

- Пропажа собак является серьезной проблемой, и существует потребность в эффективных инструментах для поиска пропавших собак. Информационная система может предоставить удобный и надежный способ получения информации о собаках, а также обеспечить возможность эффективного поиска пропавших питомцев, что отвечает актуальным потребностям владельцев собак.

- Развитие технологий и цифровизация общества предоставляют уникальные возможности для создания и использования такой информационной системы, обеспечивая удобство, доступность и оперативность взаимодействия между владельцами собак и медицинскими специалистами.

Заключение

В заключение, информационная система по подбору, уходу и воспитанию собак, а также поиску пропавших питомцев представляет собой ценный ресурс для владельцев и потенциальных владельцев собак. Она обеспечивает доступ к обширной информации о различных породах, помогает принять осознанное решение при выборе питомца, предоставляет рекомендации и советы по уходу и воспитанию, а также помогает в поиске пропавших собак.

База данных для информационной системы по подбору, уходу и воспитанию собак, а также поиску пропавших питомцев, является важным шагом в обеспечении безопасности, благополучия и счастья собак и их владельцев. Она помогает создать доверительные и ответственные отношения между людьми и животными, способствуя укреплению связи между ними.

ГЕНЕРАЦИЯ СБАЛАНСИРОВАННОГО КОМПЛЕКТА БИЛЕТОВ ДЛЯ КОНТРОЛЯ ЗНАНИЙ УЧАЩИХСЯ

Сущинская В. А. (студент гр. ИТИ-42)

Гомельский государственный технический университет имени П.О.Сухого, Гомель, Республика Беларусь

Научный руководитель – Захаренко Владимир Сергеевич

(к.т.н., доцент кафедры «Информационные технологии» ГГТУ им. П.О. Сухого)

Аннотация: В данной работе рассматривается проблема формирования сбалансированного комплекта экзаменационных билетов (тестов) и её решение с помощью нескольких алгоритмов. В итоге был выбран наиболее эффективный алгоритм для решения данной проблемы.

Ключевые слова: экзаменационный билет, тест, алгоритм, баланс, объем, сложность.

Введение

Задача формирования экзаменационных билетов, обеспечивающих сбалансированный уровень сложности и объема вопросов, представляет собой значимую проблему в образовательном процессе. Традиционные методы ручного составления билетов требуют значительных затрат времени преподавателя, а существующие программные средства для генерации билетов, в свою очередь, не всегда способны обеспечить сбалансированный уровень сложности и объема вопросов, что может привести к несправедливой оценке знаний учащихся.

Поэтому создание приложения для автоматизации формирования билетов, с учетом сложности и объема вопросов, является актуальной задачей [1].

Результаты и обсуждение

Для данного приложения было разработано и проанализировано три алгоритма. Также было протестировано применение генетического алгоритма для решения данной задачи.

Введем следующие обозначения: пусть n – количество вопросов в билете (тесте), d – сложность вопроса, $\langle d \rangle$ – средняя сложность всех вопросов, v – объем вопроса, $\langle v \rangle$ – средний объем всех вопросов, i – номер вопроса в банке вопросов, j – номер билета (теста),

k – номер вопроса в билете (тесте), m – количество билетов (тестов).

Первый алгоритм формирования сбалансированного комплекта билетов по двум критериям состоит из трёх основных шагов:

- Добавление первых $n - 1$ вопросов в билет.
- Удаление этих вопросов из общего списка вопросов.
- Поиск наиболее подходящего последнего вопроса по данным двум критериям.

Подробнее рассмотрим третий шаг. Введем обобщенный показатель сложности и объема вопросов билета φ_j , по которому будем оценивать степень оптимальности добавления i -го вопроса в j -ый билет. Чем ближе значение φ_j к нулю, тем более подходящим является вопрос для включения в j -ый билет.

$$\varphi_j = \left| \langle d \rangle \cdot n - \sum_{k=1}^{n-1} d_k - d_i \right| + \left| \langle v \rangle \cdot n - \sum_{k=1}^{n-1} v_k - v_i \right|, \varphi_j \rightarrow 0 \quad (1)$$

В итоге, просматриваются все вопросы в текущем списке вопросов и осуществляется поиск наиболее подходящего вопроса с минимальным значением φ_j по формуле (1).

Второй алгоритм состоит из двух основных шагов:

- Сортировка списка вопросов.

Для каждого вопроса вычисляется величина ω по формуле (2). И по этой величине вопросы сортируются по возрастанию.

$$\omega_i = (\langle d \rangle - d_i) + (\langle v \rangle - v_i) \quad (2)$$

2) Оценка количества вопросов в билете n (задаются пользователем).

Если n – четное число, то для каждого билета берется первый и последний вопрос поочередно в текущем списке вопросов.

Если n – нечётное число, то берется $n - 1$ вопросов по принципу первый и последний вопрос поочередно в текущем списке вопросов. А последний вопрос в билете ищется по алгоритму поиска наиболее подходящего вопроса, описанному в первом алгоритме.

Третий алгоритм является комбинацией первого и второго алгоритма, где осуществляется поиск $n - 2$ подходящих вопросов и идет сравнение каждого вопроса с каждым.

Для определения подходящего алгоритма для использования в приложении был проведен анализ эффективности. Для этого были вычислены $\langle \theta_d \rangle$ по формуле (3) и $\langle \theta_v \rangle$ по формуле (4) – средние отклонения по сложности и объему соответственно для каждого алгоритма.

$$\langle \theta_d \rangle = \frac{\sum_{j=1}^m \left| \langle d \rangle \cdot n - \sum_{k=1}^n d_k \right|}{m}, \langle \theta_d \rangle \rightarrow 0 \quad (3)$$

$$\langle \theta_v \rangle = \frac{\sum_{j=1}^m \left| \langle v \rangle \cdot n - \sum_{k=1}^n v_k \right|}{m}, \langle \theta_v \rangle \rightarrow 0 \quad (4)$$

Графики оценки отклонения по сложности и объему для алгоритмов представлены на рис. 1.

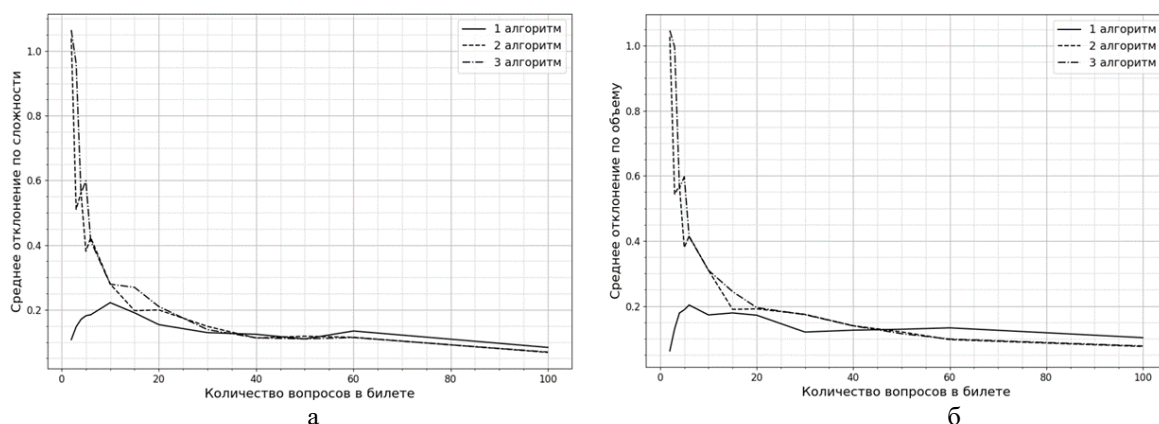


Рис. 1. Графики оценки отклонения по сложности и объему: а – зависимость $\langle \theta_d \rangle$ от количества вопросов в билете; б – зависимость $\langle \theta_v \rangle$ от количества вопросов в билете

Исходя из графика на рис. 1а можно сделать вывод, что на начальном участке до 30 вопросов в билете (тесте) первый алгоритм значительно превосходит второй и третий по данной величине. На отрезке от 30 до 60 вопросов алгоритмы приблизительно выравниваются, а после этого первый алгоритм начинает уступать второму и третьему.

Результат графика на рис. 1б схож с отклонением по сложности. На отрезке до 45 вопросов первый алгоритм превосходит второй и третий, далее же картина меняется и первый алгоритм начинает уступать второму и третьему, которые в свою очередь идут друг с другом вровень.

Генетический алгоритм [2] для решения данной задачи оказался неэффективным.

Заключение

Из данного эксперимента следует, что первый алгоритм подходит лучше для формирования сбалансированного комплекта билетов (тестов) с количеством вопросов от 2 до 40 в одном билете (тесте), что вполне подходит для внедрения в образовательный процесс университета. И он был использован для разработки приложения.

Литература

3. Формирование экзаменационных билетов [Электронный ресурс]. – Режим доступа: <https://technology.snauk-a.ru/2015/04/6412>. – Дата доступа: 01.02.2024.

4. Генетический алгоритм [Электронный ресурс]. – Режим доступа: https://ru.wikipedia.org/wi-ki/Генетический_алгоритм. – Дата доступа: 06.02.2024.

ТЕНЗОРЫ В МАШИННОМ ОБУЧЕНИИ [МИНИ ОБЗОР]

Тимошенко И. Ю. (студент гр. ИТИ-21)

Гомельский государственный технический университет имени П.О. Сухого, Гомель,
Республика Беларусь

Научный руководитель – Карась О. В.

(ассистент кафедры «Информационные технологии» ГГТУ им. П.О. Сухого)

Аннотация: в работе проводится обзор использования тензоров в машинном обучении, анализе текста, изображений, видеозаписей и других видов информации.

Ключевые слова: тензоры, многомерные массивы, искусственный интеллект.

Введение

Тензоры играют ключевую роль в современном машинном обучении, обеспечивая эффективную обработку и представление данных. В данной работе рассматривается значимость тензоров в контексте машинного обучения и их разнообразное применение в различных задачах анализа данных.

Как правило, чтобы привести компьютер к полезной работе, необходимо явно задать правила – программу, которой он должен следовать, чтобы преобразовать входные данные в нужные ответы. Но машинное обучение переворачивает этот подход: здесь машина изучает входные данные и соответствующие ответы, выясняя самостоятельно, какие правила следует