VMI-PSL: Visual Model Inspector for Probabilistic Soft Logic

AARON RODDEN∗, University of California, Santa Cruz, USA
TARUN SALH∗, University of California, Santa Cruz, USA
ERIQ AUGUSTINE, University of California, Santa Cruz, USA
LISE GETOOR, University of California, Santa Cruz, USA

Hybrid recommender systems achieve state-of-the-art performance by integrating several different information sources along with multiple recommendation approaches. Probabilistic Soft Logic (PSL) has been shown to be an accessible and effective means of creating extensible hybrid recommenders [11]. PSL allows users to easily create intuitive models that incorporate background information and capture complex interactions. However these complex interactions can sometimes make PSL models difficult to inspect, debug, and understand. In this paper, we present a generic visual model inspector for PSL, and show how our inspector can be used on a hybrid recommender system to: debug errors in the model, analyze the performance of individual components of the model, and explain recommendations made by the model.

ACM Reference Format:

1 INTRODUCTION

Hybrid recommender systems achieve state-of-the-art performance by integrating several different information sources with multiple recommendation approaches [1, 5, 6, 8, 11, 14, 15]. Statistical Relational Learning (SRL), which combines weighted logical rules with probabilistic inference to model complex data dependencies and make structured predictions, has shown to be an accessible and effective means of creating hybrid recommenders [7, 9, 11, 13, 20, 22, 23]. Probabilistic Soft Logic (PSL) [3] is one such SRL framework that has been used in several hybrid recommenders [9, 11, 13, 22]. One of the appeals of PSL models is that they are intuitive to create and make it easy to include background knowledge and complex data interactions. However, these complex interactions can sometimes make PSL models difficult to inspect, debug, and understand. While there has been significant work in explaining hybrid recommenders [2, 4, 5, 10, 12, 16–19, 21, 24], there has been far less work in debugging them. While the two are closely related, this paper focuses on debugging.

In this paper, we present our inspection tool for PSL: Visual Model Inspector for Probabilistic Soft Logic (VMI-PSL), and demonstrate its use on hybrid recommender systems [11]. VMI-PSL provides an easy-to-use web-based inspector for PSL models that requires no additional setup, uses data tables and aggregate statistics to display information about a model in a scalable way, and uses contextual information to display only relevant portions of the model.

∗Both authors contributed equally to this research.
To understand how a FSL model can be inspected, we must first discuss the basic structure of a FSL model. A FSL model is constructed by instantiating weighted logical rules with data, these instantiations are referred to as ground rules. The components in a ground rule are referred to as ground atoms. Ground atoms assume a continuous truth value in the range [0, 1], and ground rules take on a degree of satisfaction in the range [0, 1].

Our key contributions are as follows: 1) we augment the PSL framework to output the necessary information for model inspection, 2) we create a new visual inspection tool for PSL models, VMI-PSL, 3) we provide sample workflows to show how VMI-PSL can be used with a recommender system model. To the best of our knowledge, VMI-PSL is the first generic visual model inspector for any SRL framework.

2 SYSTEM OVERVIEW

VMI-PSL is designed to mimic the workflow of an experienced PSL user as they manually inspect a PSL model. Information is organized into three different context layers, each containing modules that automatically activate when a user selects an item. These layers are the model layer, the ground atom layer, and the ground rule layer. The information displayed starts general at the model layer, but then becomes more detailed as the user drills down into more specific context layers. Figure 1 shows a module from each of the three layers on a hybrid recommender system model.[11]

![Module Overview](image)

Fig. 1. The three different context layers in VMI-PSL, along with a module from each layer. Here, the ground atom \text{RATING}(5, '1769') is the context for the ground atom layer and the ground rule \text{BPMF}\_\text{RATING}(5, '1769') \rightarrow \text{RATING}(8, '1769') is the context for the ground rule layer.

The model layer contains information about the model as a whole, and contains five modules: 1. Rule Overview - a table of all the rules used in the model along with each rule's weight, count, total dissatisfaction, and mean dissatisfaction. 2. Violated Constraints - a table of all violated hard constraints. 3. Rule Compatibility - a bar graph containing all the rules, the display can be toggled between total satisfaction, mean satisfaction, total dissatisfaction, and mean dissatisfaction. 4. Ground Rule Count - a bar graph showing the number of ground rules instantiated from each rule. 5. Truth Table - a table that shows every ground truth value and its corresponding prediction.

When a ground atom is selected, the ground atom layer is activated with two modules: 1. Associated Rule Compatibility - a bar graph similar to the model layer's Rule Compatibility module, however only ground rules containing the selected ground atom are aggregated. 2. Associated Ground Rules - a table that shows every ground rule that contains the selected ground atom along with the dissatisfaction of the ground rule.

Lastly, the ground rule layer contains a single module: 1. Rule Satisfaction - a table that shows the total satisfaction of the ground rule along with the truth value of each ground atom involved in that ground rule.

1 Code available at https://github.com/lnq/pml-vml
3 EXAMPLE WORKFLOWS

VMI-PSL is supported in all PSL interfaces, and no additional setup is required. To use VMI-PSL, a user must first run their existing PSL model with the additional --visualization option. PSL generates a JSON file that contains all the necessary information for model inspection. The user must then go to https://lists.githack.io/psl-vmi and select the generated JSON file when prompted. Finally, VMI-PSL rebuilds the model from the JSON file and displays all the relevant information.

![Diagram](image_url)

**Fig. 2.** Three different workflows showcasing different use cases of VMI-PSL: debugging an erroneous model, analyzing the performance of components of a working model, and explaining specific predictions.

Figure 2 shows three different workflows of VMI-PSL on the HyPER hybrid recommender model[11]. In the Debug workflow, VMI-PSL is being used to answer the question: Why is my model giving incorrect results? A common cause of erroneous models is a rule unintentionally generating zero ground rules. This can be caused by corrupted data files or incorrectly written rules. The workflow for this use case is simple, the Rule Overview, Rule Compatibility, and Ground Rule Count modules will all show zeroes in their respective metrics for that rule. The Analysis workflow answers the question: What components of my model can be improved? To answer this question, the first step is to select an atom with an incorrect prediction from the Truth Table module. Then, the Ground Atom layer is activated and the Associated Rule Compatibility can be used to find the rules that most contribute to the incorrect prediction. Finally, the Explainability workflow answers the question: What causes a specific prediction? The first step is looking up the specific prediction in the Truth Table module. Once the prediction is selected, the Associated Ground Rules module can be sorted in ascending order by dissatisfaction to find the specific ground rules that contribute most to this prediction. Selecting each ground rule will open the Rule Satisfaction module where the individual contribution of each ground atom in the selected rule can be observed. These explanations are found in the same way as [13].

4 CONCLUSION AND FUTURE WORK

In this paper, we introduce a tool for inspecting PSL models, and show how it can be applied to hybrid recommenders. VMI-PSL is generic, easy-to-use, and requires no additional setup. In addition, we have shown workflows of this tool being used in a hybrid recommendation setting to debug an erroneous model, analyze the performance of individual components in a model, and explain the predictions made by a model. A promising future direction is to attach VMI-PSL to an online inference engine for PSL, making it possible to see the effects of changing rules or atoms without needing to re-run PSL.

ACKNOWLEDGMENTS

This work was partially supported by the National Science Foundation grants CCF-1740850 and IIS-1703331, AFRL and the Defense Advanced Research Projects Agency.
REFERENCES


[18] Denis Parra, Peter Brusilovsky, and Christoph Trattner. 2014. See what you want to see: visual user-driven approach for hybrid recommendation. In Intelligent User Interfaces (IUI). ACM, Haifa, Israel.


