

Додаток А

Програмний код

```

using System;
using System.Windows.Forms;

namespace AdaptiveSystemDemo
{
    public partial class Form1 : Form
    {
        int quest;
        int clickAmount;
        int deactiveCount;
        public Form1()
        {
            InitializeComponent();

            quest = 1;
            clickAmount = 0;
            deactiveCount = 0;

            button1.Enabled = true;
            button1.Text = "Продовжити";
            label10.Text = "Зроблено кліків миші: 0";
            label11.Text = "Еталонна кількість кліків: " + Questions.Length
* 2; // В ідеалі, кліків не повинно бути більше за 1 вибір варіанта та 1
клік по кнопці "Продовжити" на кожному питанні.
            label12.Text = "Кількість зайвих кліків: 0";
            dataGridView1.Rows.Clear();

            CleanSelections();
            CleanButtonTags();
            InitQuestions();
            InitTimer();
        }

        // Таймер
        DateTime startTime;
        TimeSpan span;

        private void InitTimer()
        {
            Timer timer = new Timer();
            timer.Interval = 500;
            timer.Enabled = true;
            timer.Tick += new EventHandler(timer_Tick);
            startTime = DateTime.Now;
        }

        void timer_Tick(object sender, EventArgs e)
        {
            span = DateTime.Now - startTime;
            label13.Text = "Час: " + span.ToString(@"m\:ss");
        }
    }
}

```

```

        // Перевірка тексту на його прочитання
        float minWordCount = Questions[quest - 1][0].Split(new char[] {
' ' }, StringSplitOptions.RemoveEmptyEntries).Length +
radioButton1.Text.Split(new char[] { ' ' },
StringSplitOptions.RemoveEmptyEntries).Length;
        float fullWordCount = minWordCount +
radioButton2.Text.Split(new char[] { ' ' },
StringSplitOptions.RemoveEmptyEntries).Length + radioButton3.Text.Split(new
char[] { ' ' }, StringSplitOptions.RemoveEmptyEntries).Length;
        float minTimeToRead = minWordCount / 378f * 60f + 1f; // 1 -
кількість додаткового часу на прийняття рішення
        float maxTimeToRead = fullWordCount + 5f; // 5 - кількість
додаткового часу на прийняття рішення
        label7.Text = "(Час менший за " +
(int)Math.Round(minTimeToRead) + " сек.)";
        label9.Text = "(Час більший за " +
(int)Math.Round(maxTimeToRead) + " сек.)";
        if (maxTimeToRead > span.TotalSeconds && span.TotalSeconds >
minTimeToRead)
        {
            radioButtonR11.Checked = false;
            radioButtonR12.Checked = true;
        }
        else if (span.TotalSeconds > maxTimeToRead)
        {
            radioButtonR12.Checked = false;
            radioButtonR13.Checked = true;
        }
    }
}

// Питання
string[][] Questions = new string[5][]; // Кількість питань можна
змінювати, але для прикладу обрано 5

private void InitQuestions()
{
    // Для прикладу питання введено вручну:
    Questions[0] = Question("Lorem ipsum dolor sit amet,
consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et
dolore magna aliqua. Tortor posuere ac ut consequat?", "Невірна відповідь",
"Вірна відповідь", "Невірна відповідь", "2");
    Questions[1] = Question("Tortor posuere ac ut consequat?",
"Вірна відповідь", "Невірна відповідь", "Невірна відповідь", "1");
    Questions[2] = Question("Sit amet luctus venenatis lectus. Sed
vulputate mi sit amet. Quisque id diam vel quam elementum pulvinar etiam?",
"Невірна відповідь", "Невірна відповідь", "Вірна відповідь", "3");
    Questions[3] = Question("Iaculis eu non diam phasellus
vestibulum lorem sed risus ultricies. Nec nam aliquam sem et tortor
consequat id porta. Tellus cras adipiscing enim eu. At imperdiet dui
accumsan sit amet nulla facilisi?", "Невірна відповідь", "Вірна відповідь",
"Невірна відповідь", "2");
    Questions[4] = Question("Sed viverra ipsum nunc aliquet
bibendum enim facilisis gravida. Vel pharetra vel turpis nunc eget lorem
dolor sed. Bibendum at varius vel pharetra. Dictumst vestibulum rhoncus est
pellentesque elit ullamcorper dignissim?", "Вірна відповідь", "Невірна
відповідь", "Невірна відповідь", "1");
    ShowQuestion(1);
}

```

```

    }

    public string[] Question(string question, string answer1, string
answer2, string answer3, string rightAnswer)
    {
        string[] s = new string[5];
        s[0] = question;
        s[1] = answer1;
        s[2] = answer2;
        s[3] = answer3;
        s[4] = rightAnswer;
        return s;
    }

    private void ShowQuestion(int number)
    {
        number = quest;
        int questAmount = Questions.Length; // Підрахунок загальної
кількості питань
        label2.Text = "Питання: " + number + "/" + questAmount;
        richTextBox1.Text = Questions[number - 1][0]; // Вивід питання
        radioButton1.Text = Questions[number - 1][1]; // Вивід
варіантів відповіді
        radioButton2.Text = Questions[number - 1][2];
        radioButton3.Text = Questions[number - 1][3];
        if (Questions[number - 1][4] == "1") radioButton1.Tag =
"Correct"; // Назначення правильної відповіді
        else if (Questions[number - 1][4] == "2") radioButton2.Tag =
"Correct";
        else if (Questions[number - 1][4] == "3") radioButton3.Tag =
"Correct";
        else
        {
            MessageBox.Show("Вірну відповідь не було вказано.\nЇї
тимчасово встановлено на перший варіант.");
            radioButton1.Tag = "Correct";
        }

        // Зміна вигляду кнопки
        if (quest >= questAmount)
        {
            button1.Text = "Закінчити тест";
        }
    }

    private void CleanButtonTags()
    {
        radioButton1.Tag = "";
        radioButton2.Tag = "";
        radioButton3.Tag = "";
        radioButton1.Checked = false;
        radioButton2.Checked = false;
        radioButton3.Checked = false;
    }

    private void CleanSelections()
    {
        radioButtonR11.Checked = true;
    }

```

```

        radioButtonR12.Checked = false;
        radioButtonR13.Checked = false;
    }

    // Кнопка "Продовжити"
    private void button1_Click(object sender, EventArgs e)
    {
        ClickHandler();
        DataGridViewRow row =
(DataGridViewRow)dataGridView1.Rows[0].Clone();
        row.Cells[0].Value = quest; // Перенесення у результати номеру
питання
        if (radioButton1.Checked) // Перенесення у результати стану
відповіді
        {
            if (radioButton1.Tag == "Correct") row.Cells[1].Value =
"Вірно";
            else row.Cells[1].Value = "Невірно";
        }
        else if (radioButton2.Checked)
        {
            if (radioButton2.Tag == "Correct") row.Cells[1].Value =
"Вірно";
            else row.Cells[1].Value = "Невірно";
        }
        else if (radioButton3.Checked)
        {
            if (radioButton3.Tag == "Correct") row.Cells[1].Value =
"Вірно";
            else row.Cells[1].Value = "Невірно";
        }
        // Перенесення у результати стану читання тексту
        if (radioButtonR11.Checked) row.Cells[2].Value = "Його
пропущено";
        else if (radioButtonR12.Checked) row.Cells[2].Value = "Hi";
        else if (radioButtonR13.Checked) row.Cells[2].Value = "На ньому
затримались";
        row.Cells[3].Value = span.ToString(@"m:ss");
        dataGridView1.Rows.Add(row);
        // Перехід до наступного питання
        int questAmount = Questions.Length;
        if (quest < questAmount)
        {
            quest++;
            CleanButtonTags();
            CleanSelections();
            InitQuestions();
            InitTimer();
        }
        else if (quest >= questAmount)
        {
            button1.Enabled = false;
        }
    }

    // Кількість кліків миші
    private void ClickHandler()
    {

```

```

        clickAmount++;
        label10.Text = "Зроблено кліків миші: " +
clickAmount.ToString();
        int extraClicks = clickAmount - Questions.Length * 2;
        if (extraClicks > 0) label12.Text = "Кількість зайвих кліків: "
+ extraClicks;
    }

    private void radioButton1_Click(object sender, EventArgs e)
    {
        ClickHandler();
    }

    private void radioButton2_Click(object sender, EventArgs e)
    {
        ClickHandler();
    }

    private void radioButton3_Click(object sender, EventArgs e)
    {
        ClickHandler();
    }

    private void richTextBox1_Click(object sender, EventArgs e)
    {
        ClickHandler();
    }

    // Кількість разів, коли вікно програми ставало неактивним
    private void Form1_Deactivate(object sender, EventArgs e)
    {
        deactiveCount++;
        label15.Text = "Вікно втрачало фокус: " + deactiveCount;
    }
}
}

```

Додаток Б
Слайди презентації

АТЕСТАЦІЙНА РОБОТА МАГІСТРА

ТЕМА: «ДОСЛІДЖЕННЯ МОДЕЛЕЙ ПОБУДОВИ МЕТРИК
АДАПТАЦІЇ НАВЧАЛЬНИХ МАТЕРІАЛІВ В СИСТЕМАХ
ЕЛЕКТРОННОГО НАВЧАННЯ»

Виконав:
Студент групи ІПЗм-17-2
Папай О. О.

Керівник проекту:
проф. Шубін Ю. І.

2

Актуальність роботи

- ▶ Дистанційне навчання – популярний засіб отримання нових знань людьми різного віку
- ▶ Адаптивні системи тільки нещодавно стали активно з'являтися і загальноприйнятих методів їх розробки ще не існує
- ▶ Неадаптивні системи сильно поступають традиційному очному навчанню у якості отримуваної освіти через складнощі з універсальністю подачі матеріалу та проблеми з етапом перевірки знань
- ▶ Навіть при очному навчанні складно отримати персональний шлях навчання, що враховуватиме особистості кожного учня

3

АНАЛІЗ СТАНУ РОЗВ'ЯЗАННЯ ПРОБЛЕМИ

Існуючі інтерактивні навчальні системи

<p>Плюси:</p> <ul style="list-style-type: none"> - Схожість процесу отримання знань з традиційним <p>Мінуси:</p> <ul style="list-style-type: none"> - Непристосовано до використання у автономному режимі - Немає автоматичної системи зашкодження шахраювання під час контролю знань 	Назва інтерактивної навчальної системи	Ступінь освіти	Адаптування матеріалу	Контроль знань	
		Відкриті університети	+	± (через зв'язок з викладачем)	+
		Віртуальні університети	±	± (за наявності зв'язку з автором)	±
		Навчальні платформи	-	-	-

4

АНАЛІЗ СТАНУ РОЗВ'ЯЗАННЯ ПРОБЛЕМИ

Системи контролю знань

<p>Плюси:</p> <ul style="list-style-type: none"> - Корисний інструмент очного контролю знань - Складність програмного втручання <p>Мінуси:</p> <ul style="list-style-type: none"> - Не надають матеріалу - Відсутність автоматичної системи зашкодження шахраюванню 	Назва системи контролю знань	Гнучкість	Використання медіа	Корисність у автономному режимі	
		OpenTEST	+	-	-
		На основі Flash	+	+	-

5

АНАЛІЗ СТАНУ РОЗВ'ЯЗАННЯ ПРОБЛЕМИ

Порівняння неадаптивних і адаптивних систем

	Неадаптивні системи	Адаптивні системи
Адаптивність матеріалу	±	+
Адаптивність контролю знань	-	+
Автономне використання	-	+
Персональний підхід до кожного користувача	±	+
Запобігання шахраюванню	-	+
Оцінювання в реальному часі	-	+

6

АНАЛІЗ СТАНУ РОЗВ'ЯЗАННЯ ПРОБЛЕМИ

Адаптивні системи на основі інструментів штучного інтелекту

	Системи на основі машинного навчання	Системи на основі складних алгоритмів	Системи на основі правил	Системи на основі дерев рішень
Легкість розробки	-	±	+	+
Низька вартість підтримки	-	±	+	+
Самостійна зміна методів викладання	+	±	-	-
Профілі учнів	+	+	-	-
Можливість працювати без зв'язку з інтернетом	-	±	+	+

Постановка задачі

- ▶ Проаналізувати існуючі адаптивні і неадаптивні системи у пошуку засобів досягнення програмної адаптивності матеріалу у навчальних системах
- ▶ Проаналізувати існуючі технології, що роблять адаптацію можливою
- ▶ Проаналізувати існуючі метрики, що можуть бути використані для реалізації адаптивності
- ▶ Проаналізувати які дані можуть збиратись і обробляться адаптивною навчальною системою, щоб на їх основі розробляти метрики
- ▶ У результаті повинні вийти метрики, що дозволять створити програмну реалізацію адаптивної навчальної системи.

Технології, що роблять адаптацію можливою

Алгоритми машинного навчання:

- ▶ штучні нейронні мережі;
- ▶ кластерний аналіз;
- ▶ тощо.

Графи:

- ▶ ланцюги Маркова;
- ▶ мережі Петрі;
- ▶ тощо.

Метрики

Метрики

- ▶ Використання метрик підійде усім видам розташування адаптивної навчальної системи: і онлайн, і локально.
- ▶ Машинному навчанню необхідна велика кількість користувачів, щоб постійно отримувати нову інформацію і на її основі самовдосконалюватись. Метрикам – ні.
- ▶ Метрики допоможуть добитись найкращих результатів, якщо використовувати для підрахунків та адаптації додаткову інформацію різних типів даних.
- ▶ Метрики, корисні у розробці адаптивних навчальних систем було розбито на два типи: кількісні і якісні.

Методи розробки метрик

	Кількісні	Якісні
Вхідні дані: кількісні	+	+
Вхідні дані: якісні	±	+
Вихідні дані: кількісні	+	-
Вихідні дані: якісні	-	+
Корисність у розрахунках	+	-
Корисність у програмуванні	+	+

11

Наукова складова роботи

- ▶ Ціллю роботи є дослідження і вдосконалення існуючих засобів програмного аналізу навчальними системами рівня знань своїх користувачів
- ▶ Результатом роботи стали метрики, що дозволяють реалізувати процеси адаптації матеріалу без застосування складних алгоритмів та інструментів машинного навчання, а також автоматичні методи запобігання шахраюванню під час проходження етапу контролю знань

12

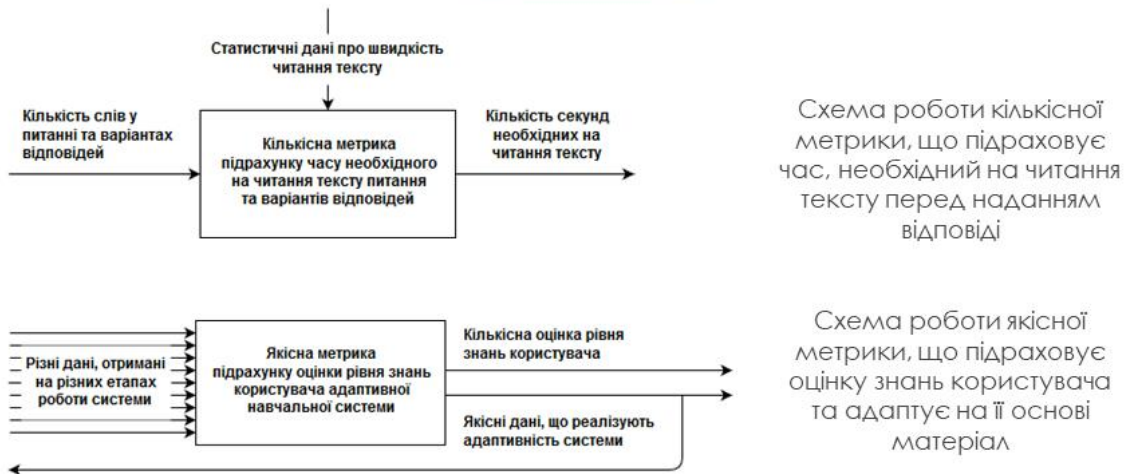
Реалізація алгоритму

- ▶ Швидкість читання середньостатистичного читача становить 120-200 слів на хвилину. Крайніми значеннями є 60 при уважному читанні та 378 слів на хвилину при скорочитанні.
- ▶ Максимальний час, за який питання має бути прочитане:
 $T_{max} = x + 5$, де x – кількість слів у питанні та всіх варіантах відповідей, а 5 – додаткові секунди, необхідні на прийняття рішення.
- ▶ Мінімальний час, за який питання має бути прочитане:
 $T_{min} = y * 6.3 + 1$, де y – кількість слів тільки у запитанні та першому варіанту відповіді, а 1 – секунда, необхідна на прийняття рішення.
- ▶ При наданні відповіді на питання, програма збирає різні дані за допомогою якісної метрики *Answer(questNum, isCorrect, isRead, time, questTheme, questSubtheme, finalMark)* та на їх основі реалізує процеси адаптивності.

13

Реалізація алгоритму

Схеми



14

Опис розробленої системи

- ▶ Розроблена система є системою на основі правил
- ▶ Система симулює етап проходження контролю знань
- ▶ Адаптація відбувається, базуючись на інформації отриманій під час відповідей на питання у контролі знань
- ▶ Деяка інформація отримується у режимі реального часу, деяка під час відповіді на кожне питання, деяка тільки після проходження контролю

15

Корисність результатів

- ▶ Отримані результати будуть корисними при розробці універсальних адаптивних навчальних систем, що потребують чи не потребують онлайн зв'язку з профілями користувачів.
- ▶ Адаптивні навчальні системи можуть покращити або ж замінити собою існуючі інструменти дистанційного навчання.
- ▶ Метрики та типи адаптивних навчальних систем, що досліджувались, будуть цікаві як інженерам програмного забезпечення, так і дослідникам штучного інтелекту.

16

Демонстрація програми

Adaptive System DEMO

ТЕСТ

Час: 0:04 Питання: 1/5

Питання:

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Tortor posuere ac ut consequat?

Варіанти відповідей:

Невірна відповідь

Врна відповідь

Невірна відповідь

РЕЗУЛЬТАТ

Чи прочитано текст перед наданням відповіді?

Ні, його пропущено (Час менший за 5 сек.)

Так (Час у допустимому діапазоні)

Так, але дій не предпринято (Час більший за 35 сек.)

Активність кліків миші

Зроблено кліків миші: 0

Еталонна кількість кліків: 10

Кількість зайвих кліків: 0

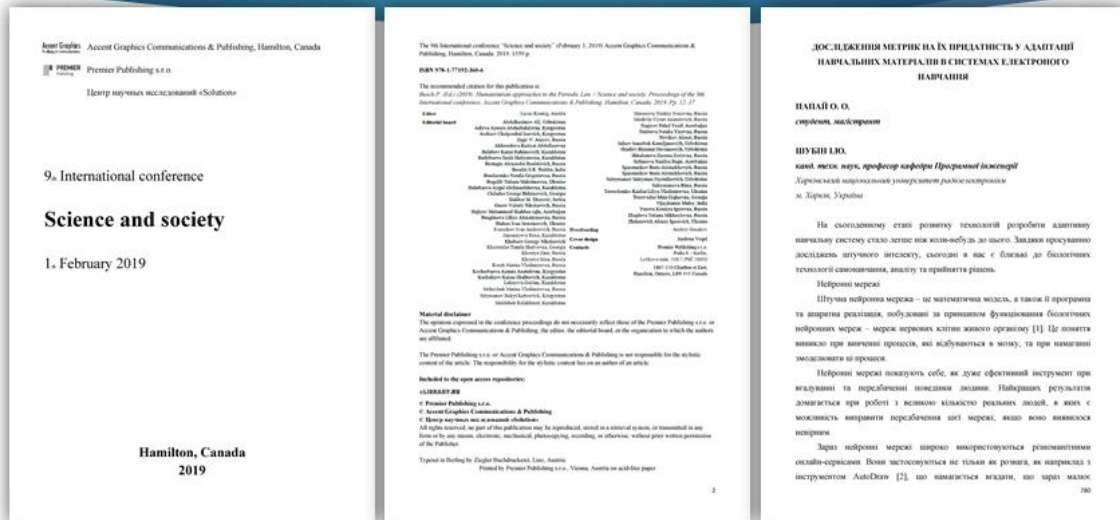
Активність вікна програми

Вікно втрачало фокус: 0

Результати

№	Стан відповіді	Чи були проблеми з текстом	Час
*			

Апробація матеріалу




Результати атестаційної роботи було опубліковано у IX Міжнародній науковій конференції «Science and Society», що відбулась 1 лютого 2019 р. у м. Гамільтон, Канада.

ВИСНОВКИ

- ▶ У результаті виконання атестаційної роботи було проаналізовано всі відомі типи неадаптивних навчальних систем. Було знайдено їх переваги перед очним отриманням освіти, а також їх недоліки.
- ▶ Було проаналізовано та розбито на типи всі відомі типи адаптивних навчальних систем. Було знайдено переваги та недоліки кожного з типів.
- ▶ Метрики було проаналізовано на придатність у адаптації в системах електронного навчання. А також було проаналізовано типи даних, які можуть використовуватись цими метриками.
- ▶ Було вдосконалено існуючі засоби програмного аналізу навчальними системами рівня знань своїх користувачів
- ▶ У результаті було розроблено програмну реалізацію адаптивної навчальної системи, що використовує як чисельні, так і якісні метрики у своїй роботі. Програмно реалізовано запобігання шахраюванню під час проходження контролю знань, а також, отримання даних, на основі яких можна реалізувати адаптацію матеріалу у реальному або близькому до реального часі.

Додаток В
Апробація результатів роботи

 Accent Graphics Communications & Publishing, Hamilton, Canada

 Premier Publishing s.r.o.

Центр научных исследований «Solution»

9th International conference

Science and society

1st February 2019

**Hamilton, Canada
2019**

The 9th International conference “Science and society” (February 1, 2019) Accent Graphics Communications & Publishing, Hamilton, Canada. 2019. 1359 p.

ISBN 978-1-77192-360-6

The recommended citation for this publication is:

Busch P. (Ed.) (2019). Humanitarian approaches to the Periodic Law // Science and society. Proceedings of the 9th International conference. Accent Graphics Communications & Publishing. Hamilton, Canada. 2019. Pp. 12–17

Editor	Lucas Koenig, Austria	Morozova Natalay Ivanovna, Russia
Editorial board	Abdulkasimov Ali, Uzbekistan	Moskvin Victor Anatolevich, Russia
	Adieva Aynura Abduzhalalovna, Kyrgyzstan	Nagiyev Polad Yusif, Azerbaijan
	Arabaev Cholponkul Isaevich, Kyrgyzstan	Naletova Natalia Yurevna, Russia
	Zagir V. Atayev, Russia	Novikov Alexei, Russia
	Akhmedova Raziya Abdullayevna	Salaev Sanatbek Komiljanovich, Uzbekistan
	Balabiev Kairat Rahimovich, Kazakhstan	Shadiev Rizamat Davranovich, Uzbekistan
	Barlybaeva Saule Hatiyatovna, Kazakhstan	Shhahutova Zarema Zorievna, Russia
	Bestugin Alexander Roaldovich, Russia	Soltanova Nazilya Bagir, Azerbaijan
	Boselin S.R. Prabhu, India	Spasennikov Boris Aristarkhovich, Russia
	Bondarenko Natalia Grigorievna, Russia	Spasennikov Boris Aristarkhovich, Russia
	Bogolib Tatiana Maksimovna, Ukraine	Suleymanov Suleyman Fayzullaevich, Uzbekistan
	Bulatbaeva Aygul Abdimazhitovna, Kazakhstan	Suleymanova Rima, Russia
	Chiladze George Bidzinovich, Georgia	Tereschenko-Kaidan Liliya Vladimirovna, Ukraine
	Dalibor M. Elezović, Serbia	Tsersvadze Mzia Giglaevna, Georgia
	Gurov Valeriy Nikolaevich, Russia	Vijaykumar Muley, India
	Hajiyev Mahammad Shahbaz oglu, Azerbaijan	Yurova Kseniya Igorevna, Russia
	Ibragimova Liliya Ahmatyanovna, Russia	Zhaplova Tatiana Mikhaylovna, Russia
	Blahun Ivan Semenovich, Ukraine	Zhdanovich Alexey Igorevich, Ukraine
	Ivannikov Ivan Andreevich, Russia	Andrey Simakov
	Jansarayeva Rima, Kazakhstan	Andreas Vogel
	Khubaev Georgy Nikolaevich	Premier Publishing s.r.o.
	Khurtsidze Tamila Shalvovna, Georgia	Praha 8 – Karlín,
	Khoutyz Zaur, Russia	Lyčkovo nám. 508/7, PSČ 18600
	Khoutyz Irina, Russia	1807-150 Charlton st East,
	Korzh Marina Vladimirovna, Russia	Hamilton, Ontario, L8N 3x3 Canada
	Kocherbaeva Aynura Anatolevna, Kyrgyzstan	
	Kushaliyev Kaiser Zhalitovich, Kazakhstan	
	Lekerova Gulsim, Kazakhstan	
	Melnichuk Marina Vladimirovna, Russia	
	Meymanov Bakyt Kattoevich, Kyrgyzstan	
	Moldabek Kulakhmet, Kazakhstan	
	Proofreading	
	Cover design	
	Contacts	

Material disclaimer

The opinions expressed in the conference proceedings do not necessarily reflect those of the Premier Publishing s.r.o. or Accent Graphics Communications & Publishing, the editor, the editorial board, or the organization to which the authors are affiliated.

The Premier Publishing s.r.o. or Accent Graphics Communications & Publishing is not responsible for the stylistic content of the article. The responsibility for the stylistic content lies on an author of an article.

Included to the open access repositories:

eLIBRARY.RU

© Premier Publishing s.r.o.

© Accent Graphics Communications & Publishing

© Центр научных исследований «Solution»

All rights reserved; no part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without prior written permission of the Publisher.

Typeset in Berling by Ziegler Buchdruckerei, Linz, Austria.

Printed by Premier Publishing s.r.o., Vienna, Austria on acid-free paper

ДОСЛІДЖЕННЯ МЕТРИК НА ЇХ ПРИДАТНІСТЬ У АДАПТАЦІЇ НАВЧАЛЬНИХ МАТЕРІАЛІВ В СИСТЕМАХ ЕЛЕКТРОНОГО НАВЧАННЯ

ПАПАЙ О. О.

студент, магістрант

ШУБІН І.Ю.

канд. техн. наук, професор кафедри Програмної інженерії

Харківський національний університет радіоелектроніки

м. Харків, Україна

На сьогоденному етапі розвитку технологій розробити адаптивну навчальну систему стало легше ніж коли-небудь до цього. Завдяки просуванню досліджень штучного інтелекту, сьогодні в нас є близькі до біологічних технології самонавчання, аналізу та прийняття рішень.

Нейронні мережі

Штучна нейронна мережа – це математична модель, а також її програмна та апаратна реалізація, побудовані за принципом функціонування біологічних нейронних мереж – мереж нервових клітин живого організму [1]. Це поняття виникло при вивченні процесів, які відбуваються в мозку, та при намаганні змоделювати ці процеси.

Нейронні мережі показують себе, як дуже ефективний інструмент при вгадуванні та передбаченні поведінки людини. Найкращих результатів домагається при роботі з великою кількістю реальних людей, в яких є можливість виправити передбачення цієї мережі, якщо воно виявилось невірним.

Зараз нейронні мережі широко використовуються різноманітними онлайн-сервісами. Вони застосовуються не тільки як розвага, як наприклад з інструментом AutoDraw [2], що намагається вгадати, що зараз малює

користувач; Deep Dream [3], що працює із зображеннями та редагує їх, основуючись на своїх алгоритмах розпізнання; так і, наприклад, у таких інструментах як Google Translate [4], що після введення у дію нейронних мереж став перекладати набагато вірніше.

Але, через необхідність постійно отримувати новий досвід для покращення своїх алгоритмів, нейронні мережі найбільш підійдуть тільки для онлайн-сервісів, що працюють з великою кількістю користувачів.

Якщо адаптивна навчальна система планується не як онлайн-сервіс, що навчатиметься правильній поведінці від своїх чисельних користувачів, писати таку програму доведеться самостійно, використовуючи для оцінки знань користувача існуючі метрики чи розробивши свою власну. Тож, розглянемо існуючі метрики та розглянемо чи можна їх використати у розробці.

Чисельні метрики

Чисельні метрики характеризують об'єкт у чисельному виді. Якщо осноувати аналіз розуміння користувачем теми на чисельних метриках, то для досягання своєї мети можна використати числа для оцінки якості знання теми в цілому.

Нехай, P – оцінка знання користувачем теми, x – кількість вірних відповідей, y – кількість невірних. Так ми зможемо отримати відсоток теми, яку користувач системи зрозумів: $P = \frac{x}{x+y} \%$. При недоборі відсотків, тема не буде вважатися пройденою і буде починатися з початку, поки тест не буде пройдено.

Такий метод аналізу мало відрізнятиметься від звичайного збору статистики і надасть такий же самий результат, що і неадаптивна навчальна система.

Якщо ж, наприклад використовувати оцінну систему і ввести деяку M_i – оцінку за відповідь на питання по підтемі i , то окрім загальної оцінки за тему ($\sum_0^i M_i$) ми зможемо отримати й оцінку знань користувача саме у якійсь конкретній підтемі. Такий підхід є простим, але вже працюючим способом реалізації адаптивної навчальної системи.

Але недоліком чисельного засобу є обмеженість роботою лише з такими властивостями, що можуть бути виражені кількісно. Наприклад: кількість вірних та невірних відповідей; час, витрачений на відповідь; тощо. Якісні числа не матимуть такого ж впливу на підрахунок оцінки, як кількісні. Але для вірного підрахунку результату такі значення також важливі. Тож, переходимо до наступного типу метрик.

Якісні метрики

Якісні метрики характеризують об'єкт у вигляді слів, фраз чи просто символів. Деякі значення просто неможливо чи нерозумно записувати у вигляді чисел. Допустимо, що кожна відповідь на тест має вигляд:

Question(id, topic, theme, state), де *id* – порядковий номер у контролі, *topic* – назва теми, що перевіряється, *theme* – назва підтеми, до якої відноситься питання, а *state* – стан відповіді (вірно, невірно).

Таким чином, ми матимемо точніші дані і зможемо впізнати з якою підтемою в нього проблеми. Такий спосіб також надає можливість зробити працюючу адаптивну навчальну систему і є набагато інтуїтивнішим для програмістів.

Інформація, що буде отримана таким засобом може відображатись у текстовому вигляді і бути статистикою знань користувачем теми. Проте вирахувати з неї чисельну оцінку буде неможливо без комбінування цього засобу з чисельними. Зробити це можна просто задавши якісним значенням чисельних пріоритетів чи ваги, що пізніше буде приймати участь у підрахуванні кількісної оцінки.

Графи

За допомогою графів можна визначати рівень знань користувача, опираючись на елементі, на якому зупиниться переміщення по графу.

Наприклад, за допомогою дерев можна побудувати схему по якій користувач при невірній відповіді на питання по певній підтемі потрапляв би у вітку з більшою кількістю легших питань по цій самій підтемі, щоб упевнитися чи є у користувача з нею проблеми чи ні. При проходженню по різних листях

дерева, опираючись на вміст цих листів, можна динамічно скласти матеріал для наступної теми, що буде після контролю.

Графи допоможуть не тільки скласти види контролю, що будуть динамічно змінюватись під час їх проходження, що облегшить розуміння прогалин у знаннях користувача не тільки самою програмою, а й самим користувачем, а й допоможуть одночасно оцінити його успіхи, так як листя дерев можуть мати різну цінність.

Висновок

Тож, з вищесказаного стає зрозуміло, що для створення адаптивної навчальної системи має сенс використовувати комбінацію чисельних, якісних метрик та графів. Таким чином можна досягти оцінювання знань та адаптивної подачі матеріалу системою без залучення у навчання реальних людей-вчителів.

Використана література

1. Штучна нейронна мережа. URL:
https://uk.wikipedia.org/wiki/Штучна_нейронна_мережа (дата звернення: 23.01.2019).
2. AutoDraw By Google Creative Lab. URL:
<https://experiments.withgoogle.com/chrome/autodraw/> (дата звернення: 23.01.2019).
3. DeepDream. URL: <https://en.wikipedia.org/wiki/DeepDream> (дата звернення: 23.01.2019).
4. A Neural Network for Machine Translation, at Production Scale. URL:
<https://research.googleblog.com/2016/09/a-neural-network-for-machine.html> (дата звернення: 23.01.2019).

Додаток Г
Електронні матеріали