

Spectral Clustering for Axiom Selection

Zishi Wu

Department of Computer Science
University of Miami

May 17, 2020

UNIVERSITY
OF MIAMI



Outline

- 1 Introduction
- 2 Related Work
- 3 Methodology



Introduction

Definitions

- What is Automated Theorem Proving (ATP)?
- *Logical formulae* are statements about a domain:
- Show that the *conjecture* is a *logical consequence* of the axioms (a.k.a premises).
- Applications:
 - Formal verification of software - Compilers (e.g. gcc, llvm)
 - Formal verification of hardware - CPU (e.g. 1994 Intel Pentium floating-point division bug)
 - Interactive proof assistants for mathematics (e.g. Isabelle, Mizar)

Introduction

Example

- Axiom 1: *All men are mortal.*
- Axiom 2: *Socrates is a man.*
- Conjecture: *Socrates is mortal.*



Logical Consequence

Logical Consequence

- Every model of the axioms is a model of the conjecture.
- A set of axioms has a *model* if there is an *interpretation* (assignment of boolean values) to the axioms such that the conjunction of the axioms evaluate to *True*.
- If we list all interpretations of N formulae on a truth table, we get 2^N rows. This search space grows exponentially.
- The faster method is to show that the union of the axioms and the negation of the conjecture is *unsatisfiable*. $Ax \cup \neg C = \emptyset$
- In other words, if no model of the axioms is a model of the negated conjecture, then all models of the axioms are models of the conjecture.

Problem Statement

Problem Statement

- A *large-theory* problem consists of a conjecture to be proven, and a large number of axioms to be considered.
- However, the solution set(s) usually consist of a few axioms.
- How do we select the necessary axioms? This is known as the problem of *premise selection*.



Benchmark Data

MPTP2078 Dataset

- Thousands of Problems for Theorem Provers (TPTP) [Sut17]
 - Standard set of test problems.
- Benchmark dataset of 2078 problems known as the Mizar Problems for Theorem Provers (MPTP2078) [AHK⁺14].
 - Encodes problems from the Mizar Mathematical Library (MML) of formalized mathematics into first-order logic form.
- There are two versions of each problem:
 - Bushy = smaller version (3 to 40 axioms, 1 to 15 needed)
 - Chainy = larger version (10 to 500 axioms, 2 to 119 needed)
- Premise selection performance compared to state-of-the-art Automated Theorem Provers:
 - E [Sch13]
 - Vampire [KV13]

Graph Methods

Minimal Proof Dependency

- Alama et al. [AHK⁺14]. constructed a knowledge base of minimal proof dependencies that a problem depends on and trained kernel-based machine learning methods on a feature matrix representation of that knowledge base.
- Minimal dependencies encoded as an adjacency matrix, where the $(i, j)^{th}$ cell of the matrix has a value of 1 if the i^{th} formula of a problem is used in the proof of the j^{th} formula of a problem, and 0 otherwise.
- Formulate premise selection as a *ranking problem*:
 - Rank the axioms by how likely they are to prove a conjecture (e.g. shared minimal dependencies).
 - Choose a threshold value.
 - Select all axioms whose score is above that threshold.

Related Work

Hutchinson Distance

- Least Generalized Ancestor [Hut97]

Methodology

Data Representation

- NOTE: this should be a comprehensive summary of the entire methodology, not details of one part
- Qinghua designed a dissimilarity metric [LXH17]
- Problem is converted into an undirected fully-connected graph
 - Vertices $V = \{\text{Axioms} \cup \text{Conjecture}\}$
 - Edges $E =$ dissimilarity weights between vertices

Graph Theory

Spectral Graph Theory [Chu97]

- Adjacency matrix A consists of similarity values between vertices
- Degree matrix D is a diagonal matrix where the i^{th} element is the sum of the elements of the i^{th} column of A
- Un-normalized Graph Laplacian matrix
 - $L = D - A$
- Normalized Graph Laplacian matrix contains *features* of the graph
 - $L_{norm} = I - (D^{-1/2} L D^{-1/2})$

Example Graph

Calculate Normalized Laplacian Matrix

- Adjacency, Degree, and Un-normalized Graph Laplacian

$$A = \begin{bmatrix} 0 & 3 & 0 \\ 3 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \quad D = \begin{bmatrix} 3 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad L = \begin{bmatrix} 0 & 3 & 0 \\ 3 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

- Normalized Graph Laplacian

$$L_{norm} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} - \begin{bmatrix} \frac{1}{\sqrt{3}} & 0 & 0 \\ 0 & 1/2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 3 & 0 \\ 3 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \frac{1}{\sqrt{3}} & 0 & 0 \\ 0 & 1/2 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Selection Method

Spectral Clustering [vL07]

- Given graph $G = (V, E)$
- Partition vertices in V into k clusters: C_1, C_2, \dots, C_k
- Denote the cluster containing the conjecture as C_C
- Problem may have more than one set of solutions
- Successful conjecture cluster only needs to contain one solution

Spectral Clustering Algorithm

A Tutorial on Spectral Clustering [vL07]

- 1 Construct a weight matrix W (i.e. adjacency matrix A)
- 2 Compute the normalized Laplacian matrix L_{norm} from W
- 3 Compute the first k eigenvectors v_1, \dots, v_k of L_{norm} and construct a feature matrix U from those eigenvectors
- 4 For $i = 1, \dots, n$, let p_i be the feature vector for the i^{th} vertex, corresponding to the i^{th} row of U
- 5 Cluster the vertices based on their feature vectors into k clusters: C_1, C_2, \dots, C_k

Spectral Clustering Algorithm

K-Means Initialization Problem

- ⑤ Cluster the feature vectors into k clusters: C_1, C_2, \dots, C_k
 - Each run of k -means chooses a different set of initial centroids for the k clusters
 - Results in different clusterings each run
 - We need a deterministic way of clustering that doesn't change over multiple runs of k -means



J. Alama, T. Heskes, D. Külwein, E. Tsivtsivadze, and J. Urban.
Premise Selection for Mathematics by Corpus Analysis and Kernel
Methods.

Journal of Automated Reasoning, 52(2):191–213, 2014.



F. R. K. Chung.

Spectral Graph Theory.

American Mathematical Society, 1997.



A. Hutchinson.

Metrics on Terms and Clauses.

In M. van Someren and G. Widmer, editors, *Proceedings of the 9th European Conference on Machine Learning*, number 1224 in Lecture Notes in Artificial Intelligence, pages 138–145. Springer-Verlag, 1997.



L. Kovacs and A. Voronkov.

First-Order Theorem Proving and Vampire.

In N. Sharygina and H. Veith, editors, *Proceedings of the 25th International Conference on Computer Aided Verification*, number 8044 in Lecture Notes in Artificial Intelligence, pages 1–35.
Springer-Verlag, 2013.



Q. Liu, Y. Xu, and X. He.

New terms metric based on substitutions.

In *2017 12th International Conference on Intelligent Systems and Knowledge Engineering (ISKE)*, pages 1–6, 2017.



S. Schulz.

System Description: E 1.8.

In K. McMillan, A. Middeldorp, and A. Voronkov, editors, *Proceedings of the 19th International Conference on Logic for Programming, Artificial Intelligence, and Reasoning*, number 8312 in Lecture Notes in Computer Science, pages 477–483. Springer-Verlag, 2013.



G. Sutcliffe.

The TPTP Problem Library and Associated Infrastructure. From CNF to TH0, TPTP v6.4.0.

Journal of Automated Reasoning, 59(4):483–502, 2017.



Ulrike von Luxburg.

A tutorial on spectral clustering.

Statistics and Computing, 17(4):395–416, Dec 2007.