LaSer : search engine for mathematical formulae

Prithwish Mukherje, Biswajit Paria, Agnivo Saha, Haritabh Singh, Anurag Anand, Sandesh C, and Mayank Singh*

Department of Computer Science, IIT, Kharagpur

E-mail: mayank4490@gmail.com

Abstract

We present a search engine for mathematical formulae. The MathWebSearch system harvests the web for content representations (currently MathML) of formulae and indexes them using unigrams, bigrams and trigrams. The query for now is limited to Latex format only and can and will be extended to free form queries.

Introduction

As the world of information technology grows, being able to quickly search data of interest becomes one of the most important tasks in any kind of environment, be it academic or not. This paper addresses the problem of searching mathematical formulae from a semantic point of view, i.e. to search for mathematical formulae not via their presentation but their structure and meaning.

*To whom correspondence should be addressed

Related Work

We have seen that early work has been done in this fields, allowing users to search for mathematical equations. There are multiple sites available such as http://latexsearch.com/, http://uniquation.com/en/, http://www.searchonmath.com/ that provide equation searching features. Most of the work done involves converting the equations available into xmls, sustitution trees and indexing via various open source software. On other hand we are extracting various features other than simple xmls of the equations that help in increasing the efficiency of the system, as described in the following sections.

Outline of workflow

1 Convert Latex Formula to xml

The conversion from Latex formulae to xml has been performed using MathML

e.g : $\lambda = 1$ is coverted to the following xml

<?xml version="1.0" encoding="UTF-8"?>

<m:math xmlns:m="http://www.w3.org/1998/Math/MathML" display="block">

<m:mrow>

<m:mi> λ </m:mi>

<m:mo¿=¡/m:mo>

<m:mn¿1¡/m:mn>

</m:mrow>

</m:math>

2 Simplifying using equations and generating additional xmls.

The simpification has been done using python library sympy and MathML. e.g : ((a + b) + c) is simplified to a + b + c and then xml is generated as above.

3 Number Normalization and Unicode Normalization.

Here we are coverting all unicode variants of a letter back to the same letter to increase search results.

Add examples.

4 Operator Grouping

Add description

5 Unigram + Bigram + Trigram Feature Extraction

Here the xml is treated simply as string and the unigram, bigram and trigram features are extracted.

Add examples and description

6 TF-IDF weights calculation

Each equation is converted into a vector representation using the extracted unigram, bigram and trigram features

The $tf_{weight} = 1 + log_{10}tf$, $idf_{weight} = 1 + log_{10}(N/df)$

7 Query Normalization

Latex query given as input is also normalised as mentioned above to reduce unicode variants. The resulting equation is also simplified. Finally the query is converted into a high dimensional vector space using the extracted features mentioned above.

8 Cosine Similarity

Currently cosine similarity is being used as the metric to determine the similarity between query vector and the indexed vectors.

9 Backend and frontend

The Backend is running on a python server and webpage is used as the frontend

The LaSer Application

Add snapshots of UI here

Conclusions and Future Work

- Support free form queries as input.
- Retrieve on basis of structural similarity, for example: $x + y = 1$ and $a + b$ and $(2 * X - 1) + (2 * Y)$ are all structurally similar.
- Retrieve on simplification for example $x*x = 1$ and $x^2 = 1$ are same after simplification.
- Support search by names of popular equations.

References

1. A Search Engine for Mathematical Formulae, Michael Kohlhase and Ioan A. Sucan
2. $5e^{x+y}$: A Math Aware Search Engine(for CDS), Master Thesis Project, Arthur Oviedo, Supervisors: CERN: Nikolaos Kasioumis, EPFL: Karl Aberer