Интернет-журнал «Отходы и ресурсы» <https://resources.today>
Russian Journal of Resources, Conservation and Recycling

2020, №3, Том 7 / 2020, No 3, Vol 7 <https://resources.today/issue-3-2020.html>

URL статьи: <https://resources.today/PDF/01INOR320.pdf>

DOI: 10.15862/01INOR320 (<http://dx.doi.org/10.15862/01INOR320>)

Ссылка для цитирования этой статьи:

Резникова К.М., Савченко Д.В., Смышляева А.А. Разработка программы для диагностирования заболеваний методом тестового распознавания // Интернет-журнал «Отходы и ресурсы», 2020 №3, <https://resources.today/PDF/01INOR320.pdf> (доступ свободный). Загл. с экрана. Яз. рус., англ. DOI: 10.15862/01INOR320

For citation:

Reznikova K.M., Savchenko D.V., Smyshlyaeva A.A. (2020). Development of a program for diagnosing diseases by the test recognition method. *Russian Journal of Resources, Conservation and Recycling*, [online] 3(7). Available at: <https://resources.today/PDF/01INOR320.pdf> (in Russian) DOI: 10.15862/01INOR320

УДК 007.5

ГРНТИ 28.23.29

Резникова Ксения Михайловна

ФГОУ ВО «Дальневосточный федеральный университет», Владивосток, Россия

Магистрант

E-mail: a-da_97@mail.ru

Савченко Денис Валерьевич

ФГОУ ВО «Дальневосточный федеральный университет», Владивосток, Россия

Магистрант

E-mail: savchenko.dv@students.dvfu.ru

Смышляева Анна Андреевна

ФГОУ ВО «Дальневосточный федеральный университет», Владивосток, Россия

Магистрант

E-mail: anyac957@mail.ru

Разработка программы для диагностирования заболеваний методом тестового распознавания

Аннотация. Теория распознавания образов является важным теоретическим и прикладным направлением в информатике. В зависимости от природы поставленной задачи для распознавания используют различные подходы. В данной статье авторами исследуется текущее состояние применения суперкомпьютеров в области медицинской диагностики и предлагается доступное прикладное решение для диагностирования заболеваний, основанное на применении тестового распознавания.

Тестовое распознавание основано на таком комбинаторно-логическом подходе, как тестовый алгоритм. Тестовые алгоритмы основываются на анализе множества тупиковых тестов таблицы и зависят от размерности заданной матрицы объектов и их признаков. Тупиковым тестом считается несжимаемый набор признаков, содержащий все сведения о разделении таблицы на классы.

В работе кратко представлена хронология развития тестовых алгоритмов для решения задач распознавания.

Авторами разработана интерпретация тестового алгоритма для автоматизации диагностирования заболеваний путем выявления близости искомого подмножества (имеющихся симптомов) к одному из уже существующих (диагнозов) посредством определения тупиковых тестов.

В дополнение к представленному алгоритму представлены скриншоты с примером пошагового решения в виде консольного приложения, созданного на языке высокоуровневого программирования C#.

На основе разработанного алгоритма авторами предложен пример готового программного решения в виде оконного приложения для диагностирования заболеваний по имеющимся у пациента симптомам. Разработанное программное решение позволяет отметить имеющиеся у пациента симптомы и автоматически рассчитать наиболее вероятное заболевание. Программа сужает область допустимых значений (диагнозов) в зависимости от отмеченных пользователем симптомов и выдает оценку возможных заболеваний, помогая врачу в постановке диагноза и сводя к минимуму ошибки, допустимые человеческим фактором.

В статье приведена оценка достоинств и недостатков разработанного решения, рассмотрены другие области применения тестового распознавания.

Ключевые слова: распознавание; тест; тестовый алгоритм; тупиковый тест; минимальный тест; тестовое распознавание; классификация; диагностирование; модель голосования

Введение

На сегодняшний день возможности суперкомпьютеров в медицине имеют огромный потенциал, обладая невероятными вычислительными мощностями, но, несмотря на все предоставляемые возможности и пользу обществу, их внедрение остается под большим вопросом.

Суперкомпьютер от Сбербанка – Кристофари – позиционируется как самый мощный суперкомпьютер в России и готов к сдаче в аренду по базовой цене в 5,7 тыс. рублей за минуту. Также Сбербанк запустит тариф, в котором за 100 рублей клиент получит мощности суперкомпьютера на 100 часов, но сможет обработать не больше 10 Гб данных [1]. В действительности для медицинских нужд этого объема недостаточно, поскольку количество данных пациентов огромно и все время растет. И, если текстовые данные могут весить относительно немного, то графические, такие как рентгеновские снимки, результаты кардиограмм и т. д. требуют обработки в огромных объемах. В России суперкомпьютеры на данный момент используются только в Российской академии наук (РАН) и в образовательных процессах технических высших учебных заведений. Так, суперкомпьютер Санкт-Петербургского политехнического университета Петра Великого (СПбПУ), названный «Политехник – РСК Торнадо», используется для решений междисциплинарных задач естественно-научного профиля и поддержки проектирования сложных технических систем для высокотехнологичных и наукоемких областей науки и промышленности, а межведомственный суперкомпьютер РАН применяется исследователями для решения фундаментальных и прикладных задач в сфере науки и образования¹. Также Балтийский государственный технический университет «Военмех» им Д.Ф. Устинова создал высокопроизводительный

¹ РСК лидирует среди Российских производителей суперкомпьютеров в рейтинге TOP50 [Электронный ресурс] // РСК. URL: <http://www.rscgroup.ru/ru/news/383-rsk-lidiruet-sredi-rossiyskih-proizvoditeley-superkompyuterov-v-reytinge-top50> (дата обращения: 14.07.2020).

кластер, обеспечивающий противодействие существующим средствам воздушно-космического нападения [2]. И, хотя в связи со сложившейся эпидемиологической обстановкой суперкомпьютер МГУ ищет лекарство от коронавируса [3], в российских медицинских учреждениях их еще не используют и вряд ли появится такая возможность в ближайшем будущем. В поисках информации, хранящейся в открытом доступе, об их внедрении ничего не обнаружено. Однако в качестве альтернативного помощника в диагностировании заболеваний медицинскому сотруднику может помочь дополнительное программное средство, основанное на таком методе искусственного интеллекта, как тестовое распознавание.

Распознавание образов является научной дисциплиной, целью которой служит классификация объектов по некоторым категориям – классам. Эти объекты называются образами. Классификация объектов основывается на прецедентах. Прецедент – образ, у которого известна правильная классификация. Все объекты разбиты на конечное число непересекающихся классов, для каждого изучено и известно конечное число объектов – прецедентов. Таким образом, задача распознавания состоит в том, чтобы отнести новый распознаваемый объект к какому-либо классу. Для классификации образов используются измерения, называемые признаками. Признак является некоторым количественным измерением объекта произвольной природы [4].

Теория тестового распознавания

В 60-е годы были проведены исследования и среди алгоритмов комбинаторно-логического подхода важную роль сыграли именно тестовые алгоритмы. Тестовые алгоритмы базируются на работе с заданной таблицей, столбцы которой соответствуют признакам объектов, а каждая строка служит набором значений признаков, описывающим один из объектов. Строки таблицы, в свою очередь, разбиты на непересекающиеся классы.

Тестовые алгоритмы основаны на анализе множества тупиковых тестов таблицы. Тупиковым тестом является несжимаемый набор признаков, который содержит все сведения о разделении таблицы на классы. Обычно таких избыточных наборов признаков много и процесс их построения очень трудоемкий. Поэтому тестовый алгоритм распознавания существенно зависит от алгоритмов построения тупиковых тестов [5].

В.Б. Кудрявцев и А.Е. Андреев формулируют задачу распознавания следующим образом: даны матрица T' и ее подматрицы $T_{i'}$, необходимо указать оператор ϕ , который вычисляет состояние объекта A по этим подматрицам и заданному набору признаков. Состояние объекта A представлено предъявленным набором. Очевидно, что поиск ϕ требует различных допущений относительно свойств A . К таким допущениям относятся алгебраические, вероятностные и геометрические свойства A , которые породили соответствующие направления в теории распознавания [6].

И.А. Чегис и С.В. Яблонский заметили, что можно осуществлять такое распознавание, используя не всю матрицу T , а лишь ее часть. Они предложили следующую формулировку понятия теста для T : пусть σ это набор признаков, а $T_{i,\sigma}$ это часть таблицы T_i , образованная столбцами, соответствующими σ , а T_σ – все $T_{i,\sigma}$. Набор σ образует тест для T , если у $T_{i,\sigma}$ и $T_{i',\sigma}$ нет общих строк при $i \neq i'$. В результате тест σ после проведения анализа множества T_σ и набора α_σ способен самостоятельно решать задачу распознавания. Тест в роли эксперта принимает решение по части набора α и матрицы T [7].

Позднее В.И. Переяславский и А. Шайбеба решили использовать каждый тест σ из множества T' для α , тем самым получив вектор «голосов» $\chi(\alpha) = (k_1, \dots, k_m, k_{m+1})$, где k_i –

количество тестов, которые высказались за T_i при $i \neq 1, 2, \dots, m$, а k_{m+1} – количество «воздержавшихся» [6].

Такой вектор «голосов» характерен для моделей вычисления оценок (голосования) – Г-модели. Модели голосования основаны на принципе прецедентности, т. е. они анализируют «близость» между ранее классифицированными объектами и теми объектами, которые необходимо распознать. Если близость существует, она служит частичным прецедентом и оценивается посредством числовой оценки. По набору таких оценок близости формируется общая оценка распознаваемого объекта для конкретного класса, которая и будет определять принадлежность искомого описания объекта к объекту существующего класса [8].

Несмотря на то, что функционал χ был опробован в решении ряда прикладных задач, обнаружили некоторые трудности его использования. Основную сложность представила сложность и слабое согласование результатов с реальностью (для отдельных задач распознавания), что привело к уходу от голосования по всему множеству тестов, заменив его на особые семейства – тупиковые, короткие и минимальные тесты, а также их ограничения.

Таким образом, тупиковый тест представляет собой такой тест, у которого никакое собственное подмножество признаков не образует теста. Короткий тест – это тест, который по числу признаков близок к логарифму числа признаков в матрице T . К минимальному тесту относится наименьшее количество признаков из всех возможных [6].

Восстанавливая хронологию, стоит обратить внимание на то, что первые решенные Ю.И. Журавлевым и его коллегами задачи имели матрицу с количеством строк и столбцов всего до 20. Для масштабных задач эта матрица очень мала, поэтому научное сообщество принялось активно заниматься теорией тестового распознавания для расширения его возможностей. Так, работа В.Е. Кузнецова дала возможность решать задачи нахождения тупиковых тестов в матрице до 100 строк и столбцов. Е.В. Дюкова смогла расширить решения задач до матриц с 200 столбцами и 50 строками, а А.Е. Андреев – до матриц с 200 столбцами и 200 строками. А.А. Кибкало достиг использования матриц с 500 столбцами и 500 строками. Сегодня современные компьютеры позволяют на порядок и более увеличить размеры обрабатываемых матриц [9].

Также В.Б. Кудрявцев и А.Е. Андреев [6] в своих работах освещают примеры тестовых алгоритмов для матриц с малым числом строк и для матриц с заданным графом сравнения. От выбранного метода (алгоритма) построения тупиковых тестов зависит результат распознавания.

Разработанный алгоритм

Целью разработанного алгоритма является выявление близости некоторого подмножества к одному из существующих подмножеств посредством определения тупиковых тестов.

Порядок действий:

1. Поиск теста, близкого к минимальному:

А. Для исходной матрицы построить элементарную предикатную матрицу (ЭПМ):

$$T_{m,n}^l \rightarrow M$$

В. Упорядочить по убыванию столбцы ЭПМ по числу единиц:

$$M \rightarrow M'$$

С. Имя первого столбца P' зафиксировать в качестве первого элемента искомого множества. Поставить соответствие зафиксированному P' из таблицы $T_{m,n}^l$ элемент X' :

$$P' \rightarrow X'$$

Д. Вычеркнуть из элементарной предикатной матрицы зафиксированный столбец P' и все строки, у которых в этом столбце зафиксированы единичные компоненты:

$$M' \rightarrow M''$$

Е. Перейти к пункту В.

Г. Выполнять шаги до тех пор, пока не будут вычеркнуты все строки.

2. Поиск всех тупиковых тестов путем раскрытия скобок:

А. Для исходной матрицы построить элементарную предикатную матрицу:

$$T_{m,n}^l \rightarrow M$$

В. Строки ЭПМ упорядочить по числу единичных компонентов по возрастанию:

$$M \rightarrow M'$$

С. В ЭПМ вычеркнуть все строки-расширения по отношению к оставшимся:

$$M' \rightarrow M''$$

Строкой расширения i -ой строки элементарной предикатной матрицы является такая строка j -ая строка, которая совпадает с i -ой строкой по единичным компонентам и содержит большее число единиц.

Д. На множестве строк ЭПМ (M'') необходимо построить выражение:

$$\Pi \Sigma (\&, \vee);$$

Е. Преобразовать $\Pi \Sigma \Rightarrow \Sigma \Pi$

Г. Рассчитать информационные веса.

Информационным весом i -ого признака называется отношение k_i к N :

$$R_i = \frac{k_i}{N},$$

где k_i – число вхождений i -ого признака найденного множества тупиковых тестов, а N – число всех обнаруженных тупиковых тестов. Максимальный вес равен 1, минимальный – 0.

В системах, основанных на тестовом распознавании, актуальна задача вычисления информационных весов признаков (весовых коэффициентов). В своей работе А.Е. Янковская и С.И. Колесникова [10] на основе методов получения весовых коэффициентов признаков рассматривают меру важности как отношение числа вхождений признака во все безизбыточные (тупиковые) тесты к числу всех безизбыточных тестов. Авторы находят этот подход логичным, интуитивно прозрачным и легко интерпретируемым. Данный метод используется для расчета меры важности в разрабатываемой программе.

Полученные результаты

В качестве небольшого примера на рисунках 1–3 приведен результат работы алгоритма в консольном приложении. Первым шагом является ввод параметров задачи распознавания (рисунок 1).

```
I. Укажите параметры задачи :  
Количество строк : 6  
Количество столбцов : 7  
Исходная матрица <подмножество> <значение> ... :  
a 1 1 0 1 1 0 0  
b 0 1 1 0 1 1 1  
c 0 0 1 1 1 0 1  
d 1 0 1 1 0 1 1  
e 0 1 0 0 1 1 0  
f 1 0 1 1 0 1 1  
Введите значения множества Q : 1 0 1 0 0 1 1
```

Рисунок 1. Ввод параметров задачи (разработано авторами)

Для наглядности взят небольшой пример с 6 строками и 7 столбцами. Пользователю предлагается ввести исходную матрицу в формате «<подмножество> <значение>» и искомое в ней множество Q. Задачей программы стоит поиск принадлежности множества Q к описанию признаков одного из введенных выше подмножеств: a, b, c, d, e, f.

На рисунке 2 пошагово изображен поиск тупиковых тестов.

```
II. Поиск тупиковых тестов :  
Элементарная предикатная матрица M :  
1 0 1 1 0 1 1  
1 1 1 0 0 0 1  
0 1 1 0 1 1 1  
1 0 0 1 0 1 0  
0 1 1 0 1 1 1  
0 1 0 1 0 1 0  
1 1 0 1 1 0 0  
0 0 1 0 0 0 1  
1 1 0 1 1 0 0  
1 0 0 0 1 1 0  
0 1 1 1 0 1 1  
1 0 0 0 1 1 0  
1 1 1 1 1 0 1  
0 0 0 0 0 0 0  
1 1 1 1 1 0 1  
Отсортированная матрица M' :  
0 0 0 0 0 0 0  
0 0 1 0 0 0 1  
1 0 0 1 0 1 0  
0 1 0 1 0 1 0  
1 0 0 0 1 1 0  
1 0 0 0 1 1 0  
1 1 1 0 0 0 1  
1 1 0 1 1 0 0  
1 1 0 1 1 0 0  
1 0 1 1 0 1 1  
0 1 1 0 1 1 1  
0 1 1 0 1 1 1  
0 1 1 1 0 1 1  
1 1 1 1 1 0 1  
1 1 1 1 1 0 1  
Матрица M'' :  
0 0 0 0 0 0 0  
0 0 1 0 0 0 1  
1 0 0 1 0 1 0  
0 1 0 1 0 1 0  
1 0 0 0 1 1 0  
1 0 0 0 1 1 0  
1 1 1 0 0 0 1  
1 1 0 1 1 0 0  
1 1 0 1 1 0 0  
1 0 1 1 0 1 1  
0 1 1 0 1 1 1  
0 1 1 0 1 1 1  
0 1 1 1 0 1 1  
1 1 1 1 1 0 1  
1 1 1 1 1 0 1  
Тупиковые тесты P :  
<P3+P7> <P1+P4+P6> <P2+P4+P6> <P1+P5+P6> <P1+P5+P6> <P1+P2+P4+P5> <P1+P2+P4+P5>  
=  
= <P3+P4> <P3+P6> <P7+P1> <P7+P4> <P7+P6> <P3+P1+P2> <P2+P1> <P2+P5> <P4+P1> <P  
4+P5> <P6+P2> <P6+P4>
```

Рисунок 2. Поиск тупиковых тестов (разработано авторами)

Все начинается с поиска теста, близкого к минимальному, как указано в пунктах 1.A–1.D предложенного выше алгоритма. Выполнив все шаги алгоритма, кроме 2.F, приложение по матрице M'' выводит все обнаруженные тупиковые тесты – в данном примере их 12.

Последним шагом является проверка множества Q на принадлежность к существующему классу и расчет информационных весов R для каждого теста.

Соответствуя модели голосования, приложение выводит сумму «голосов», тем самым определяя максимальное соответствие искомого множества с реальным описанием объектов. На рисунке 3 показано соответствие множества Q классам d и f , но, поскольку d встречается раньше, то программа выбирает именно его.

```
III. Проверка множества Q :  
Информационные веса R :  
R1 = 4/12  
R2 = 4/12  
R3 = 3/12  
R4 = 5/12  
R5 = 2/12  
R6 = 4/12  
R7 = 3/12  
Результат :  
a b c d e f  
x3x4 0 1 0 0 0 0  
x3x6 0 1 0 1 0 1  
x2x1 0 0 0 1 0 1  
x2x4 0 1 0 0 0 0  
x2x6 0 1 0 1 0 1  
x3x1x2 0 0 0 1 0 1  
x2x1 0 0 0 1 0 1  
x2x5 0 0 0 1 0 1  
x4x1 0 0 0 0 0 0  
x4x5 0 0 0 0 0 0  
x6x2 0 0 0 1 0 1  
x6x4 0 1 0 0 1 0  
Сумма: 0 5 0 7 1 7  
Множество d принадлежит Q
```

Рисунок 3. Проверка множества Q (разработано авторами)

В рамках исследуемой темы подмножество b тоже бы соответствовало диагнозу, но с меньшей вероятностью.

Оконное приложение

Пример рабочего варианта для диагностирования заболеваний в виде оконного приложения представлен на рисунке 4. Для наглядности взято несколько заболеваний и характерных для них симптомов. По мере выбора имеющихся в наличии симптомов у пациента программа рассчитывает наиболее соответствующий диагноз и менее вероятный (при его наличии).

Form1

Диагностирование лор-заболеваний

Сбросить

Симптомы пациента :

- ☒ Боль в горле / Отек горла
- ☐ Наличие гиперемии
- ☐ Охриплость / Потеря голоса
- ☐ Воспаление миндалин
- ☒ Наличие сухого кашля
- ☒ Ощущение кома в горле / Першение
- ☐ Увеличенные лимфоузлы
- ☐ Заложенность носа / Отечность
- ☐ Продолжительность насморка
- ☐ Боль при наклоне / Повышенная чувствительность лица

Коэффициенты диагностики* :

Фарингит. K = 24
Простуда/ОРВИ. K = 4

*Чем выше коэффициент диагностики, тем выше вероятность наличия указанного заболевания

Диагноз пациента :

Фарингит

Рисунок 4. Оконное приложение, расчет 1 (разработано авторами)

Выбрав в качестве симптомов боль в горле, наличие сухого кашля и ощущение кома в горле, алгоритм выводит предположение, что пациент болен фарингитом. Также алгоритм выдает оценку вероятности другого диагноза – ОРВИ. Таким образом, можно наблюдать перечень всех возможных заболеваний, не упуская их из виду.

При добавлении/удалении галочек, тут же меняются коэффициенты диагностики и результирующий диагноз. Так, на рисунке 5 можно наблюдать другое «предположение» программы при добавлении нового симптома.

Рисунок 5. Оконное приложение, расчет 2 (разработано авторами)

Можно предположить, что расчеты будут точнее при матрице гораздо больших размеров, ведь с увеличением описаний объектов класса уменьшается порог вхождения в «чужое» подмножество (мнение субъективно).

Заключение

Очевидным достоинством данного решения является быстрота расчетов. Расширив приложение для работы с заболеваниями других направлений (или всех сразу), можно проверять студентов медицинских учреждений и сотрудников на знание недугов в своей области. Если есть возможность модифицировать функционал установленных в конкретной больнице информационных систем, такой помощник мог бы оказаться очень удобным.

Главным недостатком выступает сложность понимания тестовых алгоритмов для последующей реализации из-за отсутствия какой-либо современной информации.

Помимо медицинской диагностики разработанная интерпретация тестового алгоритма может быть эффективна и в других областях, где необходимо распознавание по набору определенных признаков, например, для обнаружения технических неполадок или в геологических исследованиях.

Современные технологии не стоят на месте, искусственный интеллект и кибернетика активно развиваются и находят свое место в новых областях. И, хотя технологии не всегда и везде могут заменить человека, они могут обеспечить ему хотя бы поддержку в принятии важных решений.

ЛИТЕРАТУРА

1. Святослав Иванов. Сбербанк представил самый мощный в России суперкомпьютер «Кристофари» [Электронный ресурс] // Хайтек. URL: <https://hightech.fm/2019/11/08/cristofer> (дата обращения: 14.07.2020).
2. Щеглов Д.К., Федоров Д.А. Опыт внедрения суперкомпьютерных технологий в образовательный процесс технического вуза // Образование и инновации. 2018. №1 (231).
3. Арсений Скрынников. «Ломоносов» против COVID-19: как суперкомпьютер МГУ ищет лекарство от коронавируса // RT на русском. URL: <https://ru.rt.com/frd1> (дата обращения: 14.07.2020).
4. Местецкий Л.М. Математические методы распознавания образов: курс лекций. – МГУ, ВМиК, кафедра «Математические методы прогнозирования», 2002–2004.
5. Мадатян Х.А. О тупиковых тестах для бинарных таблиц // Журнал вычислительной математики и математической физики. – 1997. – Том 37, №2. – с. 243–249.
6. Кудрявцев В.Б., Андреев А.Е. Тестовое распознавание // Фундаментальная и прикладная математика. – М.: Издательский дом «Открытые системы». – 2009. – Том 15, №4. – с. 67–99.
7. Чегис И.А., Яблонский С.В. Логические способы контроля электрических схем // Тр. МИАН СССР. – 1958. – Том 51. – с. 270–360.
8. Журавлев Ю.И. Об алгебраических методах в задачах распознавания и классификации. Распознавание, классификация, прогноз // Математические методы и их применение – М.: Наука. – 1989. – Том 1. – с. 9–16.
9. Кудрявцев В.Б., Андреев А.Е., Гасанов Э.Э. Теория тестового распознавания. – М.: ФИЗМАТЛИТ, 2007. – 320 с.
10. Янковская А.Е., Колесникова С.И. Сравнительный анализ методов получения весовых коэффициентов признаков в задаче распознавания образов // Математические методы распознавания образов (ММРО-11). – 2003. – №94. – с. 227–229.

Reznikova Kseniya Mikhailovna

Far Eastern federal university, Vladivostok, Russia
E-mail: a-da_97@mail.ru

Savchenko Denis Valerievich

Far Eastern federal university, Vladivostok, Russia
E-mail: savchenko.dv@students.dvfu.ru

Smyshlyaeva Anna Andreevna

Far Eastern federal university, Vladivostok, Russia
E-mail: anyac957@mail.ru

Development of a program for diagnosing diseases by the test recognition method

Abstract. Theory of pattern recognition is an important theoretical and applied area in computer science. Depending on the nature of the task for recognition are used different approaches. In this article the authors investigate the current state of the use of supercomputers in the field of medical diagnostics and propose an affordable applied solution for diagnosing diseases based on the use of test recognition.

Test recognition is based on such a combinatorial-logical approach as a test algorithm. Test algorithms are based on the analysis of a set of deadlock table tests and depend on the dimension of a given matrix of objects and their features. A deadlock test is an incompressible set of features that contains all the information about dividing a table into classes.

In this work briefly presents the chronology of the development of test algorithms for solving recognition problems.

The authors have developed an interpretation of the test algorithm to automate the diagnosis of diseases by identifying the proximity of the desired subset (available symptoms) to one of the existing ones (diagnoses) by defining deadlock tests.

In addition to the presented algorithm, screenshots are presented with an example of a step-by-step solution in the form of a console application created in the high-level programming language C#.

On the basis of the developed algorithm, the authors propose an example of a ready-made software solution in the form of a window application for diagnosing diseases based on the patient's symptoms. The developed software solution allows to mark the patient's symptoms and automatically calculate the most probable disease. The program narrows the range of acceptable values (diagnoses) depending on the symptoms marked by the user and provides an assessment of possible diseases, helping the doctor in making a diagnosis and minimizing errors made by the human factor.

The article provides an assessment of the advantages and disadvantages of the developed solution, considers other areas of application of test recognition.

Keywords: recognition; test; test algorithm; deadlock test; minimal test; test recognition; classification; diagnosing; voting model