УДК 005:37

Лізунов Петро Петрович
Доктор технічних наук, професор, завідувач кафедри основ інформатики, *orcid.org/0000-0003-2924-3025*
Київський національний університет будівництва і архітектури, Київ
Білощицький Андрій Олександрович
Доктор технічних наук, професор, заступник декана факультету інформаційних технологій, *orcid.org/0000-0001-9548-1959*
Київський національний університет ім. Т. Шевченка, Київ
Чала Лариса Ернестівна
Кандидат технічних наук, доцент, доцент кафедри штучного інтелекту, *orcid.org/0000-0002-9890-4790*
Харківський національний університет радіоелектроніки, Харків
Білощицька Світлана Василівна
Кандидат технічних наук, доцент кафедри інформаційних технологій проектування та прикладної математики, *orcid.org/0000-0002-0856-5474*
Київський національний університет будівництва і архітектури, Київ
Кучанський Олександр Юрійович
Кандидат технічних наук, доцент кафедри інформаційних технологій, *orcid.org/0000-0003-1277-8031*
Київський національний університет будівництва і архітектури, Київ
Удовенко Сергій Григорович
Доктор технічних наук, професор, завідувач кафедри інформатики та комп'ютерної техніки, *orcid.org/0000-0001-5945-8647*
Харківський національний економічний університет ім. С. Кузнеця, Харків

ГІБРИДНИЙ ПІДХІД ДО АНАЛІЗУ ТА РОЗПІЗНАВАННЯ МАТЕМАТИЧНИХ ФОРМУЛ З МЕТОЮ ВИЯВЛЕННЯ В НИХ ПОДІБНОСТЕЙ

Анотація *Складність аналізу і розпізнавання математичних формул, які містяться в текстових документах, полягає в тому, що для знаходження неповних дублікатів необхідно аналізувати не просто графічне зображення, проводячи фільтрацію, виділення контурів і застосовуючи специфічні методи порівняння, а й текстову інтерпретацію формули, щоб мати змогу ідентифікувати неповний дублікат, за умови, що у формулі було змінено позначення літер, символи математичних операцій, форми дужок тощо. Тому для знаходження неповних дублікатів математичних формул пропонується гібридний підхід, що передбачає використання шаблонів, створених відповідно до особливостей графічних редакторів, та спеціальних конверторів формул з різних форматів до канонічного формату.*

Ключові слова: математичні формули; редактор формул; шаблон; дублікат; конвертор форматів

Вступ

Для перевірки наукових текстових документів на наявність подібностей з публікаціями, що можна знайти в загальнодоступних мережевих джерелах зазвичай використовують так звані системи «Антиплагіат», що реалізують технологію моніторингу та перевірки через стандартні пошукові алгоритми, які забезпечують достатньо швидкий та ефективний пошук повних або неповних дублікатів, а також гарантують коректну обробку текстів [1-8]. Втім такі системи не дозволяють здійснювати перевірку зображень та математичних формул. При цьому виникають проблеми, що неможливо вирішити за допомогою стандартних процедур.

Майже всі наукові тексти, зокрема дисертаційні роботи та наукові статті, містять математичні формули та зображення. Для вирішення цих проблем необхідно виробити нестандартні підходи. Аналіз зображень в текстах можна, наприклад, здійснювати з використанням стандартних програмних додатків та нестандартних додаткових модулей. При цьому для поліпшення якості розпізнавання треба реалізувати процедури відновлення зашумлених зображень, аналізу форми та текстури, виділення фрагментів зображень, зіставлення двох зображень і т.д.

Додаткові специфічні проблеми виникають в процесі аналізу математичних формул в порівнюваних текстах з метою виявлення в них подібностей.

Характерною рисою математичної інформації є використання скадної та високорозвиненої двовимірної символьної системи позначень. Складність аналізу і розпізнавання математичних формул полягає в тому, що для знаходження неповних дублікатів необхідно аналізувати не просто графічне зображення, проводячи фільтрацію, виділення контурів і застосовуючи специфічні методи порівняння, а й текстову інтерпретацію формули, щоб мати змогу ідентифікувати неповні дублікати за умови, що у формулі було змінено позначення літер, символи математичних операцій, форми дужок тощо [9]. Тому для знаходження неповних дублікатів математичних формул доцільно використовувати гібридні методи їх аналізу та порівняння.

В даній роботі запропоновано підхід, який дозволяє аналізувати подібність формул в наукових текстах, що порівнюються, з використанням спеціальних шаблонів та конверторів.

Мета статті

Метою дослідження є аналіз наявних засобів побудови формул за допомогою сучасних редакторів та методів порівняння формул в наукових текстах, що дозволяють розробити підхід до виявлення повних або часткових формульних дублікатів у публікаціях.

Виклад основного матеріалу

1. Засоби побудови формул

У тексті формули можна наводити як малюнок або як графічний об’єкт, створений за допомогою одного з редакторів формул. Редактор формул – комп’ютерна програма, призначена для створення та редагування математичних та інших формул. Редактори формул засновані на таких технологіях:

– застосування спеціальної мови розмітки, наприклад TeX, MathML у редакторі LaTex, Math для редактора OpenOffice;

– створення формул за допомогою графічного інтерфейсу: KFormula, MathType, MathCastmula, WIRIS Editor, MathCast (у цих редакторах формула будується зі складових, які надаються програмою);

– вбудовані компоненти: Math Expression Editor Light;

– символьні обчислення: Mathematica.

1.1. Спеціальні мови розмітки

Найбільш поширеними спеціальними мовами розмітки є LaTex та Math для OpenOffice.

LaTex – це система підготовки текстів, придатна, зокрема, для підготовки наукових публікацій, які містять математичні формули. Вона може використовуватись також для багатьох інших видів документів. LaTex побудована на базі TEX’a [10].

Мова розмітки TEX є досить поширеною мовою комп’ютерного верстання, розробленою Д. Кнутом [11]. Досі деякі відомі видання приймають до публікації технічні та математичні статті, що підготовлені саме у форматі TEX. Перевагою представлення формул в такому форматі є можливість повного управління зовнішнім виглядом будь-якої формули. Це є особливо важливим для запису складних математичних виразів, які містять матриці, системи рівнянь або нерівностей різних типів та т.ін. В цьому разі зображення формул в тексті є естетично привабливим. Мова розмітки текста TEX дещо близька до HTML та його узагальненню XML. У форматі TEX формули записуються рядками. При цьому використовуються макрокоманди мови, що починаються символом «\», за якими формується послідовність інших символів. Наприклад, грецька літера α записується як \alpha. Такий спосіб дозволяє записувати як прості, так і складні символи та математичні вирази. Для роботи з формулами на мові TEX використовуються спеціальні фільтри, що контролюють ознаки початку та кінця формули. Приклади та правила запису формул на мові TEX можна знайти, наприклад, в [12]. Ще одним засобом запису формул в текстах є використання алгебраїчного синтаксиса за правилами TEX , але при цьому формули на мові розмітки відокремлюються символами @@ з використанням текстового фільтру, що дозволяє розпізнавати ці обмежувальні символи. Переваги та недоліки такого представлення є такими ж, як і при використанні мови TEX.

Пізнішими різновидами мови TEX є мови розмітки типу LaTex або MikTEX [13]. Математичні вирази обробляються LaTex’ом у особливому режимі роботи, який називається математичною модою. Загальний для TEX’a та LaTex’a контрольний символ переходу в цю моду є знак долара $. У наукових статтях, дисертаціях, монографіях тощо одиничні математичні символи, або короткі, прості формули, які розташовані безпосередньо в тексті, відокремлюються одинарними знаками $...$, формули ж, які друкуються окремим рядком – подвійними знаками – $$...$$. Зазначимо, що у мові Math для редактора OpenOffice використовуються подібні позначки форматування, які дещо відрізняються від LaTex, але значної різниці між ними немає.

Наприклад, у Math для створення формули:

$$\Delta G = \Delta G^0 + RT\ln\frac{P_M^m P_N^n}{P_A^a P_B^b}$$

необхідно ввести текст:

%DELTA G= %DELTA G^0 + RT ln {P^m_M P^n_N} over {P^a_A P^b_B}.

Для позначення змінних використовуються букви латини, грецької мови (з використанням повного напису літер, «\alpha», «\beta»,) тощо. Для позначення дій користуються спеціальними позначками (наприклад, «^», «_»), математичними операторами (наприклад, «+», «–», «=», «\neq» (≠), «\sum» (∑) тощо), стандартними математичними функціями (наприклад, «\sin», «\min», «\ln» тощо), спеціальними словами-операторами, якими визначаються дроби, біноміальні коефіцієнти, радикали (наприклад, «\sqrt», «\frac {чисельник} {знаменник}» та інші). Зазначимо, що для запису математичних формул на web-сторінках останнім часом використовується також Math-система ASCIIMathML. Загальний принцип використання MathML полягає в тому, що математичні конструкції вбудовуються до звичайного HTML-документу та адекватно відображуються після його завантаження з мережі. Перше, що відрізняє мову розмітки MathML від аналогів – це використання двох способів кодування виразів. Один з них пов'язаний з безпосередньою передачею синтаксиса формули (presentation), другий, навпаки, відображує семантику виразу (*content*). Презентаційна розмітка описує математичну символіку з виразами, що будуються з використанням деяких схем виводу та засобів розміщення таких фрагментів виразів, як дроби, верхні і нижні індекси. Семантична розмітка описує математичні об'єкти та функції, де для кожного вузла будується дерево виразу згідно з деякою конкретною схемою, а гілки цього дерева віповідають підвиразам.

Оскільки питання розміщення текстів, що містять математичні формули та вирази, на web-сторінках є важливим, то останнім часом набули поширення практичні дослідження цієї проблеми. Зокрема, для цього використовуються Java-апплети, що визуалізують математичні формули в HTML-сторінках та дозволяють переглядати та редагувати їх у вікні браузера (апплет – прикладна програма на мові програмування Java у формі байт-кода, що виконується у веб-браузері з використанням віртуальної Java-машини). Наприклад, опис апплетів WebEQ Viewer Control та Input Control, розроблених фірмою Design Science, Inc для візуалізації формул, наведено в [14]. Крім того, для роботи з web-текстами, що містять формули, може використовуватись редактор формул DragMath Equation Editor. Після виклику цього редактора здійснюється запуск Java-машини та з'являється вікно побудови формул, що нагадує відповідне вікно редактора типу Microsoft Equation. При цьому правила розмітки формул в редакторах DragMath Equation Editor та Microsoft Equation є практично однаковими.

Таким чином, порівняння формул, створених у редакторах розглянутого типу, не викликає труднощів. За формулою створюється шаблон, який складається зі спеціальних символів, службових слів, позначень функцій. Під час порівняння формул порівнюються їх шаблони, і, якщо вони збігаються, постає питання про можливу тотожність формул. За аналогією аналізу текстів, під час аналізу формул, створених за допомогою спеціальних мов розмітки, треба відкидати так звані «стоп-слова». У мовах математичної розмітки такими словами можуть бути символи форматування, пробілів тощо.

1.2. Редактори з графічним інтерфейсом

Для створення математичних текстів широко використовується редактор формул Equation editor, що має специфічні можливості для побудови формул і входить до комплекту Microsoft Office. Цей редактор формул – скорочена версія програми «MathType» від фірми Design Science. Створені за допомогою цього редактора формули можна використовувати в додатках корпорації Microsoft, насамперед в текстовому редакторі Word, а також в програмі презентацій PowerPoint та табличному процесорі Excel. В цьому редакторі формул можна створювати складні математичні формули, використовуючи символи та шаблони панелі інструментів (рис. 1).

Панель містить понад 150 математичних символів та 120 шаблонів дробів, сум, границь і т.ін. Шаблони можна вкладати один в одний для створення багатоступеневих формул. Формули при цьому мають виключно ілюстративний характер, фактично вони є стилізованими малюнками, через що за ними не можна здійснювати обчислення. Формули в процесі створення оформлюються згідно з правилами запису математичних виразів, що прийняті в редакторі, але їх стилі та розмір можна змінювати за бажанням користувача. Формула, створена в Microsoft Equation, є «об'єктом», що займає в документі прямокутну область і може бути разташована зверху або усередині тексту.

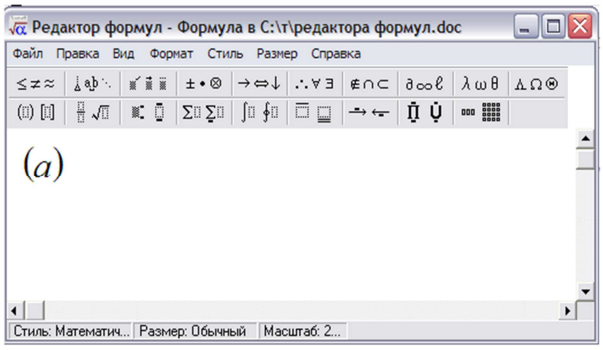

Рисунок 1 – Інтерфейс редактора MS Equation Editor 3.0

Назви деяких математичних функцій (наприклад, «sin», «cos») автоматично розпізнаються і пишуться прямим шрифтом.

MathType – це професійна версія редактора формул, яка працює з Word, Corel WordPerfect та багатьма іншими додатками, а також значно покращує можливості Equation Editor. MathType тісніше інтегрований в текстовий процесор. Цей редактор працює з формулами, як з частиною тексту, а не картинками, вставленими в текст, завдяки чому зникає багато проблем.

Інтерфейс користувача програми MathType, як і інтерфейси більшості інших додатків Windows, є інтуїтивним та орієнтованим на візуальне сприйняття. Для кожного математичного поняття в MathType є шаблон, який містить символи та пробіли для заповнення. Всього налічується біля 175 шаблонів, включаючи дроби, радикали, суми, інтеграли, матриці та різні види дужок.

Редактор MathType має інтерфейс, наведений на рис. 2.

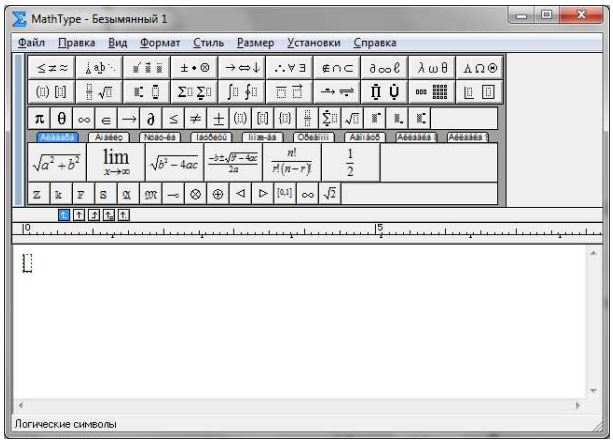

Рисунок 2 – Інтерфейс редактора MathType

Редактор формул у MS Word 2010 є однією з версій MathType. У Microsoft Word 2010 завжди є можливість перетворити формулу, створену за допомогою MathType, у малюнок формату bmp. В цьому випадку порівняння формул має здійснюватись як порівняння малюнків, тим більше, що формат формул-малюнків буде збігатися за розміром (у випадку однакових формул).

1.3. Редактори напису хім-ічних формул

Крім звичайних математичних формул, в деяких наукових публікаціях можуть бути наявні хімічні формули. Для роботи з такими документами інколи неможливо використовувати лише текстові або графічні редактори загального призначення. Редагування текстів, що містять хімічну інформацію, потребує використання спеціалізованих програмних інструментів.

Для напису хімічних формул авторами найчастіше використовується редактор LibreOffice Math. Хімічні формули зазвичай будуються у прямому зображенні. При використанні Writer хімічні формули обробляються як об’єкти текстового документу, що полегшує їх подальше порівняння в наукових текстах.

Розробниками Microsoft Research та Центру Unilever було представлено нове розширення для MS Office: редактор Chem4Word, що дозволяє додавати та редагувати хімічні формули в документах, створених за допомогою Word, Excel та PowerPoint.

Програмне забезпечення цього редактора хімічних формул, що підтримує версії MS Word 2007 та MS Word 2010, розповсюджується безкоштовно. Редактор Chem4Word дозволяє використовувати специфічні символи, мітки та вставляти до документів, які редагуються, хімічні формули та зображення.

На рис. 3 наведено приклад розмітки формул у редакторі LibreOffice Math. Редагування текстів, що містять складну хімічну інформацію, а також створення та зберігання хімічних структурних формул та схем хімічних реакцій, інколи здійснюються з використанням спеціалізованого хімічного редактора Symyx Draw [15].

Частина хімічної речовини	Формула	Розмітка
Молекули	H2SO4	H_2 SO_4 (пропуск між елементами обов’язковий)
Ізотопи	$^{238}_{92}\mathrm{U}$	U lsub 92 lsup 238
Іони	$\mathrm{SO_4^{2-}}$	SO_4^{2-{}} або SO_4^{2"-"}

Рисунок 3 – Приклад розмітки хімічних формул у редакторі LibreOffice Math

Документи цього редактора можуть входити до складу документу Word або розміщуватися в окремих графічних файлах. В останньому випадку створені в Symyx Draw хімічні формули можна зберігати в одному з поширених графічних форматів (bmp, gif т.ін.)

2. Методи порівняння формул у наукових текстах

2.1. Порівняння формул з використанням шаблонів

Порівняння формул є значно складнішим, ніж порівняння тексту (наприклад, формули x2 і a2 – за шаблонами є ідентичними, а за найменуваннями змінних різняться).

Якщо формули збережені як об'єкт MathType, доцільно застосовувати порівняння за шаблоном.

Наприклад, потрібно порівняти формули:

$$W_1 \xleftarrow{F_{(w_1,L_1)}^{\downarrow}} L_1 \xleftarrow{F_{(w_2,L_1)}^{\downarrow}} W_2 \text{ та } Q_1 \xleftarrow{F_{(q_1,P_1)}^{\downarrow}} P_1 \xleftarrow{F_{(q_2,P_1)}^{\downarrow}} Q_2$$

Віповідний шаблон для їх порівняння наведено на рис. 4.

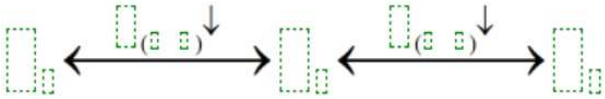

Рисунок 4 – Шаблон для порівняння формул

Таким чином, з'явиться можливість спочатку порівнювати шаблони, а потім – найменування змінних.

Треба зауважити, що деякі математичні та фізичні величини мають стійкі загальновживані позначення, тому слід зважати на контекст, у якому згадується формула. Зазвичай пояснення до формул надаються у абзацах до або після формули. Тому для різних галузей треба створити словники з означенням величин та їх позначень. Наприклад, сила струму позначається – I, ймовірність – P тощо.

Стандартами оформлення документації (Державним стандартом України 3008–95, міждержавним стандартом 2.105–95 та іншими стандартами) встановлені правила оформлення формул та пояснень до них:

– позначення символів та числових коефіцієнтів, які входять у формули, має бути наведене безпосередньо під формулою;

– кожне позначення наводять з нового рядка без абзацу, дотримуючись послідовності, з якою вони розміщені у формулі;

– перший рядок розшифрування має починатися словом «де» без двокрапки після нього. Після формули (перед словом «де») ставиться кома.

2.2. Порівняння формул з використанням конверторів

Перспективним напрямом автоматизованого аналізу математичних формул є створення конверторів з різних форматів (TeX, Equation, MathType) до канонічного формату XML/MathML [16]. Такі конвертори мають формуватися у вигляді онлайнових сервісів, що є доступними як для користувачів, так і для систем, де використовуються математичні формули.

Традиційно для створення математичних текстів активно використовується редактор MS Word, що має специфічні можливості для побудови формул. Цей редактор є більш звичним для користувачів, ніж класична система роботи з математичними текстами (La)Tex, оскільки не потребує інсталяції додаткового програмного забезпечення. Разом з тим, слід зазначити, що поліграфічна якість математичних формул у MS Word є гіршою, ніж в системі TeX. Ще однією проблемою є закритість форматів Microsoft. З цього випливає обмеженість використання word-форматів для публікації інформації в мережі Інтернет – доводиться конвертувати текст до html, причому зазвичай MS Word при такому конвертуванні використовує застарілий метод переведення формул до растрової графіки (наприклад, до gif-формату).

Наявність різних стандартів представлення математичних та природничо-наукових текстів додає труднощів до вирішення проблеми порівняння математичних формул в документах. Використання декількох стандартів в дисертаційних роботах, періодичних виданнях, матеріалах конференцій, наукових монографіях призводить до того, що інформаційне середовище заповнюється документами, які представлені у важкосумісних форматах. Це викликає погіршення цілісності інформаційного середовища та виникнення проблем з використанням текстів у різних представленнях. Перспективним підходом до роботи з природничо-науковими текстами є розробка та застосування спеціальних діалектів (словників) XML. Умовним стандартом побудови математичних формул у Web-середовищі та розміщення у ньому математичних текстів вважається мова математичних символів MathML (Mathematical Markup Language) [17]. Ця мова є підмножиною розширеної мови розмітки XML (eXtensible Markup Language), що часто використовується для створення інших мов. Таке застосування XML є цілком природним і добре працює тоді, коли використання HTML для передачі даних нових типів обмежене узгодженням особливостей їх форматів.

До найбільш поширених мов розмітки, які основані на XML, належать:

– Wireless Markup Language (WML): формат даних для бездротових пристроїв, що працюють з протоколом WAP (мобільні телефони);

– Synchronized Multimedia Integration Language (SMIL): задає часову розмітку, зовнішній вигляд і т.ін. для мультимедійних презентацій; визначає послідовність відображення ультимедійних файлів;

– Scalable Vector Graphics (SVG): використовується для опису двовимірної векторної графіки;

– Chemical Markup Language (CML): використовується для опису хімічних формул.

Разом зі специфікаціями, пов'язаними із задачею стандартизації структури текстів

(наприклад, DocBook), MathML здатен сформувати канонічний формат представлення природничо-наукових текстів, що використовує розмітку логічного рівня та дозволяє здійснення трансляції до будь-якого класичного або нестандартного формату, орієнтованого на зовнішнє представлення документів. Втім слід зазначити, що мова MathML створювалася з метою її використання для побудови комунікативного ланцюжка та автоматизації роботи сервісів. Через це вона не орієнтована на безпосереднє користування внаслідок її складності. Для генерування MathML-кодів необхідно використовувати WYSIWYG-редактори або конвертори з інших систем побудови математичних текстів. Зокрема, доцільно розробити програму-конвертор формул з документів Microsoft Word, що створені з використанням редактора формул Microsoft Equation Editor 3.0 та MathType 5.0 до формату MathML з оптимізацією для перегляду у різних Інтернет-браузерах. Реалізація такої програми може бути виконана на мовах VBA або Java та базуватися на витяганні структур формул з RTF-файлів. Застосування Java дозволяє здійснювати трансляцію формул незалежно від конкретного редактора MS Word. Наприклад, можна застосовувати редактор формул Equation Editor як WYSIWYG-редактор, що генерує MathML-код, а також дозволяє транслювати до стандартного формату раніше створені математичні тексти. Така технологія трансляції математичних формул на основі обробки RTF-файлів реалізована як онлайновий сервіс, може бути застосована для трансляції формул з документів Microsoft Office до різних форматів, в тому числі і до TeX.

Завдання, що були поставлені робочою групою W3C з математики при створенні MathML:

– забезпечення кодування матеріалів математичного характера для комунікацій усіх рівней освітнього та наукого типу;

– забезпечення кодування як математичної символіки, так і її значень;

– підтримка створення шаблонів та інших інструментів математичного редагування;

– забезпечення перетворень до інших математичних форматів як суто презентаційного, так і семантичного характеру.

Підтримку конвертування математичних формул з розміткою LaTex до представлення HTML може бути здійснено за допомогою програмного конвертора Pandoc. Для виводу математики до HTML можуть бути використані спеціальні перетворювачі на основі MathML, Java-Script, онлайн-сервисів, код яких вставляється в сконвертований HTML-файл.

Приклад з результатами такого конвертування наведено на рис. 5.

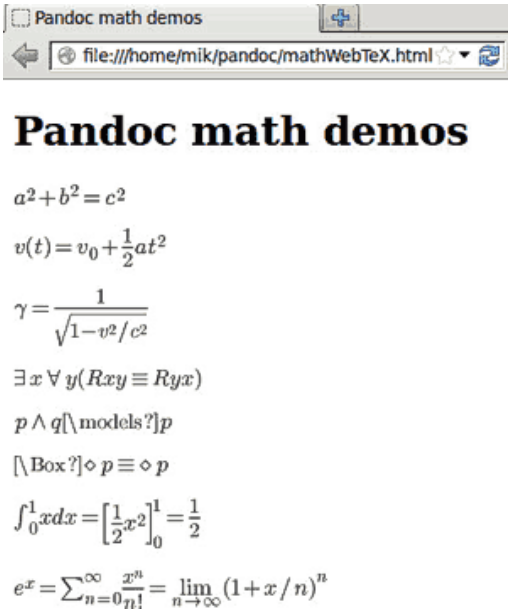

Рисунок 5 – Приклад конвертування формул з використанням програми Pandoc

Для конвертування математичних формул з розміткою LaTex можна використовувати такі опції програми Pandoc:

– mathml – перетворює формули LaTex до розмітки MathML;

– webtex – перетворює формули LaTex за допомогою онлайн-сервіса Google Chart API;

– mathjax – перетворює формули LaTex за допомогою розширення MathJax для MediaWiki;

– LaTexmathml – перетворює формули LaTex за допомогою JS-бібліотеки LaTexmathml.

2.3. Гібридна схема порівняння формул

Запропонована гібридна схема порівняння формул здійснює перевірку оригінальності формульного об’єкта, що аналізується, як за джерелами мережі Інтернет, так і за власними базами текстів з формулами (статей, дисертацій, курсових, дипломних та магістерських робіт і т.ін.).

Джерела, що містять формули, можуть зберігатися як у вигляді повних текстів, необхідних для оцінки характеру подібностей (за результатами перевірки), так і у вигляді спеціального пошукового індексу, необхідного для швидкої перевірки збігу текстів та бази джерел.

Припустимо, що згідно з однією з наявних систем пошуку текстів (без урахування можливої близькості наявних там формул), що є близькими (за сучасними критеріями близькості текстів або їх фрагментів), була сформована обмежена множина (A) об’єктів з високим ступенем подібності до текстового об’єкта з формулами, оригінальність якого аналізується.

Розглянемо можливі варіанти представлення текстових об'єктів з формулами, що входять до множини A:

а) формули в текстах збережені як об'єкт MathType (варіант 1);

б) об'єкти збережені у вигляді тексту з графічними зображеннями формул (наприклад, у форматах GIF або JPEG) (варіант 2);

в) об'єкти, що містять формули, збережені повністю у вигляді документів у форматі PDF (варіант 3).

У першому випадку (варіант 1) доцільно застосовувати порівняння за шаблоном згідно з п. 2.1.

Алгоритм порівняння полягає в такому:

– формуються шаблони формульних об'єктів, що аналізуються (шаблони X);

– поступово формуються шаблони формул з текстових об'єктів з формулами, що входять до множини A (шаблони Y);

– здійснюється перевірка повних збігів структури шаблонів X з шаблонами Y (процедура такої перевірки є тривіальною) та формується клас шаблонів Z, що відповідають випадкам повного збігу за результатами перевірки;

– для шаблонів X та Y з однаковою структурою (тобто шаблонів з класу Z) здійснюється перевірка тотожності найменування змінних;

– в разі наявності однієї або кількох формул в множині Z, що мають повний збіг (як за шаблонами, так і за відповідними найменуваннями змінних) з формульними об'єктами, що аналізуються, здійснюється перевірка наявності посилань в текстових фрагментах;

– в разі відсутності посилань на первинне джерело для випадків повного збігу (як за шаблонами, так і за відповідними найменуваннями змінних) фрагменти (сторінки) з однаковими формулами (в первинному джерелі та джерелі, що перевіряється на плагіат), автоматично заносяться до електронної форми «Індикація можливості формульного плагіату», що створюється для кожного аналізованого об'єкта.

Для варіанта 1 передбачається також можливість виявлення часткового дублювання формул.

Зауважимо, що редактор MathType містить транслятор формул та математичних виразів до MathML формату.

В разі необхідності аналізу та порівняння об'єктів, що збережені у вигляді тексту з графічними зображеннями формул (варіант 2), може бути застосована технологія, яка реалізована у програмі Infty [18]. Ця програма дозволяє звести завдання варіанта 2 до традиційного підходу роботи з відсканованими документами. Вона використовує технологію оптичного розпізнавання символів OCR (Optical character recognition) і застосовує структурний аналіз до отриманого результату. У той час як ця технологія розпізнає складні математичні формули в документі, вона ігнорує іншу текстову інформацію. Використання програми Infty при розпізнаванні математичних формул із растрових графічних зображень дозволяє їх представляти мовою MathML та здійснювати в разі необхідності перетворення їх на текст.

В роботі [19] наведено результати досліджень щодо автоматизації процесу декомпозиції формул, записаних мовою MathML, з подальшою програмною реалізацією перетворення їх на текстовий опис. Пошук формул у тексті відбувається на основі удосконаленого методу трансформації синтаксичного дерева та методу пошуку за ключовими словами. Розроблений авторами [19] модуль знаходить математичні формули у документі відповідно до маски пошуку. Математичні формули в документі HTML/XHTML починаються із тегу $і закінчуються тегом$. У результаті всі теги, які розміщені між цими двома, озвучуються. Отже, математична формула, яка записана мовою математичної розмітки MathML, відповідно до методу трансформації синтаксичного дерева і розроблених правил, перетворюється на текстовий опис і записується у текст документа.

Модуль перетворення формул на текстовий опис забезпечує пошук у тексті математичних формул та спеціальних символів відповідно до розроблених правил. Для перетворення MathML на текст використовується модель трансформації синтаксичного дерева [20].

Аналіз документів, у яких зустрічаються математичні формули у форматі PDF (варіант 3), як і раніше є складним та не остаточно вирішеним завданням. На сьогодні розпізнавання складних математичних формул із PDF-зображень можливе лише зі значними неточностями передавання змісту формули. Програма Infty може бути використана для аналізу як графічної, так і текстової інформації, яка міститься в документі PDF. Для цього використовується розроблений авторами [19] дворівневий аналізатор контенту і структури математичних формул з подальшим їх записом мовою MathML.

Зазначимо, що різноплановість представлення текстових об'єктів з формулами, що входять до множини A (варіанти 1, 2 та 3), підтверджує доцільність розробки гібридного методу, який забезпечує конвертування формул різних форматів (MathType, TeX, LaTex, JPEG, PDF тощо) до уніфікованого формату мови математичної розмітки MathML.

У загальному випадку, аналіз подібностей та виявлення повного або часткового дублювання формул може бути здійснено за такою схемою:

– сканування формули з відповідного інформаційного джерела (друкована стаття, електронний документ, зображення тощо);

– розпізнавання відсканованої формули та подання її мовою MathML. Розпізнавання формули або графічного зображення відбувається засобами програми Infty (серед засобів перетворення документу TeX на мову MathML найбільш якісною вважається програма TtM, розроблена А. Хатчінсоном (I. Hutchinson) [21]. Ця програма перетворює документ TeX у файл формату XHTML, до складу якого входять MathML формули). У результаті формується файл із розширенням *.mml, в якому записана формула;

– аналіз файлів формул, що порівнюються, з метою виявлення їх повного або часткового дублювання;

– запис результату аналізу до електронної форми «Індикація можливості формульного плагіату», що створюється для кожного аналізованого об'єкта.

Згідно з цією схемою можна виділити такі варіанти її реалізації на етапі перетворення математичних формул у різних форматах на мову математичної розмітки MathML (залежно від форматів представлення формул, що аналізуються) [22]:

1. **.doc → «GrindEQ Math» → TeX → Teacode LaTex → MathML*: перетворення за допомогою плагіну GrindEQ Math Utilities та онлайн-сервісу Teacode LaTex на мову MathML набору формул, записаних у форматі текстового документу Microsoft Word;

2. *MathType → MathML*: перетворення на мову MathML набору формул, записаних засобами MathType;

3. **.html → виділення графічних об'єктів → JPG, PNG, GIF → Infty → MathML*: перетворення набору формул, записаних у вигляді графічних об'єктів веб-сторінки *.html на мову MathML засобами програми Infty;

4. **.pdf → Infty → MathML*: перетворення на мову MathML набору формул, записаних у форматі Adobe Reader, *.pdf за допомогою програми для розпізнавання символів Infty.

5. *LaTex → Pandoc → MathML*: перетворення на мову MathML набору формул, записаних засобами LaTex.

Примітка: для варіанта 1 може виявитися достатнім застосування порівняння за шаблоном згідно з п. 2.1 та відповідним алгоритмом, наведеним вище. В разі ж необхідності уточнення аналізу може бути і в цьому варіанті додатково застосовано схему з перетворенням на мову MathML формул, записаних засобами MathType.

Результатом реалізації запропонованої гібридної схеми може бути формування електронної форми «Індикація можливості формульного плагіату», що створюється для кожного аналізованого об'єкта, з визначенням загальних коефіцієнів подібності формульних документів, які перевіряються, та побудовою порівняльної гістограми.

Висновки

Наведено аналіз наявних засобів побудови формул за допомогою сучасних редакторів та методів порівняння формул в наукових текстах, що дозволив розробити підхід до виявлення повних або часткових формульних дублікатів в публікаціях, курсових, дипломних проектах або магістерських роботах тощо.

Запропонована гібридна схема порівняння формул здійснює перевірку оригінальності формульного об'єкта, що аналізується, як за джерелами мережі Інтернет, так і за власними базами текстів з формулами. Схема передбачає можливість аналізу подібностей та виявлення повного або часткового дублювання формул як з використанням шаблонів формульних об'єктів, що аналізуються, так і з застосуванням конвертації математичних формул у різних форматах на універсальну мову математичної розмітки MathML.

Отримані результати можуть бути, зокрема, використані для розширення функціональних можливостей наявних систем виявлення плагіату в наукових текстах, що містять формульні об'єкти.

Список літератури

1. *Антиплагиат [Электронный ресурс]. – Режим доступа: http://www.antiplagiat.ru*

2. *Білощицький, А.О. Ефективність методів пошуку збігів у текстах [Текст] / А.О. Білощицький, О.В. Діхтяренко // Управління розвитком складних систем. – 2013. – № 14.– С. 144 – 147.*

3. *Білощицький А.О. Оптимізація системи пошуку збігів за допомогою використання алгоритмів локально чутливого хешування наборів текстових даних [Текст] / А.О. Білощицький, О.В. Діхтяренко // Управління розвитком складних систем. – 2014. – № 19.– С. 113 – 117.*

4. Білощицький А.О. Метод вилучення помилкових збігів текстів в електронних документах [Текст] / А.О. Білощицький, С.Д. Криштоф, С.В. Білощицька, О.В. Діхтяренко // Управління розвитком складних систем. – 2015. – № 22(1). – С. 144 – 150.

5. Михайловський Ю.Б. Система Anti-Plagiarism як інструмент запобігання плагіату в навчальній та науковій діяльності [Текст] / Ю.Б. Михайловський, Н.А. Длугунович // Вісник Хмельницького національного університету. Технічні науки. – 2013. – № 3. – С. 162 – 168.

6. Лупаренко Л. А. Інструментарій виявлення плагіату в наукових роботах: аналіз програмних рішень [Текст] / Л.А. Лупаренко // Інформаційні технології і засоби навчання. – 2014. – Т. 40. – №2.– С. 151 – 169.

7. Шарапова Е.В. Исследование возможностей системы «Антиплагиат» для обнаружения заимствований [Текст] / Е.В. Шарапова // Перспективы науки и образования. – 2013 – №3.– С. 215 – 218.

8. Shenoy M. Automatic Plagiarism Detection Using SimilarityAnalysis [online] / M. Shenoy, K. C.Shet, U.D. Acharya// Advanced Computing: An International Journal. – 2012. – № 3 (3). – P. 59-62.

9. Очков В.Ф. Формулы в научно-технических публикациях: проблемы и решения [Текст] / В. Ф. Очков // Электронный журнал Cloud of Science. – 2014. – Т. 1. – № 3. – 421 – 455. – http://cloudofscience.ru

10. Kopka H., Daly P. A Guide to LATEX and Electronic Publishing [Текст] / H. Kopka, P. Daly. – 2003. – Addison-Wesley, Fourth edition – 660 p.

11. Кнут Э. Д. Все про TeX. [Текст] / Э.Д. Кнут. – М.: Изд. "Вильямс", 2003. – 560 с.

12. Очков В.Ф. Встроенные вычисления и отображение формул в электронных и печатных изданиях [Текст] / В.Ф. Очков, Е.П. Богомолова., Е.В. Никульчев., С. Герк // Известия высших учебных заведений. Проблемы полиграфии и издательского дела. – № 6. – 2015. – С. 45 – 58.

13. Гуссенс М. Путеводитель по пакету LaTex и его web-приложениям [Текст] / М. Гуссенс, С. Ратц. – М.: Мир, 2001. – 601 с.

14. Гайтан О.М. Елементи технології реалізації автоматизованого адаптивного контролю знань студентів в комп'ютерних системах навчання [Текст] / О.М. Гайтан // Радіоелектронні і комп'ютерні системи. – 2014. – № 4 (68). – С. 97-105.

15. Душкевич.О.Г. Редактор химических формул Symyx Draw: Учебно-методическое пособие для студентов химических специальностей [Текст] / О. Г. Душкевич. – Минск.: БГУ, 2014. – 40 с.

16. Давидов М.В. Метод та інформаційна технологія озвучення математичних формул українською мовою [Текст] / М.В. Давидов, О.А. Лозицький, В.В. Пасічник // Штучний інтелект. – № 1. – 2013. – С. 233 – 245.

17. Елизаров А.М. Веб-технологии для математика: основы MathML [Текст] / А.М. Елизаров, Е.К. Липачев, М.А. Малахальцев. – М.: Физматлит, 2010. – 192 с.

18. J. Baker A linear grammar approach to mathematical formula recognition from PDF [Текст] / Baker J., Sexton A. and Sorge V. // Proc. of Intelligent Computer Mathematics, 2009. – P. 127 – 133.

19. Давидов М. В. Метод озвучення математичних формул та символів українською мовою [Текст] / М. В. Давидов, О. А. Лозицький, Ю. В. Нікольський // Наукові праці [Чорноморського державного університету імені Петра Могили]. Сер. : Комп'ютерні технології. – 2013. – Т. 213, Вип. 201. – С. 24 – 29.

20. Ehrig H. Handbook of Graph Grammars and Computing byGraph Transformations [Текст] / H.Ehrig, G.Engels, H.–J. Kreowski, G.Rozenberg. – Volume 2:Applications, Languages and Tools World . – Scientific, 1999. – 132 p.

21. Ian Hutchinson. Web Publishing Mathematics With MathML [Електронний ресурс] / Hutchinson Ian // IAP 2004. – Режим доступу : http://web.m it.edu/acs/iap05/mathml/mathmlfuture.pdf.

22. Josef B. Baker, Alan P. Sexton, Volker Sorge, Masakazu Suzuki : Comparing Approaches to Mathematical Document Analysis from PDF. ICDAR 2011: 463-467 [Електронний ресурс]. – Режим доступу: http://www.inftyproject.org/en/index.html.

Стаття надійшла до редколегії 14.07.2016

Рецензент: д-р техн. наук, проф. С.Д. Бушуєв, Київський національний університет будівництва і архітектури, Київ.

Лизунов Петр Петрович
Доктор технических наук, профессор, заведующий кафедрой основ информатики, *orcid.org/0000-0003-2924-3025*
Киевский национальный университет строительства и архитектуры, Киев
Белощицкий Андрей Александрович
Доктор технических наук, профессор, заместитель декана факультета информационных технологий, *orcid.org/0000-0001-9548-1959*
Киевский национальный университет им. Т. Шевченко, Киев
Чалая Лариса Эрнестовна
Кандидат технических наук, доцент кафедры информационных технологий, *orcid.org/0000-0002-9890-4790*
Харьковский национальный университет радиоэлектроники, Харьков
Белощицкая Светлана Васильевна
Кандидат технических наук, доцент кафедры информационных технологий проектирования и прикладной математики, *orcid.org/0000-0002-0856-5474*
Киевский национальный университет строительства и архитектуры, Киев

Кучанский Александр Юрьевич
Кандидат технических наук, доцент кафедры информационных технологий, orcid.org/0000-0003-1277-8031
Киевский национальный университет строительства и архитектуры, Киев
Удовенко Сергей Григорьевич
Доктор технических наук, заведующий кафедрой информатики и компьютерной техники, *orcid.org/0000-0001-5945-8647*
Харьковский национальный экономический университет им. С. Кузнеца, Харьков

ГИБРИДНЫЙ ПОДХОД К АНАЛИЗУ И РАСПОЗНАВАНИЮ МАТЕМАТИЧЕСКИХ ФОРМУЛ С ЦЕЛЬЮ ВЫЯВЛЕНИЯ В НИХ ПОДОБИЙ

Аннотация. *Сложность анализа и распознавания математических формул, содержащихся в текстовых документах, состоит в том, что для нахождения неполных дубликатов необходимо анализирвать не просто графическое изображение, проводя фильтрацию, выделение контуров и применяя специфические методы сравнения, но и текстовую интерпретацию формулы, чтобы иметь возможность идентифицировать неполный дубликат, при условии, что в формуле были изменены обозначения букв, символы математических операций, формы скобок и т.п. В связи с этим для нахождения неполных дубликатов математических формул предлагается гибридный подход, который предусматривает использование шаблонов, созданных в соответствии с особенностями графических редакторов, и специальных конверторов формул разных форматов к каноническому формату.*

Ключевые слова: математические формулы; редактор формул; шаблон; дубликат; конвертор форматов

Lizunov Petro
DSc(Eng.), Professor, *orcid.org/0000-0003-2924-3025*
Kyiv National University of Construction and Architecture, Kyiv
Biloshchytskyi Andrii
DSc (Eng), Professor, Deputy Dean of the Faculty of Information Technology,*orcid.org/0000-0001-9548-1959*
Taras Shevchenko National University of Kyiv, Kyiv
Chala Larysa
PhD (Eng.), Docent of Artificial Itelligence Deprtment, *orcid.org/0000-0002-9890-4790*
Kharkiv National University of Radioelectronics, Kharkiv
Biloshchytska Svitlana
Ph.D., assistant professor of information technology designing and applied mathematics, *orcid.org/0000-0002-0856-5474*
Kyiv National University of Construction and Architecture, Kiev
Kuchansky Alexander
PhD(Eng.), Docent of Information Technology Department,*orcid.org/0000-0003-1277-8031*
Kyiv National University of Construction and Architecture, Kyiv
Udovenko Serhii
DSc(Eng.), Head of Informatics and Computer Technique Deprtment, *orcid.org/0000-0001-5945-8647*
Simon Kuznets Kharkiv National University of Economics, Kharkiv

HYBRID APPROACH TO ANALYSIS AND RECOGNITION OF MATHEMATICAL FORMULAS TO IDENTIFY THEIR SIMILARITY

Abstract*. The complexity of the analysis and recognition of mathematical formulas contained in text documents, is that in order to find the near duplication is necessary not only analysis a graphic image, by filtration, edge detection and using specific methods of comparison, but also textual interpretation of the formula, to be able to identify the part duplicate, provided that in the formula have been changed designation letters, symbols, mathematical operations, and the like shaped brackets In this regard, in order to find the near duplicates of mathematical formulas it had been proposed a hybrid approach that involves the use of templates, created in accordance with the features of graphic editors, special converters of formulas in various formats to the canonical format.*

Keywords: mathematical formulas, equation editor; template; duplicate; format converter

References

1. Antiplagiat. (2016). http://www.antiplagiat.ru.

2. Biloshchytskyi, A. & Dikhtiarenko, O. (2013). The effectiveness of methods for finding matches in texts. Management of complex systems, 14, 144–147.

3. Biloshchytskyi, A. & Dikhtiarenko, O. (2014). Optimization of Matching algorithms by using local-sensitive hash sets of text data. Management of complex systems, 19, 113–117.

4. Biloshchytskyi, A., Kristof, S., Biloshchytska, S. & Dikhtiarenko, O. (2015). The method of elimination of erroneous coincidences text in electronic documents. Management of Development of Complex Systems, 22(1), 144–150.

5. Myhaylovskyi, Yu., Dluhunovych, N. (2013). Anti-Plagiarism System as a Tool for Plagiarism Preventing in Educational and Research Activities. Jornal. Khmelnytskyi National University, 3, 162-168.

6. Lupanenko, L.A. (2014). Plagiarism detection tools for research works: analysis of software solutions. Information Technology and Learning Tools, Vol 40, 2, 151–169.

7. Sharapova, E. (2013). Investigation of possibilities "Anti-plagiarism" system to detect borrowing. Prospects for Science and Education, 3, 215–218.

8. Shenoy, M., Shet, K., Acharya, U. (2012). Automatic Plagiarism Detection Using Similarity. Advanced Computing: An International Journal, 3(3), 59–62.

9. Ochkov, V. (2014). The formulas in the scientific and technical publications: Problems and solutions. Cloud of Science, Vol. 1., 3, 421–455.

10. Kopka, H., Daly, P. (2003). A Guide to LATEX and Electronic Publishing: Addison-Wesley, Fourth edition, 660.

11. Knut, E. (2003). All about TeX. Moscow: Williams, 560.

12. Ochkov, V., Bogomolova, E., Nikulchev, E., Gerk, S. (2015). Embedded computing and displaying formulas in the electronic and print media. Proceedings of the higher educational institutions. Problems printing and publishing, 6, 45–58.

13. Gussens, M., Ratz, S. (2001). Guide to LaTeX package andweb-applications. Moscow: Mir, 601.

14. Gaitan, O. (2014). Elements of adaptive technologies of automated monitoring of student learning in computer systems. Radio electronic and computer systems, 4(68), 97–105.

15. Dushkevych, O. (2014). Editor chemical formulas SymyxDraw. Minsk: BGU, 40.

16. Davydov, M., Lozytskyy, O., Nikolskyy, Y. (2013). Method of automatic formulas reading in Ukrainian. Artificial Inteligence, 1, 233–245.

17. Elizarov, A., Lipachev, E., Malahaltsev, M. (2010). Web technologies for mathematics: the foundations MathML. Moscow: Fizmatlit, 192.

18. Baker, J., Sexton, A., Sorge, V. (2009). A linear grammar approach to mathematical formula recognition from PDF. Proc. of Intelligent Computer Mathematics, 127–133.

19. Davydov, M., Lozytskyy, O., Nikolskyy, Y. (2013). Method of automatic formulas and symbols reading in Ukrainian. Scientific works, Petro Mohyla Black Sea National University, 213(201), 24–29.

20. Ehrig, H., Engels, G., Kreowski, H.–J., Rozenber, G. (1999). Handbook of Graph Grammars and Computing by Graph Transformations. Volume 2: Applications, Languages and Tools World: Scientific, 132.

21. Hutchinson, I. Web Publishing Mathematics With Math ML: IAP. (2004). http://web.mit.edu/acs/iap05/mathml/mathmlfuture.pdf.

22. Baker, J., Sexton, A., Sorge, V., Suzuki, M. Comparing Approaches to Mathematical Document Analysis from PDF: ICDAR (2011).http://www.inftyproject.org/en/index.html.

Посилання на публікацію

APA *Lizunov, Petro, Biloshchytskyi, Andrii, Chala, Larysa, Biloshchytska, Svitlana, Kuchansky, Alexander, & Udovenko, Serhii. (2016). Hybrid approach to analysis and recognition of mathematical formulas to identify their similarity. Management of Development of Complex Systems, 27, 145–155.*

ГОСТ *Лізунов, П. П. Гібридний підхід до аналізу та розпізнавання математичних формул з метою виявлення в них подібностей [Текст] / П. П. Лізунов, А. О. Білощицький, Л. Е. Чала, С.В. Білощицька, О. Ю. Кучанський, С. Г. Удовенко // Управління розвитком складних систем. – 2016. – № 27. – С. 145 – 155.*