デジタルライブラリーにおける類似数式検索

東京大学大学院 情報理工学系研究科　横井 啓介
国立情報学研究所　相澤 彰子

発表の流れ

- 数式検索の意義
- MathML
- 検索手法(先行研究)
- 検索手法(提案手法)
- デジタルライブラリへ

デジタルライブラリー 数式検索の意義

- 重要な概念は自然言語のみとは限らない
  - 数式を検索できる重要性
    - この尺度は他にどのような状況で使われる？
    - 他に同じような式変形を適用してる状況は？
  - 数式は従来の自然言語のみの検索では完全に対応できない
    - 公式の名前がわからない場合
    - そもそも名前がついてない式の場合
    - 同じような概念を持つ式を調べたい場合

デジタルライブラリー 数式を扱うために

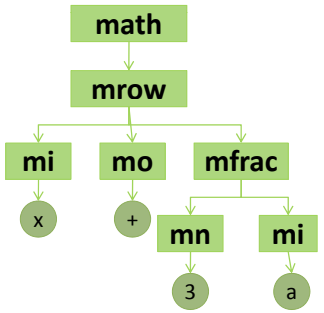
- 数式の独特な構造をいかにして扱うか？
  - 現状ではWeb上の多くの数式は画像形式
  - 数式を扱いやすい形式に変換
    - MathML: Infty Project

MathML
MathMLとは？

- Mathematical Markup Language
- 数式をWeb上に表示するための標準言語
- XMLベースの構造
- 二種類の記法
  - Presentation Markup
    - 数式をWeb上に視覚的に表現
  - Content Markup
    - 数式の意味構造を表現

MathML
Presentation Markup

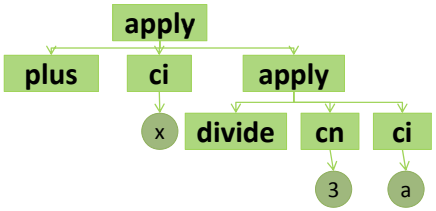
- 例: $x+\frac{3}{a}$




```
<math>
  <mrow>
    <mi> x </mi>
    <mo> + </mo>
    <mfrac>
      <mn> 3 </mn>
      <mi> a </mi>
    </mfrac>
  </mrow>
</math>
```


MathML
Content Markup

- 例: $x+\frac{3}{a}$




```
<apply>
  <plus/>
  <ci>x</ci>
  <apply>
    <divide/>
    <cn>3</cn>
    <ci>a</ci>
  </apply>
</apply>
```


数式検索手法～先行研究

- 数式検索手法には大きく２種類に分かれる
  - 変換ベース
    - 与えられたクエリを変換する
      - 標準化・正規化・自然言語化
  - 構造解析ベース
    - 数式構造を解析・比較
    - 検索システム自体を構築

数式検索手法～先行研究 (1)

- **Math GO! Prototype of A Content Based Mathematical Formula Search Engine**
  (Adeel, et al. 2008)
  - Presentation Markupで書かれたクエリから特徴要素 (キーワード) を抽出
    - 正規表現を利用
  - キーワード群を新たなクエリとして従来の検索システムを利用

数式検索手法～先行研究 (1)

- 以下のような正規表現のルールを用いている
- キーワードは ”Matrix” “Root” などの式の特徴を表す数式要素

Template Rules	Mapped Keyword
<mo>[\(\[]</mo>\s*(<mrow>)?\s*(<mtable>\s*(<mtr>(\s*<mtd>\s*\p{Graph}+\s*</mtd>){2,}\s*</mtr>){2,}\s*</mtable>)\s*(</mrow>)?\s*<mo>[\)\]]+</mo>	Matrix
<m(?:sqrt|root)>\s*(?:(<mrow>\s*)?<mn[^>]*>\d+</mn>\s*(</mrow>\s*)?)+</m(?:sqrt|root)>	Root

数式検索手法～先行研究 (1)

- 最終的に従来の検索システムを利用するので、検索対象とするページの数は膨大
- 自然言語に直して検索するため、検索対象はMathML式に限らない
- 検索要求が曖昧
  - 構造を考えないキーワードだけでは不十分

数式検索手法～先行研究 (2)

- **MathMLを用いた数式検索**　(小田切ら 2008)
  - クエリには独自で定めた数式表現記法を利用
    - 例:　&int(&sin(a), !&cos{})
  - クエリから木構造を構築し、検索対象と木構造マッチング
    - Content Markupを用いて意味構造を重視
  - ランキングは以下のものが有効とされている
    - 「式の大きさ」
    - 「マッチング部分の式全体に対する割合」

数式検索手法～先行研究(2)

- 柔軟なクエリ表現が可能
- 部分的な木構造も構築可能
  - 例: &int(sin{})　(sinを含む積分)
    - Sinの引数は問わない
- マッチングするかしないか、のみ
  - 類似性は考慮していない

数式検索手法～先行研究考察

- 意味のみを考慮

  → 構造を考えていないので曖昧さが残る
- 構造のみを考慮

  → 意味を考えない表層的なマッチングでは
  数式の類似性はわからない
- 意味と構造を共に考える！

数式検索手法～先行研究(3)

- **MathMLを対象とした**
  インデックスに関する調査
- **数式検索システムの実現化に向けて**　(橋本ら2007)
  - Presentation Markup 式の Xpathを検索に用いる
  - 各XPathを、従来の検索システムにおけるwordに対応させてindex化している

数式検索手法～先行研究(3)

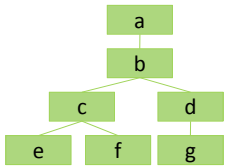
- 転置インデックスを作成できるため、
  検索時間が短い
- 類似数式検索にも適用可能
- Presentation MarkupによるXPathだけでは、
  関数間の関係を取得するのは一苦労

数式検索手法～提案手法 概要

- An Approach to Similarity Search for Mathematical Expressions using MathML
  (Keisuke Yokoi, and Akiko Aizawa 2009)
  - クエリ式に「類似」した数式を返す
    - 構造を把握しやすいContent Markupベース
    - Subpath Set を用いて構文木の類似度を算出
    - Content Markup 記述からより意味の深いSubpathを得るための構造変換
  - 類似度を数値評価するため、関数や変数の揺れに対応した類似検索を実現できる

数式検索手法～提案手法 SUBPATH SET

- Subpath: 根から葉までの経路とその一部
- 構文木の類似度の尺度 (市川ら 2005)
  - 数式に応用してみた



例:
/a, /b, /c, /d, /e, /f, /g
/a/b, /b/c, /b/d, /c/e, /c/f, /d/g
/a/b/c, /a/b/d, /b/c/e, /b/c/f, /b/d/g
/a/b/c/e, /a/b/c/f, /a/b/d/g

数式検索手法～提案手法 SUBPATH SET

- 得られた集合を用いて類似度を計算
  - Jaccard係数 を用いた時の結果が比較的良好

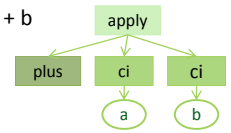
$$\frac{\|S(t_1) \cap S(t_2)\|}{\|S(t_1) \cup S(t_2)\|}$$

  - 先行実験において、Dice係数、Simpson係数、Cosine係数などで類似度算出を行った結果より良いものが得られた

数式検索手法～提案手法 CONTENT MARKUP構造変換

- より深い意味をSubpathに持たせるには？

例: a + b



Subpath:
/apply, /plus, /ci
/apply/plus, /apply/ci

  - 「変数を足す」という情報が無い
  - 「/apply/何か 」のSubpathから得られる情報

    → 「plus 関数を何かに適用」
    「何か関数を変数に適用」

数式検索手法～提案手法
CONTENT MARKUP構造変換

- apply タグとは？
  - 最初の子(関数・演算子)を
    それ以外の子に適用する
  - 関数・演算子が使われる時に限り必ず用いられる
  - 関数や演算子の適用範囲が
    構造としてわかりやすい
  - apply自体は直接的な意味は持たないため、
    検索に関しては木構造が冗長になる

数式検索手法～提案手法
CONTENT MARKUP構造変換

- 構造変換
  - applyタグの最初の子を、親であるapplyタグに置き換え、applyタグを取り除く

例: a + b

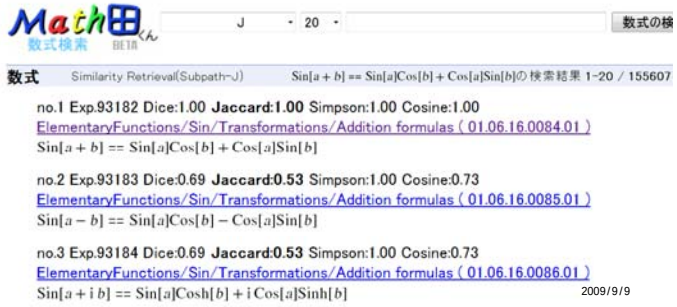
  - より深い情報を持ったSubpathが現れる

Subpath:
/apply, /plus, /ci
/apply/plus, /apply/ci

Subpath:
/plus, /ci
/plus/ci

数式検索手法～提案手法
実装・動作

- CGI アプリケーションとして動作
- 現在はWolfram Functions Siteよりクロールした155,607式を検索対象とする



数式検索手法～提案手法
検索結果

数式 Similarity Retrieval(Subpath-J) Sin[a + b] == Sin[a]Cos[b] + Cos[a]Sin[b]の検索

no.1 Exp.93182 Dice:1.00 **Jaccard:1.00** Simpson:1.00 Cosine:1.00
ElementaryFunctions/Sin/Transformations/Addition formulas (01.06.16.0084.01)
$\mathrm{Sin}[a + b] == \mathrm{Sin}[a]\mathrm{Cos}[b] + \mathrm{Cos}[a]\mathrm{Sin}[b]$

no.2 Exp.93183 Dice:0.69 **Jaccard:0.53** Simpson:1.00 Cosine:0.73
ElementaryFunctions/Sin/Transformations/Addition formulas (01.06.16.0085.01)
$\mathrm{Sin}[a - b] == \mathrm{Sin}[a]\mathrm{Cos}[b] - \mathrm{Cos}[a]\mathrm{Sin}[b]$

no.3 Exp.93184 Dice:0.69 **Jaccard:0.53** Simpson:1.00 Cosine:0.73
ElementaryFunctions/Sin/Transformations/Addition formulas (01.06.16.0086.01)
$\mathrm{Sin}[a + \mathrm{i}\, b] == \mathrm{Sin}[a]\mathrm{Cosh}[b] + \mathrm{i}\,\mathrm{Cos}[a]\mathrm{Sinh}[b]$

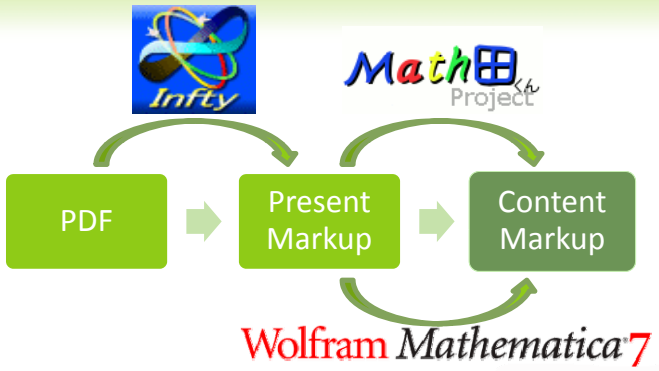
no.4 Exp.108608 Dice:0.68 **Jaccard:0.52** Simpson:0.80 Cosine:0.69
ElementaryFunctions/Cos/Transformations/Addition formulas (01.07.16.0084.01)
$\mathrm{Cos}[a - b] == \mathrm{Cos}[b]\mathrm{Cos}[a] + \mathrm{Sin}[a]\mathrm{Sin}[b]$

no.5 Exp.108607 Dice:0.57 **Jaccard:0.40** Simpson:0.65 Cosine:0.58
ElementaryFunctions/Cos/Transformations/Addition formulas (01.07.16.0083.01)
$\mathrm{Cos}[a + b] == \mathrm{Cos}[b]\mathrm{Cos}[a] - \mathrm{Sin}[a]\mathrm{Sin}[b]$

数式検索手法～提案手法 改良点・展望

- 検索対象拡張のために
  - 全ての式の類似度を計算すると
    計算時間が現実的でなくなる
  - 特徴要素でインデックス・クラスタリングなど、
    あらかじめ計算範囲を絞る必要？
- 変数・定数の中身に関して
  - 現在は全く変数名や定数値は見ていない
  - 表記の揺れに対する柔軟性は保ちつつ、
    ある程度の考慮は必要か
- Content Markup記述
  - 多くのWebページの数式は
    Content Markup記述は含まれていない
  - どれだけ正確なContent Markup記述が作れるか

デジタルライブラリーへ



- Presentation MarkupからContent Markupへの変換は
  情報量が足りない

デジタルライブラリへ 適応にむけて

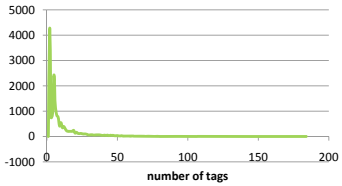
- （情報系の）論文には
  - どのくらい数式があるのか？
  - どのような使われ方をするか？
  - いかにして検索に利用可能なデータを取得するか？
    - Content Markupをどうやって作る？
      - 周辺テキスト

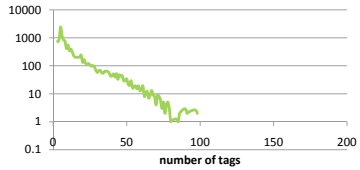
デジタルライブラリへ 論文中の数式調査

- 科学系論文にinftyのOCRをかけて、人手で
  チェックしたものを用いる
- 対象は情報処理学会関係の論文 58編
- 情報学系の文書に数式が
  どれくらい含まれているのか？
- 情報学系の文書に数式が
  どのように用いられているのか？

デジタルライブラリへ 論文中の数式調査

- MathML記述に変換された数式の数
  - 16,889式 (ただし、x など単一変数なども多い)
- Presentation Markup中のタグ数による分布





デジタルライブラリへ 論文中の数式調査

- どのような数式が存在する？
  - 情報分野における数式
    - 必ずしも証明に用いられるわけではない
    - 定義を厳密に記述するための手段など
    - 自然言語が数式中に入ることも多い

デジタルライブラリへ 論文中の数式調査

物質を高温に熱し，徐々に低温にしていく過程（焼きなまし）で，物質がエネルギー準位の低い安定な状態になることがある．SA 法はこの過程を模擬することによって，組合せ最適化問題における評価関数を最小にする系の状態を確率的に探索する近似解法の一つである．その概要を図1に示す1)．ここで，図1の，i, j は系の状態を表す変数，X は解空間，$T(k)$ は k 回目の温度更新が行われた時点での温度，$L(T(k))$ は系の温度を $T(k)$ に更新してから，系が平衡状態に達するまでに必要な7～10行目部分のループ回数を表す．また，関数 $Accept()$は，受理関数と呼ばれているもので様々なものが考えられるが，本論文では，メトロポリスのアルゴリズム 11)

$$Accept(i,\ j) = \begin{cases} 1 & E(j) - E(i) \le 0 \\ \exp\left(-\frac{E(j)-E(i)}{T}\right) & otherwise \end{cases} \quad (1)$$

を用いる．ここで，E は評価関数，T は物理系とのアナロジーから温度と呼ばれる制御変数である．

SA 法の特徴は，改悪方向への解の状態遷移を，式 (1) に示すように確率的に受理することによって，理論上は真の最適解に到達することが保証されていることにある 1), 14)．しかしながら，そのためには温度スケジュールが

$$T_n \le \frac{A}{\log n}$$

（ただし，A は関数 E の凹凸の程度を表す定数，n は温度更新回数を表す整数）に従う必要があり，非現実的に長い計算時間が必要となる．そこで通常は真の最適解への収束性を犠牲にして

$$T_{n+1} = \alpha T_n,\ 0 < \alpha < 1 \quad (2)$$

の形の温度スケジュールを用いることが多い．しかし，温度スケジュールが問題に適したものでなければ解の品質に大きく影響することも報告されており16)，温度スケジュールに関する報告もされているが5),13)，多様な問題に適用可能な万能の温度スケジュールはまだ報告されていない．

デジタルライブラリへ 論文中の数式調査

(2) (n-gram によって推定された) 前2つの単語と品詞

(3) 後2つの単語と品詞

4.3 実験と考察

Penn Treebank WSJ コーパス，RWCP コーパス，京大コーパスの3つのコーパスを用いて，修正学習法による英語品詞タグ付けと日本語形態素解析の実験を行った．

形態素解析の評価には，次のような再現率 (recall)精度 (precision)，F 値 (F-measure) を使用した．

$$\text{再現率} = \frac{\text{解析結果の正解形態素数}}{\text{正解データ中の形態素数}} \quad (12)$$

$$\text{精度} = \frac{\text{解析結果の正解形態素数}}{\text{解析結果の形態素数}} \quad (13)$$

$$\text{F 値} = \frac{2 \times \text{再現率} \times \text{精度}}{\text{再現率} + \text{精度}} \quad (14)$$

4.3.1 品詞タグ付け

コーパスとして，Penn Treebank WSJ コーパス (品詞の数は 50) を使用した．3.3節の品詞タグ付けの実験で使用したものと同じ，1,000,000トークンの訓練データ (41,342文) と，285,000トークンのテストデータ (11,771文) を使った．辞書は訓練データから作成した．確率的モデルとしては ICOPOST release 0.9.0 21) の T3を利用した．これは品詞 trigram モデルによる品詞タグ付けシステムである．二値分類器としては2次の多項式カーネルの SVM を使った．また，3章で説明した One-versus-Rest 法により SVM を適用した品詞タグ付けと TnT の結果との比較を行った．

デジタルライブラリへ CONTENT MARKUPの作り方

- Mathematicaはどこまでやれるか？
  - 実際に先の論文から得られた式にMathematicaを用いてContent Markupに変換してみる
    - どのような場合にerrorがおきるか
    - その他どのような問題があるか

デジタルライブラリへ CONTENT MARKUPの作り方

- うまく変換できた例

$$R \le \min\left(\frac{D^2}{\rho^2}, n\right) + 1 \qquad i + 2$$

$$\sum_{i=1}^{l} y_i \alpha_i = 0 \qquad 0 < t \le l$$

デジタルライブラリへ CONTENT MARKUPの作り方

- うまく変換できなかった例
  - 式の区切り $a, \quad = (a_1 b_1 + a_2 b_2 + 1)^2$
  - ベクトル $a = (a_1, a_2)$
  - その他

$\|\omega\|$ $\qquad f(x) = \mathrm{sgn}\left(\dot{wcx} + b\right)$

R_{emp} $\qquad \dot{l} + 2$

デジタルライブラリへ CONTENT MARKUPの作り方

- 知識を得ることで回避できる問題も多い
  - 複数文字による関数
  - 変数の定義
- どのように知識を得るか、が問題