Polibits

ISSN: 1870-9044

polibits@nlp.cic.ipn.mx

Instituto Politécnico Nacional

México

Yokoi, Keisuke; Nghiem, Minh-Quoc; Matsubayashi, Yuichiroh; Aizawa, Akiko

Contextual Analysis of Mathematical Expressions for Advanced Mathematical Search

Polibits, vol. 43, 2011

Instituto Politécnico Nacional

Distrito Federal, México

Contextual Analysis of Mathematical Expressions for Advanced Mathematical Search

Keisuke Yokoi, Minh-Quoc Nghiem, Yuichiroh Matsubayashi, and Akiko Aizawa

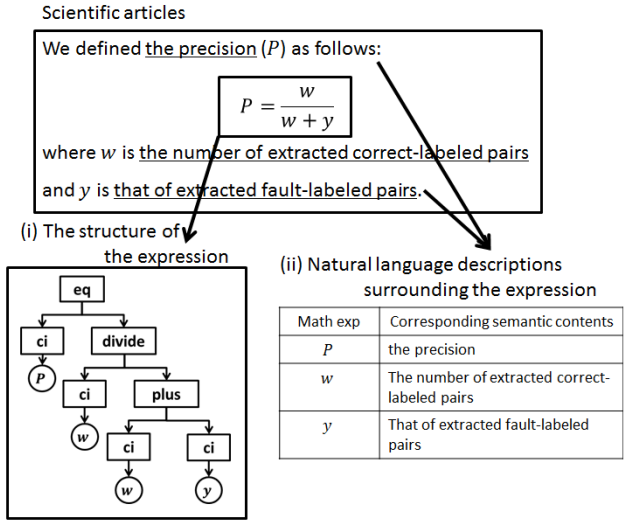
***Abstract*—We found a way to use mathematical search to provide better navigation for reading papers on computers. Since the superficial information of mathematical expressions is ambiguous, considering not only mathematical expressions but also the texts around them is necessary. We present how to extract a natural language description, such as variable names or function definitions that refer to mathematical expressions with various experimental results. We first define an extraction task and constructed a reference dataset of 100 Japanese scientific papers by hand. We then propose the use of two methods, pattern matching and machine learning based ones for the extraction task. The effectiveness of the proposed methods is shown through experiments by using the reference set.**

***Index Terms*—Natural language processing, mathematical expressions, pattern matching, machine learning.**

I. Introduction

MATHEMATICAL expressions often play an essential part in scientific communications. It is not only that they are used for numerical calculations, but that they are used for conveying scientific knowledge with less ambiguity, enabling researchers to precisely define and formalize target problems. They are also used for proving the validity of newly discovered properties. Facilitating cross-document retrieval of mathematical expressions encourages better understanding of the content: what a formula means, why it was used there, or how it was derived. However, regardless of the importance in knowledge-oriented information access, there have been only a few studies on mathematical searches so far. Consequently, with current search engines, most of the mathematical expressions are either totally excluded from the search or only a fraction of those mathematical symbols are indexed and retrieved.

Our purpose is to propose a new framework for a mathematical content search based on semantic analysis of the content. As mathematical expressions are highly abstracted and hard to manage without the accompanying natural language text, we utilize both the structure of expressions and natural language descriptions surrounding them (Fig. 1). It should be noted here, that the existing few studies on mathematical search relied solely on notation similarity of equations and do not use any context information. As far as we know, our research is a first practical attempt to use both the structure of mathematical expressions and the related descriptions within the same framework. We focus initially on a technique for connecting *elements* of mathematical expressions with their names, definitions and explanations, which we collectively call *mathematical mentions*. Examples of elements in this case are variables, functions, or other components that correspond to some newly introduced mathematical concepts in a target document.



Math exp	Corresponding semantic contents
P	the precision
w	The number of extracted correct-labeled pairs
y	That of extracted fault-labeled pairs

Fig. 1. Illustrative example of proposed mathematical content search.

As a target dataset, we selected 100 scientific papers in computer science published by the Information Processing Society of Japan [1]. First, all the mathematical expressions contained in the dataset were converted into Mathematical Markup Language (MathML) format, initially using Math OCR software and then by human check for validating and correcting unavoidable mistakes. Here, MathML is a common standard format for mathematical expressions. All the names and definitions with explicit reference to any of the MathML elements were then also manually annotated. For example, given a statement "Let e be the base of natural logarithm",

Manuscript received November 12, 2010. Manuscript accepted for publication January 10, 2011.

Keisuke Yokoi is with Department of Computer Science, University of Tokyo, Hongo 7-3-1, Bunkyo-ku, Tokyo, Japan (e-mail:kei-yoko@nii.ac.jp).

Minh-Quoc Nghiem is with Department of Informatics, The Graduate University for Advanced Studies, Tokyo, Japan (e-mail:nqminh@nii.ac.jp).

Yuichiroh Matsubayashi is with National Institute of Informatics, Tokyo, Japan (e-mail:y-matsu@nii.ac.jp).

Akiko Aizawa is with Department of Computer Science, University of Tokyo, Hongo 7-3-1, Bunkyo-ku, Tokyo, Japan and with National Institute of Informatics, Tokyo, Japan (e-mail:aizawa@nii.ac.jp).

the phrase "the base of natural logarithm" is annotated as a referrer to the mathematical element "e".

The task we define in this paper is to automatically identify the referrer/referee pairs on the above target dataset. As the majority of mathematics-related descriptions follow a limited number of template expressions, we apply a supervised machine learning framework in our approach. First, frequently appearing description patterns are collected from a separately prepared reference set. Next, using the basic patterns and other linguistic information as features, a support vector machine (SVM) is trained to decide whether given candidate pairs correspond to each other or not. The effectiveness of the proposed method is investigated using the annotated data in our experiments. It is better than the method using pattern matching.

The contribution of our paper is as follows: First, we show the importance of semantic mathematical search and introduce a new framework for extending the current mathematical search systems. For this purpose, we propose the use of a machine learning-based method that identifies the correspondence between mathematical expressions and their natural language descriptions. Second, we manually construct an annotated corpus and evaluate the performance of our method. We show that a supervised machine-learning framework can be used effectively with about 87% precision and 81% recall. Third, we define a new type of information extraction task to identify equivalent relations between natural and formal languages. Our investigation shows that our framework had a satisfactory performance for this type of problem with technical writings.

II. Related Works

We assumed mathematical expressions are represented using Mathematical Markup Language (MathML) [2]. Although it is not widespread, MathML is a worldwide standard defined for mathematical expressions recommended by W3C [3], and as such, is supported by many existing Web browsers. An increasing number of MathML compatible software tools have become available, including editors, mathematics software packages, and translators between MathML and other representations such as TEX or OpenMath; there is also Math OCR software to recognize mathematical expressions printed on paper [4].

Several researchers have done mathematical searches by using MathML and other formal languages for mathematics. Their research can be categorized by their primary goals, *mathematical search* and *mathematical knowledge-base*.

Research on *mathematical search* targets retrieving real-world mathematical documents in digital libraries or on the Web. Since such documents have a great deal of semantic ambiguity, the majority of mathematical search systems calculate similarity between mathematical concepts by considering syntactic information of the formulas. Munavalli et al. analyzed mathematical expressions written in MathML and translated the feature elements into index terms in their MathFind search system [5]. Mišutka et al. also extended the full text search engine with a formula tokenizer that converts formulas into representations of different generalized levels [6]. Adeel et al. generated keywords by using regular expressions for the mathematical equations written in MathML, and threw them to existing search systems as queries [7]. And we also proposed the use of a method for doing a similarity search for mathematical equations based on a distance calculation defined for the tree structure of MathML [8]. On the other hand, research on *mathematical knowledge-base* aims to automatically construct a comprehensive knowledge, or ontology, of mathematics. Therefore, these researches center in extracting rules or relations between mathematical elements from mathematics textbooks or documents. Kohlhase et al. proposed the use of a web-based, distributed mathematical knowledge base where relations between mathematical objects such as symbols, definitions, or proofs were stored in a database and utilized as mathematical facts [9]. Jeschke et al. presented a framework for automatic extraction of mathematical ontology from mathematical texts using natural language processing [10]. Although their framework is remarkable, general, and applicable to many mathematics systems, syntactic analysis of mathematical expressions was still left for future study.

To summarize, existing *mathematical search* studies mainly worked on "syntactic" information of mathematical formulas to identify mathematical concepts useful for indexing. Contrarily, most *mathematical knowledge-base* studies focused on the "semantic" information to extract relations between mathematics related entities. However, "syntactic" disambiguation of mathematical expressions often requires "semantic" interpretation; for example, deciding whether a symbol in an equation is a variable or a function without context information is sometimes difficult. Conversely, "semantic" information alone is often insufficient to identify precise mathematical relationship between the target elements. The final goal of our research is to combine both of the syntactic and semantic features to enable deeper analysis of mathematical expressions. For this purpose, we dedicate ourselves to extracting correspondence between mathematical elements and natural language descriptions.

III. Dataset Construction

Since no annotated corpus is available for MathML documents, we first constructed a dataset that we can use to develop and evaluate our method. The flowchart of the construction is shown in Fig. 2.

First, in the selection phase, we chose 214 papers related to the machine-learning field using a keyword list shown in Table I. We then removed 52 papers with only few mathematical expressions (162 candidates remained), and narrowed the candidate again in terms of relationship with each other, in particular, in the reference network (104 candidates remained) because this is the first step therefore it is desirable that target papers are relative as far as possible. Since we plan to extend

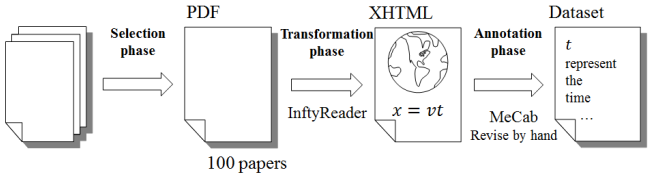


Fig. 2. Flowchart for dataset construction.

our work to mathematical content search in future, we intended our corpus to focus on a specific research topic so that the papers stand on some common mathematical grounding. We expect that with varied authors and years of the publications, sufficient diversity is still maintained for natural language expressions in the corpus.

TABLE I
KEYWORDS USED TO COLLECT RELATIVE (MACHINE LEARNING-RELATED) PAPERS

No.	Keywords (Japanese)	Keywords (English)
1	機械学習	Machine learning
2	教師あり学習	Supervised learning
3	教師なし学習	Unsupervised learning
4	サポートベクタマシン (SVM)	Support vector machine (SVM)
5	ニューラルネットワーク	Neural network

In the transformation phase, the 104 PDF papers were transformed into XHTML format where a mathematical OCR software, InftyReader [4], was used to convert printed mathematical expressions into MathML representations with manual consistency check.

In the annotation phase, we manually enumerated all the pairs of mathematical expressions and corresponding mathematical mentions. First we normalized each sentence (e.g. remove HTML tags) and then split it in morphemes by using a Japanese language morphological analyzer MeCab [11] and then put BIO tags on them to show whether each word correspond to each mathematical expression. For simplication, we only considered compound nouns as candidates for mathematical mentions here. Although mathematical mentions are often expressed as complicate noun phrases with prepositions, adjectives, or adverbs, we annotated only the last compound nouns in the phrases (note that Japanese language is a head-final language). After this process, the four papers without any pairs of mathematical expressions and descriptions were removed from the corpus, which resulted in 100 annotated papers left.

An example sentence in this dataset is shown in Table II. The target sentence can be translated into English as “Here, distribution Exp1 represents the prior probability distribution of the parameter Exp2” where Exp1 and Exp2 represent mathematical expressions. Each target expression is labeled independently using a separated column. In this case, Exp1 has two corresponding mathematical mentions “distribution (分布)” and “the prior probability distribution (事前確率分布)” and therefore these words are put B or I tags. Since only noun phrases are considered as candidates of mathematical mentions in our framework, B/I tags are not put on the phrase “the prior probability distribution of the parameter Exp2” but instead on “the prior probability distribution” as the second mathematical mention of Exp1.

TABLE II
EXAMPLE SENTENCE IN THE DATASET

ID	Morpheme	Tags	
0	ここ (here)	O	O
1	で	O	O
2	、	O	O
3	分布 (distribution)	B	O
4	Exp1	Pred	O
5	は	O	O
6	パラメータ (parameter)	O	B
7	Exp2	O	Pred
8	の	O	O
9	事前 (prior)	B	O
10	確率 (probability)	I	O
11	分布 (distribution)	I	O
12	を	O	O
13	示す (represent)	O	O

IV. METHODS FOR IDENTIFYING CORRESPONDING DESCRIPTION

In this section, we propose the use of two methods for identifying mathematical mentions corresponding to each mathematical expression: pattern matching and one based on machine-learning.

A. Basic Approach

Given a target mathematical expression, the objective here is to find phrases that represent a meaning, definition, or name of the expression. Multiple phrases can be the correct mathematical mentions for a certain mathematical expression. To simplify the problem, we presuppose that: first, all of the mathematical mentions are nouns or compound nouns and second, these mentions co-occur with the target mathematical expression within the same sentence. The problem is then attributed to the binary categorization of each noun phrase in the same sentence with the target mathematical expression.

Our basic approach for this task consists of two steps. First, the sentence containing a target mathematical expression is parsed by a morphological analyzer and the noun phrases are extracted using simple extraction rules; continuous nouns are combined to form a compound noun. Second, for each noun phrase in the sentence, a binary classification is applied to decide whether the phrase is a corresponding mathematical mention to the target or not. Note that each noun phrase in the sentence is processed multiple times if the sentence contains several mathematical expressions. If we take a sentence in Table II as an example, we see that the sentence includes two mathematical expressions (Exp1 and Exp2) and four noun phrases (“here (ここ)”, “distribution (分布)”, “parameter (パラメータ)”, “the prior probability distribution (事前確率分布)”). We, therefore, obtain eight candidate instances to classify

([Exp1, “here”], [Exp1, “distribution”], [Exp1, “parameter”], [Exp1, “the prior probability distribution”], [Exp2, “here”], [Exp2, “distribution”], [Exp2, “parameter”], and [Exp2, “the prior probability distribution”]) in total (Fig. 3).

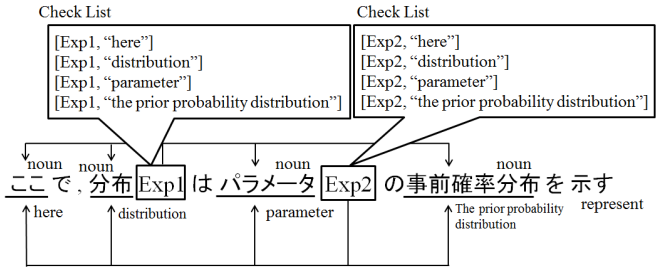


Fig. 3. Classification candidates of a sentence.

B. Pattern Matching Based Method

Our first attempt is based on a naive assumption that scientific papers use a limited number of template expressions to describe the meanings of mathematical expressions. The method based on pattern matching is used to clarify how well the mathematical mentions can be obtained by using a few representative patterns between a mathematical expression and a mathematical mention. We extract most frequent eight patterns from five randomly selected papers in IPSJ Journal (a journal on the field of information science) by hand. Note that these papers are not included in the dataset described in section III, where we choose only literature on machine learning. To keep a generality of the patterns, we did not restrict the topic of the papers. The extracted patterns are shown in Table III.

TABLE III
MOST FREQUENT EIGHT PATTERNS EXTRACTED FROM THE FIVE PAPERS

No.	Patterns
1	[Noun](+[AnotherExp]+“，”+...)+[Exp]
2	[Noun]+“を”(+...)+[Exp]+“と”+⟨する / 表す⟩
3	[Exp]+“を”(+...)+[Noun]+“と”+⟨する⟩
4	[Exp]+“は”(+...)+[Noun]+“で”+⟨ある⟩
5	[Noun]+“と”+“呼び”(+...)+[Exp]+“で”+“表す”
6	[Noun]+“を/は”+[Exp]+“，”
7	[Exp]+“を/は”+[Noun]+“，”
8	[Exp]+“は”(+...)+[Noun]+“を”+⟨示す⟩

Here, [Noun] is a candidate noun, [Exp] is a mathematical expression, A function “$\langle v \rangle$” returns the root form of the verb v, the operator “/” denotes the or function, and “(+...)” indicates that there are zero or more words there. “(+ [AnotherExp] + “,” + ...)” indicates that there are zero or more sequences of another expression and comma. For instance, pattern 1 expresses the case that the [Noun] is the previous word of target [Exp] or the case that there are only some mathematical expressions and commas between the [Exp] and [Noun]. In Fig. 3, both [Exp1, “distribution”] and [Exp2, “parameter”] match pattern 1. The pair [Exp1, “prior probability distribution”] matches pattern 8.

Using these patterns, identification is performed as follows; given a pair of a mathematical expression and a candidate noun, a classifier returns *true* if the pair matches any of the patterns used and *false* if it does not. As a preliminary experiment, we confirmed that a classifier using above eight patterns achieved 85% in F-measure for another five randomly selected papers in the IPSJ Journal. We will evaluate the patterns in a larger dataset in section V.

C. Machine Learning Based Method

We also investigated a supervised learning approach to the task, using the basic patterns above and other linguistic information as clues for classification. As described in subsection IV-A, we formalized a problem as a binary classification for each noun phrase on the condition that the target mathematical expression and automatic morphological analysis are given. Here, we used an SVM-based binary classification model. The features that we used for the classification are shown in Table IV. Every feature in the table is expressed by using a binary value. The features are categorized into four types. First, the eight patterns extracted in the previous subsection are directly used as features. Second, several types of tokens which decide the structures of the sentence are used as clues for determining the relationship between [Noun] and [Exp]. Checking through the tokens between [Noun] and [Exp], this type of features tests the existence of commas and brackets, which decide the syntactic structures, and case markers of subject and object (“は” and “を”), which determine the argument structures between the [Noun] and [Exp]. Intuitively, the likelihood of the relationship between [Noun] and [Exp] may be lower if these features are triggered. Third, neighbor tokens of [Noun] and [Exp] are used as clues. And the last type feature is about dependency analyses. The dependency relation between the [Noun] and [Exp] must provide important clues for determining corresponding pairs.

Using a training set in section III, the L2-regularized L1-loss function is minimized with the Primal Estimated sub-GrAdient SOlver (Pegasos) algorithm [12]. We used the Classias [13] to estimate the parameters.

V. EXPERIMENTS AND DISCUSSIONS

This section gives experiments for evaluating each identification method. We divided the dataset described in section III into three subsets: 60 papers for training, 20 for development and 20 for testing. The training set has $3,867$ positive and $53,153$ negative instances, the development set has $1,267$ positive and $17,440$ negative instances, and the test set has $1,193$ positive and $16,219$ negative instances. We evaluate each model in terms of precision, recall, and F1-measure on the test set. We do not use the training and development set for the method based on pattern matching.

To make a baseline, we used a simple method that returns true iff the target noun phrase is the previous token of the

TABLE IV
FEATURES USED FOR MACHINE LEARNING

Features	Explanations
Pattern (1-8)	Are triggered if target pair matches each of eight patterns.
Another mathematical expression, comma, or opening/closing brackets	Test existance of another mathematical expression, comma, or opening / closing brackets between the target noun and the mathematical expression.
Case markers "は", "を"	Test the existance of case marker "は" or "を" between taret noun and mathematical expression.
Other tokens	Test whether other types of tokens are clipped by targets or not.
Order	Test whether the target noun lies anterior to mathematical expression or not.
Noun	[Noun] itself.
Composition	Triggered if target noun is a compound noun.
Position from [Exp]	Integer numbers indicating a position from [Exp] (..., -2, -1, 1, 2, ...).
Previous/next words of [Noun]	Surface and PoS of previous/next word of target noun.
Previous/next words of [Exp]	Surface and PoS of previous/next word of target mathematical expression.
Nearest verb lemma	Lemma form of verb which first appears after latter target.
Word combination	Combination features using two to six features from features about near words.
Existence combination	Combination features using two to three features from features about existence.
Dependency relation	Tests whether the clause including target noun is dependent/head of that including mathematical expression / both clause have common head.

target mathematical expression. For example in Fig. 3, the baseline outputs two pairs [Exp1, "distribution (分布)"], [Exp2, "parameter (パラメータ)"].

The result of each model is shown in Table V and Table VI. We evaluated the following four models for the machine learning method: without pattern and dependency relation features (*w/o pat&dep*), without pattern features (*w/o pattern*), without dependency relation features (*w/o depend*), and with all the features (*All features*). We use two different evaluation criterions: based on soft and strict matching, respectively. With soft matching based evaluation, automatically extracted noun phrases are considered as *true* if they partially match human annotated ones. On the other hand, with strict matching based evaluation, the extracted phrases are considered as *true* only if they exactly agree with human annotated ones. While the former allows misidentification of the boundaries of the target noun phrases, the latter requires the exact identification.

The highest performance was achieved by machine learning model with dependency analysis features. Essentially, machine learning based models obtain higher precisions and much higher recalls than the method based on pattern matching. It can be seen that *w/o pattern* and *All features* models have no significant difference, which means the existence of pattern features doesn't have much influence on the performance. These results suggest that the manually selected patterns

TABLE V
PRECISION/RECALL/F1-MEASURE OF EACH METHOD
(SOFT MATCHING EVALUATION)

Methods		Development set			Test set		
		Prec.	Recall	F1	Prec.	Recall	F1
Baseline		95.04	57.46	71.62	95.21	61.61	74.81
Pattern Matching		89.46	69.69	78.35	91.80	75.11	82.62
Machine Learning	w/o pat&dep	92.53	82.16	87.04	92.50	85.83	89.04
	w/o depend	92.46	82.24	87.05	92.59	85.83	89.06
	w/o pattern	92.21	83.11	87.42	92.45	86.17	89.20
	All features	92.20	83.03	87.38	92.45	86.17	89.20

TABLE VI
PRECISION/RECALL/F1-MEASURE OF EACH METHOD
(STRICT MATCHING EVALUATION)

Methods		Development set			Test set		
		Prec.	Recall	F1	Prec.	Recall	F1
Baseline		87.99	53.20	66.31	89.90	58.17	70.64
Pattern Matching		82.98	64.64	72.67	87.09	71.25	78.38
Machine Learning	w/o pat&dep	86.13	76.48	81.02	87.35	81.06	84.09
	w/o depend	86.07	76.56	81.04	87.43	81.06	84.12
	w/o pattern	85.60	77.43	81.45	87.32	81.39	84.25
	All features	85.89	77.35	81.40	87.32	81.39	84.25

were implicitly complemented by the combination of features obtained via SVM learning. On the other hand, the dependency feature contributed to the performance improvement. It can be presumed that dependency information successfully captured grammatically generalized and structural patterns which cannot be represented by using sequential patterns.

Note that the performance of the proposed method is upper-limited due to our preprocessing policy of compounds and multi-words described in subsection IV-A. It caused about 6% decrease in the overall performance.

As shown in Table VII, we also evaluated each pattern individually. Pattern 1 shows the best performance and the highest frequency while pattern 5 and 8 scarcely appeared in the dataset. The high precisions of the patterns 1, 2, and 6 exemplify that we extracted the set phrases by using these patterns. On the other hand, the result of pattern 5 and 8 suggests that patterns used for describing the meaning of mathematical expressions may vary depending on the topic/field of the paper. Eventually, we used the model with all these patterns, which achieved the highest performance in the development set as the best pattern model based on matching.

TABLE VII
RESULT OF EACH PATTERN

	Development set			Test set		
No.	Precision	Recall	F1	Precision	Recall	F1
1	93.70	58.72	72.20	94.07	65.13	76.97
2	87.50	1.66	3.25	95.35	3.44	6.63
3	52.94	1.42	2.77	30.00	0.50	0.99
4	76.40	5.37	10.03	80.36	3.77	7.21
5	NaN	0.00	NaN	0.00	0.00	0.00
6	78.57	1.74	3.40	100.00	1.76	3.46
7	55.56	0.79	1.56	66.67	0.50	1.00
8	NaN	0.00	NaN	NaN	0.00	NaN
All	89.46	69.69	78.35	91.80	75.11	82.61

In addition, to evaluate the sufficiency of the dataset, we plot the learning curve of the machine-learning model using all features in Fig. 4. The dataset is comparatively sufficient for machine learning, but, even if we use the maximum size of the training set, the curve still does not converge. Therefore, the gap between pattern matching and methods based on machine learning may increase, with the size of the dataset.

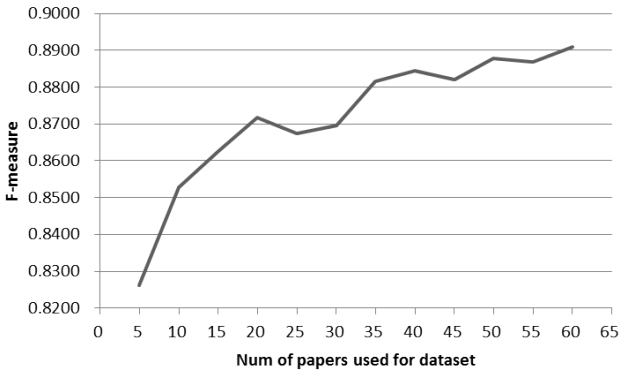


Fig. 4. Learning curve for the result using machine learning based method.

VI. Conclusion

We proposed the use of a method for extracting natural language descriptions associated with mathematical expressions in scientific papers. Our experimental results showed that the proposed machine learning framework works effectively with our dataset. We expect the performance can be further improved by using other information like mathematical expressions' structures. Since this is our first challenge for the mathematical search that includes both the syntactic and semantic aspects, in this paper we only focused here on the information extraction techniques to identify relationships between the two. We plan to incorporate the extracted information into the mathematical search system we already developed and to investigate the potential of the enhancement.

The remaining two important issues are constructing a dataset and determining mathematical mentions. First, the quality of datasets needs to be improved to enable more reliable evaluations. Our validation study showed the limitation of manual annotation particularly for appositions that frequently occur in a target dataset. For example, given a sentence like "*Distribution F is a prior probability distribution*", the apposition "*distribution F*" tends to be overlooked by a human annotator while automatic extraction methods evaluate this more accurately. Such an analysis suggests that the quality of the dataset can be improved by collecting candidates from different competing extraction methods and also by carefully reviewing. Second, we assumed that mathematical mentions are ones of the noun phrases in the same sentences as the target mathematical expressions. However, in real applications, other related descriptions are also useful. For example, given a sentence like "*W is a weight that controls the relative importance of the two operation points*", not only the term "weight" but also the succeeding that-clause is informative for users. This makes the determination of mathematical mentions a more challenging task and requires a reconfiguration of our task and dataset.

Finally, we expect the proposed scheme will be applicable to other languages as well because of the general tendency of mathematical descriptions to follow their characteristic patterns. They will be also addressed in our future study.

References

[1] "Information Processing Society of Japan," http://www.jpsj.or.jp/.
[2] World Wide Web Consortium, "Mathematical markup language (mathml) version 2.0 (second edition)," www.w3.org/TR/MathML2.
[3] "World Wide Web consortium (W3C)," http://www.w3.org/.
[4] M. Suzuki, T. Kanahori, N. Ohtake, and K. Yamaguchi, "An integrated ocr software for mathematical documents and its output with accessibility," in *Computers Helping People with Special Needs*, ser. Lecture Notes in Computer Science. Springer Berlin / Heidelberg, 2004, vol. 3118, pp. 648–655.
[5] R. Munavalli and R. Miner, "Mathfind: a math-aware search engine," in *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, ser. SIGIR '06. New York, NY, USA: ACM, 2006, pp. 735–735. [Online]. Available: http://doi.acm.org/10.1145/1148170.1148348
[6] J. Mišutka, "Indexing mathematical content using full text search engine," in *WDS' 08 Proceedings of Contributed Papers: Part I - Mathematics and Computer Sciences*, 2008, pp. 240–244.
[7] M. Adeel, H. S. Cheung, and S. H. Khiyal, "Math GO! prototype of a content based mathematical formula search engine," *Journal of Theoretical and Applied Information Technology*, vol. 4, no. 10, pp. 1002–1012, 2006.
[8] K. Yokoi and A. Aizawa, "An approach to similarity search for mathematical expressions using MathML," in *Towards digital mathematics library (DML)*, 2009, pp. 27–35.
[9] M. Kohlhase and A. Franke, "Mbase: Representing knowledge and context for the integration of mathematical software systems," *Journal of Symbolic Computation*, vol. 32, no. 4, pp. 365–402, 2001.
[10] S. Jeschke, M. Wilke, M. Blanke, N. Natho, and O. Pfeiffer, "Information extraction from mathematical texts by means of natural language processing techniques," in *ACM Multimedia EMME Workshop*, 2007, pp. 109–114.
[11] T. Kudo, "Mecab: Yet another part-of-speech and morphological analyzer," http://mecab.sourceforge.net/.
[12] S. S. Shwartz, Y. Singer, and N. Srebro, "Pegasos: Primal Estimated sub-GrAdient SOlver for SVM," in *ICML '07: Proceedings of the 24th international conference on Machine learning*. New York, NY, USA: ACM, 2007, pp. 807–814.
[13] N. Okazaki, "Classias: a collection of machine-learning algorithms for classification," 2009. [Online]. Available: http://www.chokkan.org/software/classias/