных также подтверждают более высокую точность нейросетевой аппроксимации.

Полученные результаты позволяют сделать вывод о том, что предложенный подход применения нейросетевой аппроксимации является перспективным для дальнейших исследований в направлении повышения точности прогнозирования трудозатрат при разработке ПО.

4. Заключение. Результатом применения данного подхода является получение общей функциональной зависимости прогнозных выходных характеристик производительности разработки программного проекта от его параметров трудоемкости. Достоинством предлагаемого подхода является его высокая общность и универсальность в смысле допустимости предположек о входных параметрах разработки. Так, например, при изменении исходных параметров возможно применение этого подхода и к другим моделям оценивания трудоемкости, таким как COSOMO II или метод функциональных точек.

Также можно отметить, что по результатам вычислительного эксперимента метод аппроксимации оценки трудоемкости разработки программного обеспечения с помощью нейронной сети может обеспечивать большую точность, по сравнению с методом COSOMO, при идентичных входных данных.

В качестве направлений дальнейших исследований целесообразно выделить: развитие регрессионной модели оценивания COSOMO [19], обоснование выбора архитектуры нейронной сети для

различных моделей оценки, применение нейросетевой аппроксимации к методу функциональных точек.

Литература

1. *Jørgensen M.* A Review of Studies on Expert Estimation of Software Development Effort // *Journal of Systems and Software*. 2004. vol. 70. Issue 1-2. pp. 37–60.
2. *Jørgensen M.* The Use of Precision of Software Development Effort Estimates to Communicate Uncertainty // *Software Quality Days. The Future of Systems and Software Development*. Springer. 2016. pp. 156–168.
3. *Гольфанд И.Я., Хлебутин П.С.* Оценка трудозатрат разработки программной компоненты // *Труды ИСА РАН*. 2005. Т. 15. С. 125–135.
4. *Фатрелл Р.Т., Шафер Д.Ф., Шафер Л.И.* Управление программными проектами. Достижение оптимального качества при минимуме затрат: пер. с англ. // М., СПб., Киев: Вильямс. 2004. 1136 с.
5. *Макконелл С.* Сколько стоит программный проект // М.: Русская редакция; СПб.: Питер. 2007. 304 с.
6. *Архипенков С.Я.* Лекции по управлению программными проектами // М. 2009. 127 с.
7. *Соммервилл. И.* Инженерия программного обеспечения: пер. с англ. // М.: Вильямс. 2002. 624 с.
8. *Тележкин А.М.* Применение алгоритмических сетей для оценки ресурсов в программных проектах. Дисс. канд. техн. наук // СПб. 2015.
9. *Кульдин С.П.* Генетический подход к проблеме оценки сроков и трудоемкости разработки программного обеспечения с заданными требованиями к качеству // *Прикладная информатика*. 2010. № 5 (29). С. 30–42.
10. *Voehn B.* Software Engineering Economic // Prentice Hall. 1981. 768 с.
11. *Voehn B.* Software cost estimation with COCOMO II // Englewood Cliffs, NJ: Prentice Hall. 2000. 540 с.
12. *Круглов В.В., Борисов В.В.* Искусственные нейронные сети. Теория и практика // М.: Горячая линия-Телеком. 2002. 2-е изд. 382 с.
13. *Медведев В.С., Потемкин В.Г.* Нейронные сети. MATLAB 6 // М.: Диалог-МИФИ. 2002. 496 с.
14. *Горбань А.Н.* Обобщенная аппроксимационная теорема и вычислительные возможности нейронных сетей // *Сибирский журнал вычислительной математики*. 1998. Вып. 1. № 1. С. 12–24.
15. *Hornik K.* Approximation capabilities of multilayer feedforward networks // *Neural Networks*. 1991. vol. 4. pp. 251–257.
16. *Филатова Т.В.* Применение нейронных сетей для аппроксимации данных // *Вест. Том. гос. ун-та*. 2004. № 284. С. 121–125.
17. *Павлова А.И., Лончакова О.Ю.* Сравнительный анализ применения нейронных сетей для аппроксимации функцией // *ScienceTime*. 2015. № 5 (17). С. 118–121.
18. *Хомоненко А.Д., Яковлев Е.Л.* Нейросетевая аппроксимация характеристик многоканальных немарковских систем массового обслуживания // *Труды СПИИРАН*. 2015. Вып. 4(41). С. 81–91.
19. *Рогальчук В.В., Хомоненко А.Д.* Метод обратной трассировки и оценивание его влияния на стоимость разработки программного обеспечения // *Научно-технические ведомости СПбГПУ. Информатика. Телекоммуникации. Управление*. 2008. № 4 (62). С. 146–151.

References

1. *Jørgensen M.* A Review of Studies on Expert Estimation of Software Development Effort. *Journal of Systems and Software*. 2004. vol. 70. Issue 1-2. pp. 37–60.
2. *Jørgensen M.* The Use of Precision of Software Development Effort Estimates to Communicate Uncertainty. *Software Quality Days. The Future of Systems and Software Development*. 2016. pp. 156–168.

3. Gol'fand I.Ja., Hlebutin P.S. [Effort estimation development software components]. *Trudy ISA RAN – Proceedings of ISA RAS*. 2005. vol. 15. (In Russ.).
4. Fatrell R.T., Shafer D.F., Shafer L.I. *Upravlenie programmnyimi proektami. Dostizhenie optimal'nogo kachestva pri minimumu zatrat* [Quality software project management first edition]. M. SPb.: Kiev: Vil'jams. 2004. 1136 p. (In Russ.).
5. Makkonell S. *Skol'ko stoit programmyj proekt* [Software Estimation: Demystifying the Black Art]. M.: Russkaja redakcija. SPb.: Piter. 2007. 304 p. (In Russ.).
6. Arhipenkov S.Ja. *Lekcii po upravleniju programmnyimi proektami* [Software project management lectures]. M. 2009. 127 p. (In Russ.).
7. Sommerville I. *Software Engineering*, 6th Edition. Addison Wesley Publishing Company. 2000. 6th Edition. 720 p. [Russian ed.: Sommerville I. *Inzhenerija programmnogo obespechenija*. M.: Vil'jams. 2002. 624 p].
8. Telezhkin A.M. *Primenenie algoritimicheskikh setej dlja ocenki resursov v programmnyh proektah: diss. kand. tehn. nauk* [The use of algorithmic networks to assess resources in software projects: PhD thesis]. SPb. 2015. (In Russ.).
9. Kul'din S.P. [Genetic approach to the problem of estimating the timing and complexity of software development with the specified quality requirements]. *Prikladnaja informatika – Applied Informatics*. 2010. vol. 5 (29). pp. 30–42. (In Russ.).
10. Boehm B. *Software Engineering Economic*. Prentice Hall. 1981. 768p.
11. Boehm B. *Software cost estimation with COCOMO II*. Englewood Cliffs. NJ: Prentice Hall. 2000. 540p.
12. Kruglov V.V., Borisov V.V. *Iskusstvennye nejronnye seti. Teorija i praktika*. [Artificial neural networks. Theory and practice. Second edition]. M.: Gorjachaja linija-Telekom. 2002. 2e izd. 382p. (In Russ.).
13. Medvedev V.S., Potemkin V.G. *Nejronnyeseti. MATLAB 6* [Neural Networks. Matlab 6]. M.: Dialog – MIFI. 2002. 496p. (In Russ.).
14. Gorban' A.N. [Generalized approximation theorem and computing capabilities of neural networks]. *Sibirskij zhurnal vychislitel'noj matematiki – Siberian Journal of Numerical Mathematics*. vol. 1. no. 1. 1998. 12–24 pp. (In Russ.).
15. Hornik K. Approximation capabilities of multilayer feedforward networks. *Neural Networks*. 1991. vol. 4. pp. 251–257.
16. Filatova T.V. [Application of neural networks to approximate data]. *Vestn. Tom. gos. un-ta – Bulletin of the Tomsk State University*. 2004. vol. 284. 121–125 pp. (In Russ.).
17. Pavlova A.I., Lonchakova O.Ju. [Comparative analysis of the use of neural networks for approximation of functions]. *ScienceTime*. 2015. vol. 5 (17). 118–121 p. (In Russ.).
18. Homonenko A.D., Jakovlev E.L. [Neural network approximation of characteristics of multi-channel non-markovian queuing systems]. *Trudy SPIIRAS – Proceedings of SPIIRAS*. 2015. vol. 4 (41). 81–91 p. (In Russ.).
19. Rogal'chuk V.V., Homonenko A.D. [Reverse Tracing method and evaluating its impact on the cost of the software development]. *Nauchno-tehnicheskie vedomosti SPb GPU. Informatika. Telekommunikacii. Upravlenie – Scientific and technical SPb SPU statements. Computer science. Telecommunications and Control Systems*. 2008. vol. 4 (62). pp 146–151. (In Russ.).

Титов Александр Игоревич — аспирант кафедры информационных и вычислительных систем, ФГБОУ ВПО Петербургский государственный университет путей сообщения Императора Александра I. Область научных интересов: нейронные сети, разработка программного обеспечения, управление проектами. Число научных публикаций — 2. titovvvv@rambler.ru, <http://www.pgups.ru>; Московский пр., 9, Санкт-Петербург, 190031; п.т.: 8(812)457-80-23, Факс: 8(812)310-75-25.

Titov Alexandr Igorevich — postgraduate student of information and computing systems department, Petersburg State Transport University. Research interests: artificial neural network, software development, project management. The number of publications — 2. titovvvv@rambler.ru, <http://www.pgups.ru>; 9, Moskovsky pr., Saint Petersburg, 190031; office phone: 8(812)457-80-23, Fax: 8(812)310-75-25.

РЕФЕРАТ

Титов А.И. **Использование нейросетевой аппроксимации при оценке трудоемкости разработки программного обеспечения.**

В данной статье рассматриваются вопросы оценки трудоемкости разработки программного обеспечения. Оценка трудоемкости очень важна в управлении проектом, поскольку влияет на построение календарного плана работ и объем выделяемых ресурсов. Неточная оценка несет в себе большие риски для проекта, причем отрицательно влияет как заниженная оценка, так и завышенная.

Проанализированы основные критерии для оценки трудоемкости, а также наиболее широко используемые методы оценки. Основной недостаток этих методов в том, что данные, полученные с завершенных проектов, не всегда могут быть адаптированы к новым проектам. Исходя из этого, был предложен метод на основе нейронных сетей, который позволяет решить задачу внесения изменений в метод расчета с учетом фактических данных по трудозатратам.

Предлагаемый метод состоит в том, что между входными параметрами проекта и его выходной трудоемкостью устанавливается аппроксимирующая функция, которая реализуется с помощью нейронной сети. При обучении нейронной сети в качестве исходных данных используются известные параметры проекта, выбранные в зависимости от используемой модели проекта, а также выходные характеристики трудоемкости. После того как нейронная сеть обучена на завершенных проектах, ее можно использовать для прогнозирования трудоемкости новых проектов.

Для получения оценки точности этого подхода был проведен вычислительный эксперимент, где в качестве модели проекта была использована модель СОСОМО, а в качестве исходных данных проекты, использованные при разработке этой модели. Приведенные результаты показывают более высокий уровень точности, чем в исходном методе СОСОМО при идентичных входных данных.

SUMMARY

Titov A.I. **Software Development Effort Estimation Using a Neuro Network Approximation Approach.**

The article describes the effort estimation models for software development. Efforts estimation is great importance in the management of the project, because the impact on the construction schedule of works and the amount of resources allocated. Incorrect estimation carries great risks for the project, and negatively affects both underestimation and overestimation.

It analyzes the main criteria for effort estimation, as well as the most widely used valuation methods. The main disadvantage of these methods is that the data obtained from completed projects may not always be adapted to the new projects. Therefore, the method based on neural networks was proposed, which can solve the problem of changes in the calculation method based on actual data of estimation.

The proposed method consists in the fact that between the input parameters of the project and its complexity output set approximating a function that is implemented using a neural network. When training the neural network in the known parameters of the project are used as input data, selected depending on the model of the project, as well as the complexity of the output characteristics. Once the neural network trained on completed projects, it can be used to predict the complexity of new projects.

Computational experiment was conducted to obtain an estimate of the accuracy of this approach, where the project model COCOMO model was used, and as an input to projects used in the development of this model. These results show a higher level of precision than in the original method COCOMO under identical input data.