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
4. Results
5. Conclusion

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 way 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]

Ranking Problem

Ranking Problem

- Alama et al. [AHK+14] formulated premise selection as a *ranking problem*:
  - Rank the axioms by how likely they are to prove a conjecture, based on some kind of user-defined similarity metric.
  - Choose a threshold value.
  - Select all axioms whose score is above that threshold.
- Also a *classification problem*: given a feature matrix consisting of the similarity values between formulae in a problem, train different machine learning methods on the data to classify if an axiom is *likely* or *unlikely* to be in the proof.

Related Work

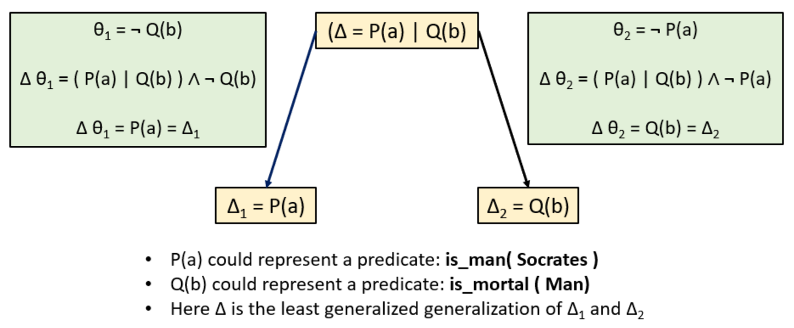
Extended Hutchinson Distance

- To construct an adjacency matrix, we require a measure of similarity or dissimilarity between each pair of nodes in a graph.
- Liu [LXH17] proposed a dissimilarity metric between two terms Δ_1 and Δ_2, that extends the Hutchinson distance [Hut97].
- Calculated by finding the Least Generalized Generalization (*lgg*) between two formulae. A term Δ is the *lgg* of Δ_1 and Δ_2 iff
  - There are substitutions θ_1 and θ_2 such that $\Delta\theta_1 = \Delta_1$ and $\Delta\theta_2 = \Delta_2$.
  - There exists no term Δ' and substitutions σ, σ_1 and σ_2 such that $\Delta\sigma_1 = \Delta'$ and $\Delta\sigma_2 = \Delta'$.

Related Work

Least Generalized Generalization Example

- Note that the substitutions θ_1 and θ_2 are not limited to a single substitution rule. They can also consist of multiple substitution rules occurring one after the other.



Related Work

Extended Hutchinson Distance

- If there is no Least Generalized Generalization Δ between two terms Δ_1 and Δ_2, then their extended Hutchinson distance is ∞. Formally, $dissimilarity(\Delta_1, \Delta_1) = \infty$ iff $lgg(\Delta_1, \Delta_2) = \emptyset$.
- If there exists a Least Generalized Generalization Δ between two terms Δ_1 and Δ_2, then the Extended Hutchinson Distance says:
  - More total substitutions required (from the *lgg* to both terms) equates to a higher dissimilarity score.
  - Fewer total substitutions required equates to a lower dissimilarity score.
- Note that this is a simplication of Qinghua's actual metric, which is more complicated.
- As expected, for any term Δ_1, $dissimilarity(\Delta_1, \Delta_1) = 0$

Graph Matrices

Graph Laplacian Matrix

- Von Luxburg [vL07] describes this technique to cluster vertices in an undirected graph according to some user-defined similarity metric.
- Adjacency matrix A consists of edge weights between pairs of vertices that represent their similarity.
- 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

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

Methodology

Graph Representation

- Used Extended Hutchinson distance to get dissimilarity values of all pairs of vertices in a problem.
- Let Φ_1 and Φ_1 represent two formulae with dissimilarity of $dsim(\phi_i, \phi_j)$. We convert dissimilarity values into similarity values using the following formula:
  - $maxdsim(\mathcal{F}) = max_{\phi_i, \phi_j \in \mathcal{F}}(dsim(\phi_i, \phi_j) \neq \infty)$
  - $sim(\Phi_1, \Phi_2, \mathcal{F}) = \max(0, maxdsim(\mathcal{F}) - dsim(\Phi_1, \Phi_2))$
- The logic problem becomes a fully-connected and undirected graph
  - Vertices $V = \{\text{Axioms} \cup \text{Conjecture}\}$
  - Edges E = similarity values between vertices

Spectral Clustering Algorithm

Spectral Clustering

- A Tutorial on Spectral Clustering [vL07]
  1. Construct a weight matrix W (i.e. adjacency matrix A) containing similarity values of the edges of the graph.
  2. Compute the normalized Laplacian matrix L_{norm} from W.
  3. Compute the first k eigenvectors $v_1, \ldots, v_k$ of L_{norm} and construct a feature matrix U from those eigenvectors.
  4. For $i = 1, \ldots, n$, 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: $C_1, C_2, \ldots, C_k$. Denote the cluster containing the conjecture as C_C.
- A problem may have more than one set of solutions, but the conjecture cluster needs only to contain the axioms for one solution to be deemed a successful clustering.

K-Means Clustering

K-Means Clustering Problems

5. Cluster the vertices based on their feature vectors into k clusters: $C_1, C_2, \ldots, C_k$.
    - 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.
- We also need to figure out the optimal value for the parameter k, the number of clusters.

K-Means Clustering

Solution to Initial Centroids Problem

- Solution description here.

K-Means Clustering

Solution to Number of Clusters Problem

- Solution description here.

Prediction of Optimal Number of Clusters

Median Regression

- Method description here.

Evaluation Metrics

Evaluation Metrics

- Selectivity
- Precision Score
- Adequacy

Results

Predicted Number of Clusters

- Result description here.

Results

Evaluation on Bushy and Chainy Problems

- Result description here.

Conclusion

Conclusion

- Spectral clustering method extensions.
- Dissimilarity method extensions.
- Mention paper submitted to conference name.
- Thanks to Qinghua Liu, Zihao Wang, and Dr. Geoff Sutcliffe.

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.