

МОДЕЛИ И АЛГОРИТМЫ СИСТЕМЫ ОБУЧЕНИЯ И ОЦЕНКИ ЗНАНИЙ

Криводубский А. Ю.

*Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь*

Серебряная Л.В. – канд. техн. наук

Современные методы и алгоритмы позволяют автоматизировать процессы обучения и оценки знаний. Наиболее эффективным методом оценки знаний является тестирование с помощью открытых вопросов. В данной работе рассмотрены существующие подходы к реализации систем оценки ответов на открытые вопросы, а также предложен вариант реализации такой системы. Рассмотрены грамматические правила извлечения сущностей из текстов и предложен алгоритм анализа ответов.

Процесс обучения в любом образовательном учреждении подразумевает проверку усвоения учащимися полученных знаний. Данная проверка важна для обратной связи и последующей корректировки процесса обучения.

Одним из эффективных видов проверки знаний обучаемых является тестирование.

Наиболее эффективным методом тестирования является тестирование с помощью открытых вопросов – где отвечающий дает развернутый или короткий ответ на естественном языке. При таком виде тестирования практически полностью исключается возможность угадывания правильного ответа, а от студента требуется умение не просто вспомнить термин или определение, но и корректно сформулировать свои мысли.

Существующие системы тестирования с помощью открытых вопросов можно разделить на несколько категорий, в зависимости от используемых методов работы с текстом.

К первой категории относятся системы, основанные на сопоставлении концепций (Concept Mapping). В таких системах, как правило, ответы тестируемого и эталонные ответы разбиваются на некоторые списки ключевых понятий, так называемых минимальных концепций, и либо подсчитывается общее количество концепций для выставления оценки, либо во время оценивания рассматривается только одна каким-то образом выбранная концепция. Каждой концепции можно приписать некий вес, и тогда в итоговой оценке учитываются эти веса.

Вторую категорию образуют системы, основанные на методах извлечения информации (Information Extraction Systems). Здесь, как правило, происходит сопоставление неким шаблонам, при этом используются регулярные выражения или деревья разбора. В результате из неструктурированных текстов извлекается информация в виде структурированных данных. Ответ может разбиваться на сегменты, и происходит оценивание каждого сегмента. Для каждого вопроса можно сформулировать более одного шаблона. Предлагаемая в данной работе система также принадлежит к этой категории.

Третью категорию составляют системы, основанные на использовании корпусов (корпусом в лингвистике называется собранный и обработанный по определенным правилам набор текстов, используемый в качестве базы для исследования языка). Обычно корпуса применяются для работы с большими текстами, но подобный метод можно применять и при анализе коротких ответов, как правило, с использованием эталонного ответа в качестве словаря для ограничения правильных ответов.

В четвертую категорию входят системы, применяющие машинное обучение. При этом обычно используются разные метрики, взятые из методов обработки естественного языка. Они либо комбинируются, либо происходит оценивание с помощью одной из имеющихся классификационных или регрессионных моделей.

К пятой категории относятся системы, не попадающие под предыдущие категории, или в которых используются комбинированные методы.

Отличительной особенностью систем тестирования с помощью открытых вопросов является сильная зависимость от языка текстов вопросов и ответов. Подавляющее большинство существующих систем не поддерживают работу с русским языком. В связи с этим является актуальной разработка новой системы тестирования.

Основа любой подобной системы – лингвистический процессор (ЛП) – приложение, осуществляющее лингвистический анализ системы. Разработка хорошего ЛП для русского языка – чрезвычайно трудоемкая задача. Поэтому для построения новой системы был выбран уже существующий и хорошо зарекомендовавший себя ЛП для работы с русскими текстами разработки компании Яндекс – Томита-парсер. Он принадлежит к категории извлечения информации, и поэтому создаваемая система анализа открытых ответов тоже принадлежит к этой категории.

Как правило, в ЛП анализ текста включает в себя следующие этапы:

1. Графематический анализ (сегментация, токенизация и т. д.).
2. Морфологический анализ (нормализация, стемминг, частеречная разметка и т. д.).
3. Предсинтаксический анализ.
4. Синтаксическая сегментация.
5. Синтаксический анализ.
6. Семантический анализ (хотя до сих пор нет универсальных математических моделей и вообще формальных средств описания смысла слов).

Анализ ответов в системе происходит по следующему алгоритму. Из базы данных берется очередной ответ тестируемого на какой-нибудь вопрос, он переводится в текстовый файл и подается на вход Томита-парсера. Томита-парсер анализирует его и записывает результаты (список фактов) в XML-файл. Этот XML-файл затем парсится, и получается список фактов в удобном для программы виде. Томита-парсер позволяет выделять из текста на русском языке сущности, которые называются фактами. Выделение происходит с помощью написанных пользователем шаблонов (или грамматических правил), наборы таких правил называются грамматиками. Факты можно представить как некие таблицы с колонками, которые называются полями фактов (например, факт «собрание» может иметь поля «место», «время» и «тема»).

В ходе разработки системы были написаны грамматические правила, которые с помощью Томита-парсера затем были переведены в собственные правила, для извлечения 11 фактов (шести фактов, описывающих части речи (имя существительное, глагол, имя прилагательное, причастие, частица, предлоги и союзы), трех фактов, описывающих члены предложения (подлежащее, сказуемое, дополнение), и двух фактов, описывающих специфичные для предметной области тестирования термины и названия).

По выделенным грамматикам системой был получен список списков фактов ответа (список списков, т. к. может существовать несколько экземпляров одного и того же факта). Например, если в предложении три существительных и два глагола, то на выходе будет получен список «Существительное» из трех фактов и список «Глагол» из двух фактов. Далее, для того же самого вопроса в базе данных находятся все эталонные ответы. Для них проводится такая же процедура по получению списка списков фактов.

После запускается алгоритм сравнения: список фактов предложения поочередно сравнивается с каждым списком из набора списков фактов соответствующих эталонных ответов.

Сравнение происходит так: в списке для каждого из фактов необходимо найти долю правильности. Доля правильности считается следующим образом: берется список, соответствующий данному факту, и для каждого члена этого списка вначале ищется совпадение с членами списка этого же факта в списке эталонных ответов. Если совпадения не нашлось, то в словаре синонимов находятся все синонимы для этого члена, и потом определяется, нет ли среди этих синонимов членов списка этого же факта в списке эталонных ответов.

Каждому члену списков фактов A_i можно сопоставить пару чисел (m, s) (от match и synonym), где $m = 1$, если A_i совпадает с одним из эталонных ответов; $m = 0$, если A_i не совпадает ни с одним из эталонных ответов; $s = 1$, если есть синоним для A_i , совпадающий с одним из эталонных ответов; $s = 0$ – нет синонима для A_i , совпадающего с одним из эталонных ответов. Возможные значения пар (m, s) – $(1,0)$, $(0,1)$ и $(0,0)$.

Суммируем все явные совпадения ($\sum m$) и совпадения с учетом синонимов ($\sum s$) – это будет числитель для f . В знаменателе будет стоять количество членов списка этого же факта в списке эталонов – обозначим это количество через σ . Таким образом, f будет равно единице, если для каждого члена списка данного факта списка эталонного ответа найдется либо точное совпадение в списке данного факта тестируемого списка, либо синоним. В общем случае, f всегда принимает значения в интервале $[0;1]$. Также обрабатываются случаи, когда данного факта вообще нет в эталонном ответе: когда числитель равен нулю, или когда тестируемый ввел несколько синонимов для одного факта в своем ответе, и знаменатель стал больше числителя.

Оценка ответа также лежит в интервале $[0;1]$. Далее нужно проделать точно такую же процедуру для следующего эталонного ответа и т. д. Итоговая оценка будет максимальной из всех оценок для всех эталонных ответов.

Анализ результатов тестирования показал, что система хорошо справляется с короткими ответами по критерию совпадения результатов оценки системы и преподавателя, если и исходный эталонный ответ не велик, но испытывает некоторые сложности при оценивании больших (порядка 40–50 слов) предложений.

Таким образом, предложенная система оценки знаний может быть доработана для практического применения в учебных заведениях различной направленности.

Список использованных источников:

1. Burrows S. *The eras and trends of automatic short answer grading* / Burrows S., Gurevych I., Stein B. // *International Journal of Artificial Intelligence in Education*, 2015. – Vol. 25, P. 60-117.

2. Burstein J. *Using lexical semantic techniques to classify free-responses* / Burstein J., Kaplan R., Wolff S., Lu C. // *Proc. of the ACL SIGLEX Workshop on Breadth and Depth of Semantic Lexicons*, 1996 – P. 20-29.

3. Horbach A. *Using the text to evaluate short answers for reading comprehension exercises* / Horbach A., Palmer A., Pinkal M. // *Proc. of the 2nd Joint Conf. on Lexical and Computational Semantics*. Atlanta, USA: Association for Computational Linguistics, 2013 – Vol. 1, P. 286-295.