

# SATzilla Results from the Data Analysis Track of the 2011 SAT Competition

*Lin Xu, Frank Hutter, Holger H. Hoos, Kevin Leyton-Brown*  
Department of Computer Science  
University of British Columbia,  
Vancouver, BC, Canada

*The real state of the art in SAT solving is not the winner of the latest competition, but the solver that can easily and automatically be built using competition entrants. Thus, this year we entered SATzilla as an evaluation procedure rather than as a solver. We constructed portfolios including all non-portfolio, non-parallel solvers from Phase 2 of the competition as component solvers, and compared them to the winners of each category (including other portfolio-based solvers). We also compared to the “virtual best solver” (VBS): an oracle that magically chooses the best solver for each instance and considers all competition entrants (including parallel solvers, solvers we discarded as being unhelpful to SATzilla, etc). The VBS does not represent the current state of the art, since it cannot be run on new instances; nevertheless, it serves as an upper bound on the performance that any portfolio-based selector could achieve.*

**Results** *The table below shows the results of our analysis. We used the latest version of SATzilla (SATzilla 2011), trained on all non-portfolio, non-parallel solvers from Phase 2 of the competition and evaluated using 10-fold cross-validation on the complete instance set from the competition. As can be seen from these results, SATzilla 2011 performed substantially better than the gold medalists (including other portfolio-based solvers), reducing the gap between these and the VBS by 25.4% on Application, by 47.3% on Crafted and 78.9% on Random.*

<b>Category</b>	<b>Gold medal winner</b>	<b>SATzilla 2011</b>	<b>Virtual best solver</b>
<i>Application</i>	71.7% (1856 CPU s) - <i>Glucose 2</i>	75.0% (1707 CPU s)	84.7% (1104 CPU s)
<i>Crafted</i>	54.3% (2602 CPU s) - <i>3S</i>	64.7% (2155 CPU s)	76.3% (1542 CPU s)
<i>Random</i>	68.0% (1836 CPU s) - <i>3S</i>	79.2% (1313 CPU s)	82.2% (1074 CPU s)

**Acknowledgement** We thank *Daniel Le Berre* and *Olivier Roussel* for providing us access to the competition data and running our feature computation code.