Reviews Thesis-Paper

Samuel Kolb

October 8, 2015

1 Review 1

The author introduces a framework for learning both constraints and optimization criteria. The paper is a cut and paste of the author's thesis and it's hard to understand. In this respect, the overall paper organization needs improvements and the text needs an accurate rewriting to allow a better understanding of the content. Moreover, it is not really clear the difference with the state-of-the-art solutions given the lack of comparison.

2 Review 2

To me, the paper presents the following weak points, for which it cannot be accepted:
- It is stated that the paper was born from a thesis and I believe few has been done to convert it into a proper research paper. The Introduction doesn't talk about state of the art, possible applications fields of the presented approach, etc.
- section 2.1 explains FOL in a top-down manner, while usually one starts from atoms, terms until clauses. Moreover no reference is included - section 2.2 should be moved in the introduction, rather than in the background, since it gives the reader an idea of the existing methods - in section 2.3 it is stated that a different implementation has been used for the clausal discovery algorithm, but it is described nowhere - section 2.4 doesn't explain what an IDP theory is - section 2.5 doesn't cite any work about MAXSAT
- Section 3 and 4 titles should be made more specific and content-related - in sect. 3 it is not shown the format (interpretations?clauses?) of the examples. I think a complete example should represent the KB, the learned constraints, the weights, ecc.. - in sect. 3 the order over the examples is unclear: which is the criterion that allows one to say that an example comes before another?
- Example 3: why are variables lowercase and constants uppercase? Don't you use FOL syntax? - In example 4 how is the function $(1 \cdot c1) + (0 \cdot c2) + (-2 \cdot c3)$ obtained from Table 1? - No reference for SVMrank is given in sect. 4.3
- section 5.1: the examples should be illustrated in much more detail (what are the learnt constraints for the different domains?) - section 5.1 - you name a paragraph "accuracy" but no accuracy values are reported on the experiments. - in Table 2 what do you mean by Time(baseline)? - "compared to humans" paragraph: the IDP system URL should be given - section 5.2: what do you mean by "All possible pairs of examples are generated"?
How is this work positioned w.r.t. the state of the art - experimental results do not report such a comparison.

3 Review 3

This paper proposes an ILP-based approach to automatically learn hard and soft constraints from examples for use in a Constraint Satisfaction based problem solver. The paper is more of an informal introduction to the proposed approach than a technical presentation thereof. It lacks

justification of the choices, formal description of the methodology, detailed description of the experimental settings, and insight of the experimental results.

There are some formal issues. E.g.: - in the basic definitions, "an atom is either true or false or a predicate" is incorrect. An atom is by definition a predicate applied to its arguments (if any), that can be true or false depending on an interpretation. true is a predicate without arguments that is always true; false is a predicate without arguments that is always false. - in Sec. 3, concerning soft constraints, what do the equalities $e_x = e_y$ and $e_a = e_b$ in the example of ranking mean? - in Sec. 3, concerning soft constraints, in the summation formula, what is 'weight'? it has no index, so it seems that it is independent of the clauses, so it should be moved to the left of the sum operator. The background information is quite informal, and does not discuss related work that is in competition with the proposed approach (it just discusses related work the proposed approach builds upon). Also, no explanation is given as to why exactly those systems are chosen to build upon. There are also other systems in the literature that might be used, why the proposed ones are better?

Sec. 4.3: where do the clauses used as constraints come from? I suppose they are not those learned in Sec. 4.2, because they were the hard constraints. Otherwise, if those were the soft constraints, where is the learning of hard constraints? Or, if the clauses learned according to Sec. 4.2 can be either hard or soft constraints, how and where is the distinction made?