Erschienen in: CIKM 2017 : proceedings of the 2017 ACM Conference on Information and Knowledge Management : November 6-10, 2017, Singapore / Lim, Ee-Peng (Hrsg.). - New York, New York, USA : ACM Press, 2017. - S. 2211-2214. - ISBN 978-1-4503-4918-5
https://dx.doi.org/10.1145/3132847.3133144

Analyzing Mathematical Content to Detect Academic Plagiarism

Norman Meuschke[1], Moritz Schubotz[1], Felix Hamborg[1], Tomas Skopal[2], Bela Gipp[1]
[1]University of Konstanz (first.last@uni-konstanz.de)
[2]Charles University in Prague (skopal@ksi.mff.cuni.cz)

ABSTRACT

This paper presents, to our knowledge, the first study on analyzing mathematical expressions to detect academic plagiarism. We make the following contributions. First, we investigate confirmed cases of plagiarism to categorize the similarities of mathematical content commonly found in plagiarized publications. From this investigation, we derive possible feature selection and feature comparison strategies for developing math-based detection approaches and a ground truth for our experiments. Second, we create a test collection by embedding confirmed cases of plagiarism into the NTCIR-11 MathIR Task dataset, which contains approx. 60 million mathematical expressions in 105,120 documents from arXiv.org. Third, we develop a first math-based detection approach by implementing and evaluating different feature comparison approaches using an open source parallel data processing pipeline built using the Apache Flink framework. The best performing approach identifies all but two of our real-world test cases at the top rank and achieves a mean reciprocal rank of 0.86. The results show that mathematical expressions are promising text-independent features to identify academic plagiarism in large collections. To facilitate future research on math-based plagiarism detection, we make our source code and data available.

1 INTRODUCTION

Academic plagiarism has been defined as "the use of ideas, concepts, words, or structures without appropriately acknowledging the source to benefit in a setting where originality is expected" [6]. Detecting academic plagiarism is a pressing problem, e.g., for educational and research institutions, funding agencies, and academic publishers. Research on information retrieval (IR) approaches for plagiarism detection (PD) has yielded mature systems that employ text retrieval to find suspiciously similar documents. These systems reliably retrieve documents containing (nearly) copied text, but often fail to identify disguised forms of academic plagiarism, such as paraphrases, translations, and idea plagiarism [23].

This paper initiates research on the approach of analyzing mathematical expressions to improve the detection of disguised forms of academic plagiarism. The idea of the approach, which we coin MathPD, has been informally discussed, e.g., at the doctoral consortium at SIGIR'15. However, to our knowledge, no methods or systems pursuing the approach have been proposed thus far.

We present our contributions for this new retrieval task as follows. Section 2 briefly reviews related work on PD and mathematical information retrieval (MathIR) to i) motivate the novelty of the MathPD approach, ii) to identify MathIR approaches that can aid in realizing MathPD, and iii) to show a lack of evaluation resources suitable for MathPD. Section 3 specifies the retrieval task of MathPD and presents findings of a manual investigation of confirmed plagiarism cases. Using the insights of this investigation, the section then describes our initial experiments on developing a first MathPD approach and presents the development and evaluation resources we provide to facilitate research on MathPD. Section 4 summarizes our work and describes our plans for future research.

2 RELATED WORK

Plagiarism detection is a ranked document retrieval task. The objective is to compare an input document to a large reference collection of genuine documents and to return all documents exhibiting similarities above a certain threshold [20]. PD systems typically follow a multi-stage process. The systems first employ computationally efficient methods, such as n-gram fingerprinting, vector space models, or citation analysis, to limit the retrieval space in the candidate retrieval stage [13, 20]. This is followed by exhaustive string comparisons in the detailed comparison stage [20, 22]. Such approaches are limited to finding near copies of text. Identified candidate documents may undergo an optional post processing stage to eliminate false positives, such as correctly quoted text. In the final stage, the retrieval results must be presented to a human examiner who judges the legitimacy of the identified similarities [20]. To detect disguised forms of academic plagiarism, researchers have proposed a variety of monolingual approaches employing semantic and syntactic feature analysis, crosslingual IR methods, and language independent feature analyses, e.g., using images or academic citations [2, 22]. To our knowledge, no approach has analyzed mathematics for PD.

Mathematical information retrieval mainly addresses three tasks: i) mathematical document retrieval, ii) formula retrieval, and iii) document synthesis [9]. The objective in *mathematical document retrieval* and *formula retrieval* is to process a user query consisting of text, mathematical notation, or both and return a ranked list of documents or formulae that match the query. *Document synthesis* describes the composition of a new document from retrieved fragments, which is mainly relevant for educational purposes, but not for MathPD [9]. MathIR research typically distinguishes between three levels of mathematical information: i) presentation, ii) structure, and iii) semantics (ordered by increasing difficulty for being accessed by automated methods). To retrieve mathematical content on the *presentation level*, researchers typically adapted text-retrieval approaches, such as specialized keyword indexes [19]. To retrieve mathematical content on the *structural level*, researchers employed substitution tree data structures [11]. To access the *semantic information* of mathematical content, when it has not bee marked

up explicitly [4], researchers proposed to adapt natural language processing methods to analyze the text that surrounds mathematical expressions [12, 16, 17]. While many research approaches have been proposed to perform similarity assessments on all three levels, Guidi and Sacerdoti Coen find that, as of 2016, only five systems or development frameworks were still available [9].

Both the PD and the MathIR research field offer established standardized evaluation frameworks. In PD, the PAN task series[1] provides a standardized collection of simulated plagiarism cases to benchmark PD systems [15]. However, the PAN collection exclusively contains textual content, which makes it unsuitable for evaluating MathPD approaches. In the MathIR field, the sesqui-annual NTCIR MathIR Task offers a standardized testbed for formula retrieval systems [1], which we adapt (cf. Section 3.3.1).

3 MATH-BASED PLAGIARISM DETECTION

The objective in MathPD is to compare the mathematical expressions in a query document to the expressions contained in documents within a large collection and perform ranked retrieval of all documents with expressions that are similar beyond a chosen threshold. The main differences of MathPD to mathematical document retrieval (cf. Section 2) are the approaches to query formulation and query processing. In mathematical document retrieval, the user formulates the query using a combination of search terms, query language operators, and mathematical features [9]. In MathPD, the query is an entire document. The potential obfuscation of unduly used content by a plagiarist is a threat to retrieval effectiveness that is specific to the MathPD task. Therefore, feature extraction for MathPD is more challenging than for mathematical document retrieval, since the extracted features should be robust against potential obfuscation.

To investigate the characteristics of mathematical plagiarism and derive a gold standard for our experiments, we manually analyzed confirmed plagiarism cases, as we describe in the next section.

3.1 Investigation of Plagiarism Cases

We collected 44 research papers that had been retracted for plagiarism and that involved mathematical content. We found 39 of those papers by reviewing 276 plagiarism cases that Halevi and Bar-Ilan had collected [10] for documents that contain significant amounts of mathematics. We retrieved an additional 3 cases from the blog *Retraction Watch*[2] and another 2 cases from the crowd-sourced project *VroniPlag*[3], which investigates plagiarism allegations.

Four individuals with degrees in computer science (3), physics (1), and mathematics (1) reviewed the cases. To ensure that the reviewers could judge the appropriateness of similar mathematical content, we limited the collection to papers in computer science (6 papers), mathematics (7 papers) and physics (4 papers). Additionally, we included one paper from bioengineering and one paper from medical engineering, for which the retraction notices described the plagiarized mathematics.

Our observations from analyzing the 19 cases that matched the area of expertise of the reviewers are as follows. First, most retracted papers contain significant amounts of mathematical expressions that were similar or identical to expressions in the source document and violated scientific practices. Second, several retracted papers also contained (near) copied text and / or figures. Third, most shared mathematics in the retracted papers closely resembled the mathematics in the source and can be categorized as:

Identical: an exact copy of math in the source document.

Equivalent: equivalent forms, e.g., due to the properties of commutativity, distributivity, and associativity.

Order changes: order of expressions within document differs.

Different presentation: structurally and semantically identical; use of different identifiers, e.g., v_t vs. θ_t, different function names, e.g., $\beta(x)$ vs. $f(x)$, or the use of different operator symbols, e.g., $\odot$ vs. $\oslash$ for min-plus deconvolution.

Splits or merges: a combination of two or more expressions is semantically identical to one expression in the source document ("split"), e.g., term substitutions or intermediate steps in a proof; also opposite relation: "merged" expressions.

Different concepts: different, yet semantically (nearly) identical, concepts, e.g., use of summation over vector components instead of matrix multiplication, discretization of expressions, e.g., transforming integrals into sums, or using multidimensional variables instead of multiple nested single-dimensional variables.

We expect that verbatim and slightly altered copies of mathematics are overrepresented in our sample, because they are easier to recognize for humans and likely identified more frequently. In two retracted papers, we encountered similarities of mathematics that are difficult to recognize and for which legitimacy is hard to assess. In both cases, the authors combined content from two sources. The two retracted papers used their own notation, but followed the order of ideas presented in the sources.

3.2 Detection Approach

This section describes our initial experiments on developing a MathPD approach. Given that most of the similar mathematics we observed in retracted papers closely resembled the mathematics in the source documents, we opted to evaluate the suitability of approaches that compare basic presentational features of mathematical expressions to identify such instances. Presentational features include all elements of mathematical notation, such as identifiers, numbers, operators, and special symbols. We configured our experimental MathPD approach as follows:

Features: We select the essential elements of mathematical notation – identifiers, numbers, and operators – as features.

Feature descriptors: Since most mathematics in retracted documents are slightly altered, approximate feature comparison approaches like vector or set comparisons, histograms, and edit distances seem promising to identify many instances of plagiarized mathematics. Due to their robustness and speed of computation, we use histograms of the frequency of feature instances within a document or document partition. In other terms, we analyze how often a specific identifier, number, or operator occurs.

Granularity: We experiment with two granularities for the feature comparison. First, we use feature descriptors for entire documents. Second, we partition documents based on the number of characters in the document into five equally-sized partitions. The partitioning approach roughly reflects the typical research paper

[1] http://pan.webis.de
[2] http://www.retractionwatch.com
[3] http://www.vroniplag.wikia.com

structure (introduction, related work, approach, evaluation, and conclusion). We add 25% of the length of each partition as overlap to the previous and the following partition.

Feature comparison: For this initial investigation, we opted for a basic pairwise comparison of all feature descriptors to all other feature descriptors instead of devising indexing structures and selection strategies for the feature comparison.

Similarity metrics: We evaluate two distance measures to compute the similarity between feature descriptors (see Equation (1)). First, we compute the distance d_e for any feature e, i.e., identifiers (ci), numbers (cn), and operators (co). d_e represents the absolute difference of the occurrence frequencies $f_{i,e'}$ of a feature instance e', i.e., the number of times a specific identifier, number, or operator occurs in two documents or partitions normalized by the sum of the larger occurrence frequency of each feature instance in either of the two documents or partitions. Second, we use the aggregated distance measure D as the sum of the individual distances d_e.

$$d_e = \frac{\sum_{e'} |f_{1,e'} - f_{2,e'}|}{\sum_{e'} max(f_{1,e'}, f_{2,e'})} \qquad D = \sum_{e \in \{\texttt{ci}, \texttt{cn}, \texttt{co}\}} d_e. \quad (1)$$

We compute the distances for all partition-partition and document-document pairs in the collection and rank documents by increasing distance score. In case of partitions, we only consider the lowest scoring partition pair for each document pair.

3.3 Experiments

This section describes the methodology of our experiments to evaluate the MathPD approaches we described in the previous section. To prevent redundant research and to contribute to establishing shared design and evaluation standards for MathIR systems, we build upon existing MathIR resources. To facilitate future research on MathPD, we make available the source code and data used at:

https://purl.org/mathpd

3.3.1 Test Collection. To create a test collection for our study, we selected ten of the retracted papers we had reviewed manually as the query documents. The papers represent typical instances of similar mathematics we observed and are from disciplines covered by the NTCIR-11 MathIR Task dataset [1], which we use to create the reference collection.

The NTCIR dataset includes approx. 60 million formulae contained in 105,120 scientific papers from the fields computer science, mathematics, physics, and statistics retrieved from the arXiv preprint repository[4]. The papers were converted via LaTeXML[5] to XHTML. Mathematical expressions are included in the XHTML files using parallel Presentation and Content MathML[6]. Since the dataset was developed for formula search, the papers are split up into 8,301,578 smaller search units (paragraphs).

To create the reference collection, we embedded the respective source documents of the ten query documents in the NTCIR dataset. To do so, we used InftyReader [21] to convert the PDFs of the reviewed papers and their source documents to LaTeX. Subsequently, we used the LaTeXML program to convert the LaTeX output of InftyReader to the XHTML format of the NTCIR dataset. We did not split-up the converted documents into paragraphs and used the Content MathML elements `ci`, `cn`, and `co` to distinguish identifiers, numbers, and operators respectively.

Manual checks confirmed a high conversion quality for basic and moderately complex mathematical expressions. For highly complex expressions involving uncommon notation, some manual cleaning of conversion errors was necessary.

3.3.2 Performance Metrics. The ground truth for our test cases is limited to one known item of relevance. As is established practice for known item retrieval, we report the ranks at which the source documents are retrieved, since ranks are most descriptive of retrieval effectiveness [3]. We also report the *Mean Reciprocal Rank* $\text{MRR} = \frac{1}{|Q|} \sum_{i=1}^{|Q|} \frac{1}{\text{rank}_i}$, i.e., the average of the reciprocal ranks at which each query $q \in Q$ retrieves the relevant item. In our case, the ten retracted documents are the queries. The best possible score of 1 is achieved if the source document is retrieved at rank 1 for each test case. Thus, the MRR measure gives an overview of the average retrieval effectiveness of an approach.

3.3.3 Implementation. To develop a MathPD system prototype, we extended the open source MathIR framework *Mathosphere*[7], which uses the distributed *Apache Flink*[8] platform for data handling. Mathosphere was developed to evaluate formula search algorithms in the context of the NTCIR MathIR task [18]. For this purpose, it supports processing the paragraph-sized search units and the query format of the NTCIR task, which allows specifying mathematical expressions, expression patterns that include wildcards, and keywords. We added to Mathosphere a separate pipeline that accepts XHTML documents including MathML markup as input and provides descriptors of mathematical features as output. Developers can use the pipeline to easily access and compare the mathematics of an input document to the documents in a collection.

3.4 Results

Figure 1 shows the ranks at which the feature comparison approaches retrieved the source documents for each of the ten test cases. The two best performing approaches were analyzing the distance for identifiers $d_{\texttt{ci}}$ and analyzing the aggregated distance D. Analyzing the distances for numbers $d_{\texttt{cn}}$ and operators $d_{\texttt{co}}$ on their own yielded very poor results. The frequencies of these features appear to be too unspecific to be useful for MathPD.

When comparing feature descriptors for entire documents, analyzing the distance measure for identifiers $d_{\texttt{ci}}$ performed best, retrieving eight of the ten source documents at rank one (MRR=0.86). This result confirms our impression during the manual analysis that many identifiers in the retracted papers literally matched identifiers of the source documents. Identifier composition seems a valuable indicator of similarity, which in many cases is distinctive enough to retrieve the correct source from the collection of 105,120 documents. The aggregated distance measure D failed to highly rank four of the source documents, because the distance measures for numbers and operators introduced false positives.

When comparing feature descriptors for document partitions, considering the identifier distance $d_{\texttt{ci}}$ retrieved five of the ten

[4] http://www.arxiv.org
[5] http://dlmf.nist.gov/LaTeXML/
[6] https://www.w3.org/Math/
[7] https://purl.org/mathpd
[8] https://www.flink.apache.org/

	full document source retrieved at rank				partitions source retrieved at rank			
Case	**D**	**d_{ci}**	**d_{cn}**	**d_{co}**	**D**	**d_{ci}**	**d_{cn}**	**d_{co}**
C1	3,606	1	27,857	30,784	1	1	85,418	99,201
C2	1	1	88,891	90,962	1	1	12,266	10,277
C3	11,628	2	28,415	3,144	1	16	34,966	5,757
C4	2,581	1	1,950	86	189	6	54,560	18,374
C5	1	1	5,790	22,408	1	6	92,951	16,180
C6	25,498	12	19,862	38,145	7,976	3	24,405	72,687
C7	1	1	4,690	1,627	19,900	1	67,614	14,758
C8	1	1	39,215	11,576	1	1	21,152	9,475
C9	1	1	13,591	35,393	1	1	11,519	32,687
C10	1	1	76,678	30,673	1	1,223	89,703	3,280
MRR								
	0.60	**0.86**	< 0.01	< 0.01	**0.70**	0.57	< 0.01	< 0.01

Figure 1: Ranks at which the feature comparison approaches retrieved the source of a retracted paper.

source documents at rank one. Considering the combined distance D retrieved seven of the ten documents at rank one (MRR=0.70). This result suggests that the pattern of selectively taking over content nearly verbatim in confined parts of a document known in PD [23] also applies to mathematical content. In such cases, including the distance information on numbers and operators, which are too unspecific for the document as a whole, can improve the similarity assessment for more confined parts of a document.

These results are promising and suggest that documents containing slightly altered copies of mathematical expressions can be identified reliably using existing MathIR technology. Future research must show how well more strongly altered instances of plagiarized mathematical expressions can be found.

4 CONCLUSION AND FUTURE WORK

This paper initiates applied research on analyzing mathematics to detect academic plagiarism. By collecting and manually reviewing confirmed plagiarism cases that involve mathematics, we derived a gold standard and insights on the characteristics of plagiarized mathematical content. We created a large-scale test collection, a parallel data processing pipeline, and a first math-based plagiarism detection approaches. The approach successfully retrieved eight of ten test documents from the collection of 105,120 documents at the top rank. These results demonstrate the potential of analyzing mathematics to identify potentially suspicious documents independent of literally matching text.

Our future plans are twofold. First, we seek to research MathPD methods that can also identify more strongly obfuscated instances of plagiarized mathematics. Identifying such instances requires a deeper understanding of the structure and semantics of math content. Adapting structured-based indexing [11] and semantic enrichment [12, 16, 17] approaches proposed for formula retrieval can help to identify such instance. The research challenge is to aggregate the evidence gathered from individual similar formulae to derive a similarity metric for document retrieval. Classical feature overlap metrics, such as the Jaccard coefficient employed by many text-based PD approaches [2, 22], could be a basic aggregation. More sophisticated approaches could analyze patterns of similar formulae, an approach that has been successfully applied for PD using text [5] and academic citations [7, 8]. Transferring scoring heuristics that proved valuable for other PD tasks could also improve MathPD. For example, similar mathematics appearing in introductory sections could be assigned a lower score than mathematics in the methodology section of a paper [7].

Second, we plan to combine MathPD with other PD approaches that analyze literal, syntactical, and semantical text features, figures, and academic citations in a productively usable PD system [14].

ACKNOWLEDGMENTS

This work was partially supported by the German Research Foundation (DFG) grant no. GI 1259/1 and the Czech Science Foundation (GAČR) project no. 17-22224S.

REFERENCES

[1] Akiko Aizawa, Michael Kohlhase, Iadh Ounis, and Moritz Schubotz. 2014. NTCIR-11 Math-2 Task Overview. In *Proc. NTCIR*.
[2] Salha M. Alzahrani, Naomie Salim, and Ajith Abraham. 2012. Understanding Plagiarism Linguistic Patterns, Textual Features, and Detection Methods. In *IEEE Trans. on Sys., Man, and Cybernetic–Part C: Appl. and Rev.*, Vol. 42. 133–149.
[3] Peter Clough and Mark Sanderson. 2013. Evaluating the Performance of Information Retrieval Systems using Test Collections. *Inform. Research* 18, 2 (2013).
[4] Howard Cohl, Marjorie McClain, Bonita Saunders, Moritz Schubotz, and Janelle Williams. 2014. Digital Repository of Mathematical Formulae. In *Proc. CICM*.
[5] Ali El-matarawy, Mohammad El-ramly, and Reem Bahgat. 2013. Article: Plagiarism Detection using Sequential Pattern Mining. *Int. J. of Applied Information Systems* 5, 2 (2013), 24–29.
[6] Teddy Fishman. 2009. "We know it when we see it"? is not good enough: toward a standard definition of plagiarism that transcends theft, fraud, and copyright. In *Proc. Asia Pacific Conf. on Educational Integrity*.
[7] Bela Gipp. 2014. *Citation-based Plagiarism Detection - Detecting Disguised and Cross-language Plagiarism using Citation Pattern Analysis*. Springer.
[8] Bela Gipp and Norman Meuschke. 2011. Citation Pattern Matching Algorithms for Citation-based Plagiarism Detection: Greedy Citation Tiling, Citation Chunking and Longest Common Citation Sequence. In *Proc. DocEng*. 249–258.
[9] Ferruccio Guidi and Claudio Sacerdoti Coen. 2016. A Survey on Retrieval of Mathematical Knowledge. *Mathem. in Computer Science* 10, 4 (2016), 409–427.
[10] Gali Halevi and Judit Bar-Ilan. 2016. Post Retraction Citations in Context. In *Proc. BIRNDL Workshop at JCDL*. 23–29.
[11] Radu Hambasan, Kohlhase Michael, and Corneliu-Claudiu Prodescu. 2014. MathWebSearch at NTCIR-11. In *Proc. NTCIR*.
[12] Giovanni Yoko Kristianto, Goran Topić, and Akiko Aizawa. 2017. Utilizing dependency relationships between math expressions in math IR. *Information Retrieval J.* 20, 2 (2017), 132–167.
[13] Norman Meuschke and Bela Gipp. 2013. State of the Art in Detecting Academic Plagiarism. *Int. J. for Educational Integrity* 9, 1 (2013), 50–71.
[14] Norman Meuschke and Bela Gipp. 2014. Reducing Computational Effort for Plagiarism Detection by using Citation Characteristics to Limit Retrieval Space. In *Proc. JCDL*. 197–200.
[15] Martin Potthast, Benno Stein, Alberto Barrón Cedeño, and Paolo Rosso. 2010. An Evaluation Framework for Plagiarism Detection. In *Proc. ACL*. 997–1005.
[16] Moritz Schubotz, Alexey Grigorev, Marcus Leich, Howard S. Cohl, Norman Meuschke, Bela Gipp, Abdou S. Youssef, and Volker Markl. 2016. Semantification of Identifiers in Mathematics for Better Math Information Retrieval. In *Proc. SIGIR*. 135–144.
[17] Moritz Schubotz, Leonard Krämer, Norman Meuschke, Felix Hamborg, and Bela Gipp. 2017. Evaluating and Improving the Extraction of Mathematical Identifier Definitions. In *Proc. CLEF*.
[18] Moritz Schubotz, Marcus Leich, and Volker Markl. 2013. Querying Large Collections of Mathematical Publications: NTCIR10 Math Task. In *Proc. NTCIR*.
[19] Petr Sojka and Martin Líška. 2011. The Art of Mathematics Retrieval. In *Proc. DocEng*. 57–60.
[20] Benno Stein, Sven Meyer zu Eissen, and Martin Potthast. 2007. Strategies for Retrieving Plagiarized Documents. In *Proc. SIGIR*. 825–826.
[21] M. Suzuki, T. Kanahori, N. Ohtake, and K. Yamaguchi. An Integrated OCR Software for Mathematical Documents and Its Output with Accessibility. In *In Proc. Int. Conf. Computers for Handicapped Persons*.
[22] K. Vani and Deepa Gupta. 2016. Study on Extrinsic Text Plagiarism Detection Techniques and Tools. *J. Engin. Sc. & Techn. Review* 9, 5 (2016).
[23] Deborah Weber-Wulff. 2014. *False Feathers: A Perspective on Academic Plagiarism*. Springer.