Distributed Multi-Relational Association Rule Mining for Ontological Data using Evolutionary Algorithms

Saikat Roy and Vijayesh Kumar Das
Institute for Informatics
University of Bonn
Distributed Big Data Analytics Lab (SS 2018)

***Abstract*—This report was created as part of the `Distributed Big Data Analytics Lab` in the Summer Semester, 2018 at the University of Bonn. The aim of the project was to explore Distributed Multi-Association Rule Mining on Ontological Data using *Spark* and *Scala*. This report discusses methods explored and the results of the project.**

I. Introduction

This report was created as part of the `Distributed Big Data Analytics Lab` in the Summer Semester, 2018 at the University of Bonn. The aim of the project was to explore Distributed Multi-Association Rule Mining using Evolutionary Algorithms. We discover hidden knowledge patterns in the form of multi-relational association rules by utilizing the evidence coming from evolving assertional data. An ontology may be incomplete, noisy and sometimes inconsistent, due to which ontologies and assertions may be out of sync. Hence, we use evolutionary algorithms to compute multi relational association rules.

Evolutionary algorithms typically suffer from the a prohibitive *time complexity* on fitness function evaluations. With the massive size of ontological databases such as dbPedia or Freebase for example, the issue of speeding up an association rule discovery algorithm becomes paramount. To achieve this, the existing evolutionary rule mining algorithm used in this work was augmented with a distributed fitness function calculator with the aim of reducing the time taken to *train* the algorithm for association rule discovery in large ontological datasets.

II. Problem Definition

The problem is defined as association rule mining in ontological knowledge base (KB) using an evolutionary algorithm. Formally the problem may be defined as:

Given an ontological KB κ, a frequency threshold θ_f, a head coverage threshold θ_{hc} and a confidence improvement threshold θ_{ic}, find a set of association rules, w.r.t. θ_f.

Association rules are defined as Sematic Web Rule Language (SWRL) rules in this work, which may be defined as a logical implication between an antecedent and a consequent of the form $B_1 \wedge B_2 \wedge ... \wedge B_N \longrightarrow H$ where B_i represents body predicates and H represents the head.

III. Approach

An association rule is defined as a *conjunction* on elementary atoms in the KB. Atoms are of 2 types: *Concept* and *Role* atoms, of the type $C(?x)$ and $R(?x, ?y)$. These are, simplistically, antonyms of *subjects/objects* and *predicate* as in relational data format (RDF) terminology. Understandably variables in *Concept* and *Role* atoms are 1 and 2 respectively.

This work is based on the research done in [1] with the additional component of distributed fitness function calculation for lower time complexity. The main algorithm is an evolutionary technique designed to work on SWRL rules by enforcing a *Language Bias* and redefining *Crossover* and *Mutation* operators for association rules in ontological KBs. The main function is composed of the following functions:

1) **CreateNewPattern:** New random patterns are generated by selecting an atom for the head from a frequent atom list and then adding them to the body until a rule length, defined at the start of the function, is reached.
2) **Recombination:** Target lengths of patterns defined, followed by adding atoms (from pool of input atoms) until the lengths are satisfied. *Language bias* is enforced by maintaining transitive dependency during recombination.
3) **Mutate:** Applied with small probability. If head coverage of a pattern is higher than threshold, it is specialized. Otherwise it is generalized (last atom removed)
4) **Specialize:** Essentially adds a frequent concept or role atom to the pattern based on a random number generated in the range [0,1). *Language Bias* is enforced by controlling the variable bindings of the newly created atoms, based on predefined criteria.

IV. Implementation

The project was programmed in *Scala* while the distributed components were implemented in *Apache Spark*. *SANSA RDF* was used for parsing the N-Triple files used in the work. However, the existing data format for triples proved ill-suited for SWRL Rules. This resulted in us defining separate classes for `Atom` and `Rule` objects. A composite data structure was defined for SWRL compliant triples using `scala.collections` and `Atom` data structures by

manually parsing the RDD generated by the *SANSA RDF* frameworks and used to defined a separate RDD.

Primarily, the implementation went according to the algorithmic descriptions in [1], [2]. However, a major challenge was the enforcement of a *Language Bias* for the atoms in the SWRL rules. This was done by defining 3 `ListBuffer` objects in each `Rule` object populated by a simple integer values for the following:

- **`Available`**: This was initially populated by elements in the range of $0 - 20$. These represented bindings for new `Atom` objects with *non-bound* variables to be added into the rule. Values are removed every time when new objects use one of them when added into the `Rule` object.
- **`NonAvailable`**: This was defined to represent bindings that are non-available for free variables. This list is initialized empty and populated any time an *Available* variable is used.
- **`Priority`**: These represent variables that are a priority to be used – typically these are dangling `Atom` objects with unbound variables which risk the containing `Rule` object losing transitive dependence among all the atoms.

The `Recombine(..)` and `Specialize(..)` functions used a combination of the above to maintain *Language Bias* in the implementation.

V. Results & Evaluation

VI. Conclusion

References

[1] dAmato, C., Tettamanzi, A. G., & Minh, T. D. *Evolutionary discovery of multi-relational association rules from ontological knowledge bases*. In European Knowledge Acquisition Workshop (pp. 113-128). Springer, Cham, (2016, November).

[2] d'Amato, Claudia, et al. *Ontology enrichment by discovering multi-relational association rules from ontological knowledge bases* Proceedings of the 31st Annual ACM Symposium on Applied Computing. ACM, 2016.