Evolving Augmented Graph Grammars for Argument Analysis

Collin F. Lynch
North Carolina State University
Raleigh, North Carolina, USA
cflynch@ncsu.edu

Linting Xue
North Carolina State University
Raleigh, North Carolina, USA
lxue3@ncsu.edu

Min Chi
North Carolina State University
Raleigh, North Carolina, USA
mchi@ncsu.edu

ABSTRACT

Augmented Graph Grammars are a robust rule representation for rich graph data. In this paper we present our work on the automatic induction of graph grammars for argument diagrams via EC. We show that EC outperforms the existing grammar induction algorithms gSpan and Subdue on our dataset. We also show that it is possible to augment the standard EC process to harvest a set of diverse rules which can be filtered via a post-hoc Chi-Squared analysis.

Keywords

Evolutionary Computation, Machine Learning, Graph Data, Augmented Graph Grammars

1. INTRODUCTION

Complex graph structures have become increasingly important to data analysis. There has been increased interest in analyzing social networks, chemical structures, user-system interaction logs, and other complex relational information. One of the primary goals of graph mining is to identify key substructures that signal important errors or chemical properties, or which can be used to classify graphs as indicative of good or poor performance. These substructures make it possible to abstract the graph structures and can act as induced features for secondary analysis.

Augmented Graph Grammars (AGG) are a robust formalism for rules about graphs [3]. They are an analogue to string grammars that allows for graph structures, complex element types and constraints and heterogeneous rules that incorporate rich node and arc content. AGGs are well-suited to analyzing rich semi-structured data such as Argument Diagrams which are a graph-based representation for argumentation [4]. Argument diagrams reify key components of arguments such as hypotheses, claims, and supporting evidence with textual content. Argument diagrams have grown increasingly popular in education, computer-supported collaborative work, and other domains.

GECCO'16 Companion July 20-24, 2016, Denver, CO, USA

ACM ISBN 978-1-4503-4323-7/16/07.

DOI: http://dx.doi.org/10.1145/2908961.2908994

Existing graph-grammar induction algorithms are limited in scope and are ill-suited to rich graphs. In this paper, we discuss our ongoing work on the application of Evolutionary Computation (EC) to grammar induction. EC-based search can examine a wider range of the search space and identify higher-performing rules than existing methods. We augmented the existing EC paradigm to harvest diverse rules over the course of the run and used Chi-Squared (χ^2) filtering to narrow the final population to a novel subset. We then compared the performance of the EC-generated rules to the existing gSpan [6] and Subdue [2] algorithms.

2. EXPERIMENT

In this work we harnessed the flexibility of EC to induce AGGs directly from an existing dataset of 104 student-produced argument diagrams [4]. This dataset was collected in a prior study on argument analysis and was graded for argument quality on a scale of -5 to 5. Our long-term goal is to induce fully-augmented hierarchical grammars that can represent classes of argument diagrams and can deal with the textual content. For the present work, however, we restricted the search to single rules that relied on ground node and arc types, avoiding both hierarchical variables and negation. This ensures that the EC system will examine a similar search space to the baseline algorithms. We did however permit the EC algorithm to induce disjoint rules, rules that include disconnected subgraphs, which are beyond the scope of our baseline algorithms.

For each experimental run we used a population of 100 independent rules and ran for 1,000 generations. In each generation, we cloned the top 10 individuals directly into the next generation under elitism, and we selected 10 individuals for point mutation and copied the results over. The remaining members of the new population were produced via crossover. Our fitness function was based upon a non-parametric correlation between the frequency with a rule matched the target graph, and the expert-assigned score. We used this same metric in our prior work on the use of argument diagrams to predict essay grades [4]. We used basic point mutation that could add, delete or modify individual elements and connecting arcs. In order to preserve stability we used a matrix crossover algorithm based upon the work of [5]. On each generation we harvested all rules where $\rho \geq 0.18$ and preserved them for later analysis.

Our baseline grammar-induction algorithms were gSpan [6] and Subdue [2]. GSpan is a frequent subgraph algorithm based upon heuristic graph walks. Subdue executes a compression-based beam search for good subgraphs. When using the former we exported all graphs with frequency $\geq$ 1% and $\rho \geq 0.18$. For the latter we used the algorithm in supervised learning mode and treated all graphs with a score ≥ 0 as positive examples and the remainder as negative. We then collected the best 12 features with $\rho \geq 0.18$. Prior experiments have shown that more than additional features fell below the threshold.

We filtered each set of candidate rules using a χ^2 test of independence. This is a statistical test that measures divergence from the expected distribution assuming that one feature occurs independently of the others [1]. A p-value ≤ 0.05 for the χ^2 test led us to reject the null hypothesis that two variables are wholly independent and conclude that the variables are significantly correlated in our dataset. In order to construct the contingency table for this analysis we treated each rule as a categorical test and collected a binary vector over the population indicating whether or not a rule mapped to any subset of a given diagram.

We then performed a greedy filtering process. The rules were ranked by ρ, and each rule was compared to its better-correlated peers. For any pair of rules where the χ^2 correlation was statistically-significant ($p - value \leq 0.05$), we discarded the lower-ranking member. While this process discarded some promising candidates it allowed us to rapidly winnow the data down to a (more) tractable subset.

3. RESULTS

Table 1 shows our overall experimental results. The "Count" column indicates the number of rules that were initially generated by each of the algorithms, and the number of independent rules after filtering. The "ρ value" columns show the highest and lowest ρ values for the best five rules from each algorithm. As Table 1 illustrates, the EC algorithm generated the largest number of candidate rules, but the number dropped from 82 to 6 after filtering. Subdue and gSpan, by contrast, generated far smaller sets of rules (no more than 12) and dropped to no more than 3 after the χ^2 filtering.

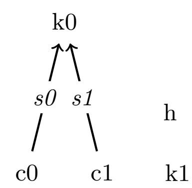
As the table illustrates, EC produced the best-correlated rules both before and after filtering. Prior to the filtering the top 5 rules were all close analogues to one another and had quite similar performance. After filtering, however, the low performance dropped substantially as the rules varied. Thus it is clear that the filtering was successful in closely-matching cases for all three algorithms.

The best of the EC-induced rules is shown in Figure 1. It matches a substructure containing two independent citations that jointly support a shared claim node alongside a

Table 1: Number of unique rules generated by the induction algorithms and the Spearman's correlation range of top five best rules before and after χ^2 filtering.

Ruleset		Count	ρ value	
			Best	Worst
EC	Raw	82	0.37	0.36
	Trimmed	6	0.37	0.19
Subdue	Raw	11	0.28	0.22
	Trimmed	3	0.28	0.18
gSpan	Raw	12	0.35	0.26
	Trimmed	3	0.35	0.23

(EC)



$$\left\{\begin{array}{c} k*.Type = \text{“}claim''\\ h.Type = \text{“}hypothesis''\\ c*.Type = \text{“}citation''\\ s*.Type = \text{“}support'' \end{array}\right\}$$

Figure 1: Best rules induced by EC

separate hypothesis and claim. Because the rules do not include explicit negation there may be additional arcs between the nodes in the matched graphs other than $s0$ and $s1$. Conceptually this structure matches cases where an arguer has identified multiple sources of support for their shared claim and where that support structure is a subset of a larger argument with at least one hypothesis and an additional claim. In the context of our study assignment this would indicate that the students sought to validate their claims with multiple pieces of evidence rather than using a single source as is often the case.

4. CONCLUSIONS

In this work, we showed that it is possible to apply Evolutionary Computation to induce Augmented Graph Grammars for rich graphs such as Argument Diagrams. Our EC algorithm is able to induce higher-performing rules than Subdue or gSpan. We also found that EC is able to induce disjoint rules, a key limitation of existing algorithms, and that those rules outperform the best connected examples. We also show that it is possible to augment the existing EC paradigm by harvesting rules and filtering them using a χ^2 test of independence to yield a heterogeneous subset.

5. REFERENCES

[1] P. Dalgaard. *Introductory Statistics with R*. Springer Verlag New York Inc., 2002.

[2] L. B. H. Diane J.Cook and N. Ketkar. Unsupervised and supervised pattern learning in graph data. In *Mining Graph Data*, chapter 7, pages 159–180. John Wiley & Sons, Inc, Hoboken, New Jersey, 2007.

[3] C. F. Lynch. Agg: Augmented graph grammars for complex heterogeneous data. In S. G. Santos and O. C. Santos, editors, *Proc. of the Workshops held at Educational Data Mining 2014*, volume 1183 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2014.

[4] C. F. Lynch and K. D. Ashley. Empirically valid rules for ill-defined domains. In J. Stamper and Z. Pardos, editors, *Proc. of The 7^{th} International Conference on Educational Data Mining (EDM 2014)*. International Educational Datamining Society IEDMS, 2014.

[5] S. Stone, B. Pillmore, , and W. Cyre. Crossover and mutation in genetic algorithms using graph-encoded chromosomes. *Unpublished*, March 2011.

[6] Y. Xifeng and H. Jiawei. gspan: Graph-based substructure pattern mining. In *Proc. of the IEEE International Conference on Data Mining (ICDM 2002)*, pages 721–724. IEEE, 2002.