

МЕТОД ОЦЕНИВАНИЯ УМЕНИЙ И НАВЫКОВ ПРИ ОБУЧЕНИИ ПРОГРАММИРОВАНИЮ

Введение

В самом общем виде процесс обучения представляет собой совместную деятельность обучающего и обучаемого. Более детальный анализ показывает, что взаимодействие преподавателя и студента при обучении состоит из ряда принципиально различных видов деятельности:

- информационная – студенту для изучения предоставляется учебный материал в некоторой форме: лекции, тексты, рисунки, графики и т.п.;
- тренирующая – студент учится применять полученные знания на практике, решая задачи;
- контролирующая – знания и умения студента проверяются и оцениваются;
- административная – ведется учет показателей процесса обучения каждого студента.

Очевидно, что при изучении любой предметной области для обучаемого преобладающими являются первые два вида деятельности, особенно при самообучении. Контроль и администрирование дают возможность управлять процессом обучения и обеспечивают обратную связь. Эти два вида деятельности практически всегда осуществляет преподаватель, хотя и при самостоятельной работе контроль необходим.

Автоматизация процесса обучения предполагает автоматизацию всех видов учебной деятельности. Специфика обучения программированию заключается в том, что тренирующая деятельность является преобладающей для обучаемого. Главное, чему должен научиться программист – это писать программы. Поэтому обучающая среда по программированию должна в первую очередь обеспечивать обучение написанию программ. При этом особую важность приобретают автоматизированные методы контроля.

В настоящее время автоматизированный контроль осуществляется исключительно посредством тестирования. Однако для определения уровня компетентности в программировании тестирования совершенно недостаточно. Любая профессиональная компетенция представляет собой известную триаду: знания, умения и навыки. Тестирование в какой-то мере позволяет определить только достигнутый уровень знаний предметной области. При обучении программированию необходимо оценивать уровень умений и навыков в написании программ.

Опытный программист отличается от начинающего тем, что он выполняет работу значительно быстрее и с меньшим количеством ошибок. Опыт – это результат многократного решения самых разнообразных программистских задач. Желательно, чтобы программист мог набрать подобный опыт в процессе обучения, а не во время профессиональной разработки. В данной работе предлагается способ управления обучением решению задач и метод оценки достигнутого уровня умений и навыков.

Неформальная постановка задачи

При выполнении любого задания обучаемый совершает некоторые операции (действия) в обучающей среде. Например, для создания диаграммы классов обучаемый должен выбрать вид диаграммы, тип узла (класс), создать узлы-классы, задать названия классов, выбрать типы связей между классами, проставить связи на диаграмме и т.п.

При создании программы тоже выполняется определенный набор действий: требуется задать последовательность операторов и объявить каждый объект программы, присвоив объектам имена. Для объектов-данных требуется задать тип значений. Для каждой процедуры и функции необходимо задавать имя и список параметров, в котором для каждого параметра указывать тип.

Для выполнения конкретного задания требуется совершить в обучающей среде определенный фиксированный набор операций. Назовем это множество операций правильными. Количество правильных операций составляет *трудоемкость* задания. Действия, совершаемые обуча-

емым при выполнении задания, могут быть либо правильными, либо ошибочными. Назовем все действия, совершаемые обучаемым, *учебными действиями*.

Очевидно, что количество ошибочных операций, совершаемых при выполнении конкретного задания, будет тем меньше, чем большим объемом знаний, умений и навыков обладает обучаемый – этот объем составляет опыт обучаемого. Начинаящий совершает много ошибок. По мере научения деятельность обучаемого становится все более безошибочной. Поэтому задачу обучения умениям и навыкам можно сформулировать как задачу минимизации ошибочных операций при выполнении заданий.

Рассмотрим работу обучающей системы в одном сеансе обучения, в котором ученику предоставляется одно задание для выполнения. Одну учебную операцию, выполняемую обучаемым, будем считать единицей учебного ресурса. Перед выполнением задания обучаемому предоставляется обучающей системой некоторый объем этого ресурса, заведомо больший, чем трудоемкость выполнения задания. Выполняя операции, обучаемый расходует этот ресурс. Система отслеживает действия обучаемого, и организует обратную связь, вычисляя расход ресурса и его остаток. Эта информация сообщается обучаемому, чтобы он мог скорректировать свои действия.

Если обучающий совершает много ошибочных действий, он быстро расходует предоставленный объем ресурса. В некоторый момент ресурса останется настолько мало, что для завершения задания придется действовать безошибочно. В этот момент система переходит в режим институционального (принудительного) управления. В этом режиме множество разрешенных операций ограничивается правильными операциями. Режим институционального управления подобен обучению под присмотром наставника, когда тот обучает ученика правильным действиям для выполнения задания. Таким образом, обучаемый получает возможность завершить задание.

Режим институционального обучения может быть включен с самого начала, если система предоставит обучаемому объем ресурса, равный трудоемкости. Подобный режим работы соответствует первому уроку, когда учитель просто показывает, как надо правильно действовать для решения поставленной задачи.

Таким образом, обучающая система может работать в трех режимах:

- институциональный – действиями обучаемого управляет система;
- свободный (контрольный) – обучаемый совершает любые действия, система не вмешивается в управление, но ведет журнал действий пользователя и вычисляет метрики-оценки;
- полусвободный (обучающий) – обучаемый совершает действия в пределах предоставленного ресурса, система переходит в институциональный режим при исчерпании ресурса, система ведет журнал действия пользователя и вычисляет метрики-оценки.

Система организует множество обучающих сеансов, что позволяет обучаемому постепенно набирать опыт и действовать все более безошибочно. Отметим, что система может управлять обучением в самых разнообразных режимах. Например, разрешить или запретить обучаемому пользоваться помощью. Или включить информационный режим, в котором каждое неверное действие обучаемого сопровождается подробным сообщением об ошибке.

Критерии оценивания

На момент выполнения контрольного задания обучаемый обладает некоторым уровнем обученности. Поскольку время выполнения контрольного задания мало по сравнению с общим временем обучения, то можно считать, что за время выполнения контрольного задания текущий уровень обученности остается неизменным. В контрольном режиме система не переходит в режим институционального управления, поэтому доля правильных действий от общего количества учебных действий, совершенных обучаемым, может служить оценкой достигнутого уровня обученности.

В контрольном режиме помимо количества ошибочных действий дополнительным критерием для определения уровня навыков служит время выполнения задания. Для контрольного задания задается нормативное время, в течение которого задание должно быть выполнено. Очевидно, что оценка обучаемого, не уложившегося в нормативное время, должна быть снижена.

Формальная постановка задачи

Предположим, что система содержит N однотипных заданий, и обучаемому предоставляется одно задание в сеансе.

Пусть Q – это множество учебных действий, D – это трудоемкость задания, а E – множество ошибочных действий. Тогда $Q = D \cup E$, и $|Q| = |D| + |E|$, где $|X|$ – мощность множества X . Система проводит несколько сеансов обучения-контроля, и в i -м сеансе обучаемому предоставляется ресурс $|Q_i|$:

$$|Q_i| = |D_i| + |E_i| \quad (1)$$

Рассмотрим сначала управление обучением без учета времени. За время i -го сеанса фактический объем учебных действий составит:

$$q_i = (d_i + e_i) \leq |Q_i|, \quad (2)$$

где d_i – количество выполненных обучаемым правильных операций, $d_i \leq |D_i|$; e_i – количество совершенных ошибочных действий, $e_i \leq |E_i|$.

При выполнении задания система вычисляет «расстояние» до завершения задания:

$$h_i = (|D_i| - d_i) / |D_i|$$

Эта величина сообщается обучаемому, чтобы он знал, сколько обучаемому еще осталось действий до завершения задания.

Система переходит в режим институционального управления, если в сеансе обучаемый совершил $q_i = (d_i + |E_i|)$ учебных действий. Лимит ошибок исчерпан, и система принудительно заставляет обучаемого совершить $f_i = |D_i| - d_i$ правильных операций. Задание считается выполненным. Если обучаемый самостоятельно выполнил $d_i = |D_i|$ правильных операций, то задание считается выполненным независимо от количества ошибок.

После завершения задания требуется вычислять текущий уровень обученности и определить объем предоставляемого ресурса $|Q_{i+1}|$ для следующего сеанса. Пусть r – уровень обученности. Без учета времени в i -м обучающем сеансе уровень обученности r_i вычисляется по формуле:

$$r_i = d_i / (|D_i| + e_i). \quad (3)$$

Очевидно, что $0 \leq r \leq 1$, причем $r = 0$ означает полное отсутствие каких-либо знаний, умений, навыков. При одновременном стремлении $d \rightarrow |D|$ и $e \rightarrow 0$ уровень обученности стремится к 1, что и является целью обучения.

Система должна проверять выполнение условия $r_i > \varepsilon_b$, где ε_b – некоторая положительная константа, $0 < \varepsilon_b < 1$. Эта константа определяет начальный минимальный уровень обученности, поэтому неравенство $r_i \leq \varepsilon_b$ означает, что обучаемый совершил слишком много ошибок, и нужно повторить в следующем сеансе то же самое задание. В этом случае объем ресурса для следующего сеанса вычисляется по формуле:

$$|Q_{i+1}| = |D_i| + |D_i|(1 - r_i). \quad (4)$$

В противном случае система выдает следующее задание, и объем ресурса вычисляется так:

$$|Q_{i+1}| = |D_{i+1}| + |D_{i+1}|(1 - r_i). \quad (5)$$

Заметим, что $|E| = |D|(1 - r)$ фактически определяет прогноз количества ошибок, которые может совершить обучаемый при выполнении предлагаемого задания.

В обучающем режиме при первом сеансе, когда $r = 0$, ресурс ошибок $|E_0| = |D_0|$. Система выделит ресурса для первого сеанса:

$$|Q_0| = 2|D_0|. \quad (6)$$

Если принять $|E_0| = 0$, то система с самого начала будет работать в институциональном режиме, заставляя пользователя выполнять только правильные операции.

Обучение заканчивается после k -го сеанса ($k \leq N$) при выполнении условия: $|1 - r_k| \leq \varepsilon_r$, где ε_r – малая положительная константа, $0 < \varepsilon_r \leq 1$.

Таким образом, алгоритм адаптивного управления формированием навыков решения задач без учета времени можно представить следующим образом:

1. Задать $\varepsilon_b, \varepsilon_r$; установить режим обучения: обучающий, контрольный
2. $i := 0, r_0 := 0, |D| := |D_0|, |E_0| := |D_0|$;
3. пока $|1 - r_i| > \varepsilon_r$
4. $i := i + 1$
5. если $r_i < \varepsilon_b$

- тогда $|Q_{i+1}| := |D_i| + |D_i|(1 - r_i)$
 иначе Выбрать новую задачу с трудоемкостью $|D_{i+1}|$,
 $|Q_{i+1}| := |D_{i+1}| + |D_{i+1}|(1 - r_i)$
6. Сеанс обучения i , получить d_i, e_i
 7. $h_i = (|D_i| - d_i) / |D_i|$
 если режим = обучающий
 и если $e_i = |E_i|$ тогда перейти в режим институционального управления
 иначе если режим = контрольный и если $e_i = |E_i|$ тогда закончить сеанс
 8. $r_i = d_i / (|D_i| + e_i)$
 9. конец цикла

В этом алгоритме неформализованным остался единственный пункт в шаге 5: выбрать новую задачу с трудоемкостью $|D_{i+1}|$. Формализация данного пункта существенно зависит от предметной области и решаемых в этой области задач и составляет отдельную тему для исследования.

Учет времени

Время рассматривается как ресурс, который выделяется системой в обучающих и контрольных сеансах. Пусть трудоемкость задания составляет $|D|$ операций, и для выполнения j -й операции требуется t_j единиц времени. Тогда общее операционное время выполнения задания равно:

$$T_0 = \sum_{j=1}^{|D|} t_j$$

В более простом варианте время выполнения всех операций считается одинаковым, и полное операционное время выполнения задания вычисляется как $T_0 = t |D|$. Заметим, что нормативное время выполнения задания должно быть несколько больше полного операционного времени T_0 , поскольку время расходуется не только на выполнение операции, но и на переход к следующей (например, на перемещение мышки от одной кнопки к другой требуется некоторое время).

Для первого обучающего сеанса система выделяет обучаемому минимальный временной ресурс в объеме T_0 единиц времени, а при завершении задания фиксирует фактическое время выполнения. Пусть обучаемый выполнит первое задание за фактическое время $t_0 \geq T_0$. Тогда на выполнение одной операции обучаемый в среднем затратит $s_0 = t_0 / q_0$ единиц времени. Второй и последующие сеансы завершаются при $t_i = T_i$, и система вычисляет величину

$$s_i = t_i / q_i = t_i / (|D_i| + e_i). \quad (7)$$

Эта величина представляет собой индивидуальный темп работы обучаемого при выполнении конкретного задания. Очевидно, что при отсутствии ошибочных действий величина s принимает минимальное значение $\min s = t / |D|$.

Обратная величина

$$v_i = q_i / t_i = (|D_i| + e_i) / t_i \quad (8)$$

представляет собой интенсивность (скорость) выполнения операций обучаемым при выполнении конкретного задания. Величины s_i и v_i представляют собой важные характеристики обучаемого, и сохраняются в модели обучаемого.

Ресурс времени для второго и последующих сеансов вычисляется по формуле:

$$T_{i+1} = s_i |Q_{i+1}|. \quad (9)$$

Таким образом, объем временного ресурса, выделяемого в сеансе, регулируется темпом работы обучаемого. Если обучаемый «медленный», то выделяется больше времени, если работает быстро – меньше.

С учетом времени система переходит в режим институционального управления либо при исчерпании ресурса ошибочных действий E , либо при исчерпании ресурса времени T .

Уровень обученности с учетом времени вычисляется следующим образом:

$$r_i = p(t_i)(d_i / (|D_i| + e_i)), \quad (10)$$

где $p(t)$ – функция снижения оценки при превышении нормативного времени. В работе [1] для данной функции получена формула $p(t) = e^{-\lambda t}$, а в работе [2] показано, что коэффициент $d = 1/\lambda$ представляет собой трудность задания. В данном случае под трудностью задания мы понимаем трудоемкость задания $|D|$, поэтому функция снижения оценки принимает вид:

$$p(t) = e^{-t/|D|}. \quad (11)$$

Пусть t_n – нормативное время, $t_n = (T_0 + t_e) \leq T_i$, где t_e – малая константа. Тогда превышение нормативного времени в i -м сеансе составит $t = (T_i - t_n)$, и эта величина подставляется в качестве аргумента в формулу (11).

Таким образом, модифицированный алгоритм формирования навыков решения задач с учетом времени можно представить следующим образом:

1. Задать ε_b , ε_r , установить режим обучения: обучающий, контрольный
2. $i := 0$, $r_0 := 0$, $|D| := |D_0|$, $|E_0| := |D_0|$, вычислить T_0
3. пока $|1 - r_i| > \varepsilon_r$
 1. $i := i + 1$
 2. если $r_i < \varepsilon_b$
тогда $|Q_{i+1}| := |D_i| + |D_i|(1 - r_i)$
иначе Выбрать новую задачу с трудоемкостью $|D_{i+1}|$,
 $|Q_{i+1}| := |D_{i+1}| + |D_{i+1}|(1 - r_i)$
 3. Сеанс обучения i , получить d_i , e_i , t_i
 4. Если режим = обучающий
и если if $e_i = |E_i|$ или $t_i = T_i$
тогда перейти в режим институционального управления
если режим = контрольный
и если $e_i = |E_i|$ или $t_i = T_i$
тогда закончить сеанс
 5. $q_i := d_i + e_i$, $s_i = t_i/q_i$
 6. $T_{i+1} := s_i |Q_{i+1}|$
 7. $r_i := p(t_i) d_i / (|D_i| + e_i)$
4. конец цикла

Информационно-энтропийный критерий уровня обученности

Выше формула (3) определяет достигнутый обучаемым уровень умений и навыков как долю успешных действий от множества учебных операций. В формуле (10) учитывается время выполнения задания. Однако возможен и другой подход к определению уровня обученности, позволяющий учесть не только успешные, но и ошибочные действия обучаемого – на основе вычисления энтропии.

Действие, совершаемое обучаемым при выполнении контрольного задания, будем считать случайным событием. Последовательность действий представляет собой поток случайных событий, обладающий следующими свойствами:

1. *Ординарность*. Вероятность совершения обучаемым двух или более действий за интервал времени Δt пренебрежимо мала по сравнению с вероятностью совершения за то же время ровно одного действия.
2. *Отсутствие последствий*. Вероятность совершения обучаемым некоторого количества действий на одном интервале времени не зависит от того, сколько действий обучаемый совершил в другом интервале времени.
3. *Стационарность*. На момент выполнения контрольного задания обучаемый обладает некоторым фиксированным уровнем обученности. Можно считать, что за время выполнения контрольного задания текущий уровень обученности, в том числе и достигнутый уровень навыков, остается постоянным.

Таким образом, последовательность действий, совершаемых обучаемым при выполнении контрольного задания, является пуассоновским потоком. Поскольку действие может быть правильным и неправильным, система контроля может пребывать в одном из двух состояний: либо в правильном состоянии, либо в состоянии ошибки. Следовательно, процесс, протекающий в контрольном режиме обучающей системы, является Марковским процессом [3], причем представляется моделью Гильберта с двумя состояниями (рис. 1).

Пусть S_i – правильное состояние, S_f – ошибочное состояние; p – вероятность перехода из правильного состояния в ошибочное, q – вероятность перехода из ошибочного состояния в правильное. Соответственно вероятность остаться в правильном состоянии составляет $1 - p$, веро-

ятность остаться в ошибочном состоянии равна $1 - q$, причем $p + q \neq 1$. Поскольку поток стационарный, существуют финальные вероятности пребывания системы в состояниях S_i и S_f :

$$p_i = q / (p + q); p_f = p / (p + q), \quad (12)$$

причем $p_i + p_f = 1$. Однако обучающая система имеет возможность вычислить эти вероятности непосредственно:

$$p_i = d / (d + e); p_f = e / (d + e), \quad (13)$$

где d – количество выполненных обучаемым правильных операций; e – количество совершенных ошибочных действий. Тогда можно вычислить энтропию системы:

$$H = -(p_i \log_2(p_i) + p_f \log_2(p_f)). \quad (14)$$

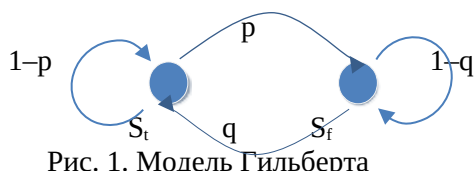


Рис. 1. Модель Гильберта

Энтропия позволяет определить интегрированный уровень обученности. В самом деле, в первом сеансе обучения, когда уровень обученности нулевой, вероятности $p_i = p_f = 0.5$, энтропия максимальна и равна 1. По мере обучения энтропия уменьшается, поэтому критерий обученности можно определить так:

$$r = 1 - H, \text{ при } p_i > p_f; r = H, \text{ при } p_i < p_f. \quad (15)$$

Отметим, что вероятности p_i и p_f можно вычислять не только по отдельному контрольному заданию, но и всему множеству действий, совершенных обучаемым в контрольном режиме при выполнении всех заданий. Таким образом, если обучаемый совершает большое количество ошибочных действий в нескольких заданиях, то его уровень обученности уменьшается.

Заключение

Предложенный подход предполагает, что обучающая система собирает полную информацию о действиях обучаемого во время сеансов обучения. Например, система может определять время между действиями, вычислять среднее время простоя, время получения помощи, собирать информацию об обращении обучаемого к подсистеме помощи, вычислять интенсивность обращения к подсистеме помощи и т. п. Накопление статистических данных позволит решить несколько важных задач:

- по собранным эмпирическим данным выявить индивидуальные кривые научения [4];
- построить индивидуальную модель ошибок обучаемого [5].

Совокупность разнообразных временных и количественных характеристик составляет индивидуальный профиль обучаемого. Знание типичных для студента характеристик позволяет, во-первых, улучшить адаптивное управление обучением, и во-вторых, повысить надежность контроля. Если в процессе сеанса обучения текущие характеристики обучаемого, собираемые системой, существенно отличаются от его статистических характеристик, это может говорить о применении студентом нечестных методов выполнения задания (например, задание выполняет другой студент по просьбе обучаемого).

Для реализации описанного подхода требуется, чтобы операции, необходимые для выполнения задания, были «крупноблочными». Например, редактор кода должен оперировать не отдельными символами, а конструкциями языка и объектами программы. Кроме того, система должна «знать», какова трудоемкость выполнения того или иного задания – в системе должны храниться эталонные результаты выполнения заданий [6].

В Астраханском государственном техническом университете на кафедре «Автоматизированные системы обработки информации и управления» разработана интегрированная среда программирования на основе семантического редактора [7], позволяющая вести учет всех действий студента. Описанный в статье метод оценивания умений и навыков реализуется в данной среде как один из методов адаптивного управления обучением. В настоящее время разработанная интегрированная среда проходит апробацию на кафедре АСОИУ. При изучении дисциплин

«Основы алгоритмизации» и «Алгоритмизация математических задач» студенты выполняют лабораторные работы в семантическом редакторе.

СПИСОК ЛИТЕРАТУРЫ

1. Сербин В.И. Методы оценки знаний в подсистеме тренинга обучающей системы // Вестник Астраханского государственного технического университета, №1(36)/2007. – с.247-251
2. Лаптев В.В., Сербин В.И. Учет времени при оценивании результатов автоматизированного контроля // Известия Волгоградского государственного технического университета: межвуз. сб. науч. ст. №11(71) / ВолгГТУ. – Волгоград: УИНЛ ВолгГТУ, 2010. – 160 с. – (Серия Актуальные проблемы управления, вычислительной техники и информатики в технических системах. Вып. 9). – с. 102-105
3. Феллер В. Введение в теорию вероятностей и ее приложения: Т.1. – М.: Книжный дом «Либроком», 2010. – 528 с.
4. Новиков Д.А. Закономерности итеративного научения. – М.: Институт проблем управления РАН, 1998. – 77 с.
5. Атанов Г.А., Пустынникова И.Н. Обучение и искусственный интеллект, или Основы современной дидактики высшей школы. – Донецк: Изд-во ДООУ, 2002. – 504 с.
6. Лаптев В.В. Требования к современной обучающей среде по программированию // Объектные системы-2010 (Зимняя сессия): материалы II Международной научно-практической конференции. Россия, Ростов-на-Дону, 10-12 ноября 2010 г. / Под общ. Ред. П.П. Олейника. – Ростов-на-Дону, 2010. – с. 104-110.
7. Семантический редактор автоматизированной обучающей системы по программированию. – Свидетельство об официальной регистрации программ для ЭВМ № 2011619172, Россия, Д.А. Грачев, В.В. Лаптев – зарегистрировано 25.11.2011.

УДК 004.023 + 004.588

В.В. Лаптев

МЕТОД ОЦЕНИВАНИЯ УМЕНИЙ И НАВЫКОВ ПРИ ОБУЧЕНИИ ПРОГРАММИРОВАНИЮ

Реферат

В статье рассматривается проблема оценивания умений и навыков в автоматизированной системе обучения. Предлагаются алгоритмы интерактивного обучения с учетом нормативного времени выполнения задания и без учета времени. Навыки оцениваются как доля успешных действий обучаемого при выполнении задания. Построена формальная модель интерактивного обучения в виде марковского процесса с двумя состояниями. Критерий обученности вычисляется как энтропия системы.

Ключевые слова: обучающая система, уровень обученности, институциональное управление, трудоемкость задания, нормативное время, энтропия

V.V. Laptev

МЕТОД ОЦЕНИВАНИЯ УМЕНИЙ И НАВЫКОВ ПРИ ОБУЧЕНИИ ПРОГРАММИРОВАНИЮ THE METHOD FOR THE SKILLS RATING DURING TRAINING IN PROGRAMMING

Abstract

This article describe a problem of the skills rating in automated training system. The algorithms of interactive education with and without time control are considered. The skills can be rated as a part of the successful actions during the task execution. The formal model of interactive education is built as a Markov process with two states. The level of competence can be rated as a system entropy.

Keywords: training system, level of competence, institutional control, task difficulty, normative time, entropy

Сведения об авторах

<i>Сведения</i>	<i>Русский вариант</i>	<i>Английский вариант</i>
Фамилия	Лаптев	Laptev
Имя	Валерий	Valery
Отчество	Викторович	Victorovich
Ученая степень	Кандидат технических наук	Candidate of Technical Sciences
Ученое звание	Доцент	
Место работы	АГТУ	ASTU
Подразделение	Кафедра Автоматизированные системы обработки информации и управления	Department of the automated systems of information processing and management
Должность	Доцент	Assistant Professor
Служебный телефон	(8512)614-109	
Эл. почта	Laptev@ilabsltd.com	