doi:10.20944/preprints202302.0317.v1

SBOannotator: a Python Tool for the Automated Assignment of Systems Biology Ontology Terms

Nantia Leonidou[1,2,3,*], **Elisabeth Fritze**[2], **Alina Renz**[1,2,3], **and Andreas Dräger**[1,2,3,4]

[1]Computational Systems Biology of Infections and Antimicrobial-Resistant Pathogens, Institute for Bioinformatics and Medical Informatics (IBMI), University of Tübingen, 72076 Tübingen, Germany

[2]Department of Computer Science, University of Tübingen, 72076 Tübingen, Germany

[3]Cluster of Excellence 'Controlling Microbes to Fight Infections', University of Tübingen, Germany

[4]German Center for Infection Research (DZIF), partner site Tübingen, Germany

[*]Correspondence: nantia.leonidou@uni-tuebingen.de

ABSTRACT

The number and size of computational models in biology have drastically increased over the past years and continue to grow. Modeled networks are becoming more complex, and reconstructing them from the beginning in an exchangeable and reproducible manner is challenging. Using precisely defined ontologies enables the encoding of field-specific knowledge and the association of disparate data types. In computational modeling, the medium for representing domain knowledge is the set of orthogonal structured controlled vocabularies named Systems Biology Ontology (SBO). The SBO terms enable modelers to explicitly define and unambiguously describe model entities, including their roles and characteristics. Here, we present the first standalone tool that automatically assigns SBO terms to multiple entities of a given SBML model, named the SBOannotator. The main focus lies on the reactions, as the correct assignment of precise SBO annotations requires their extensive classification. Our implementation does not consider only top-level terms but examines the functionality of the underlying enzymes to allocate precise and highly specific ontology terms to biochemical reactions. Transport reactions are examined separately and are classified based on the mechanism of molecule transport. Pseudo-reactions that serve modeling purposes are given reasonable terms to distinguish between biomass production and the import or export of metabolites. Finally, other model entities, such as metabolites and genes, are annotated with appropriate terms. Including SBO annotations in the models will enhance the reproducibility, usability, and analysis of biochemical networks.

Availability: The open-source project SBOannotator is freely available under the terms of LGPL version 3.0 from `https://github.com/draeger-lab/SBOannotator/`.

Keywords: SBOannotator; ontologies; SBO terms; Python; software; automated assignment; computational modeling; systems biology

Introduction

In bioinformatics, ontologies are generally used to share common knowledge and its application across communities[1]. While concepts in biology are adequately covered by appropriate ontologies, such as the Gene Ontology (GO)[2], model-related semantics are encoded by standardized SBO terms[3]. The SBO is a set of orthogonal controlled vocabulary terms used to explicitly and unambiguously describe the semantics of model instances. They are divided into eight orthogonal vocabularies and can be employed to annotate a model and describe various entities. For instance, they may represent the type or role of a single component in a model streamlining the understanding and meaning of this entity. The more specific the SBO term is, the more precise the description. As of January 2023, they consist of 694 terms, with 24 newly added in the last three years. Generally, such terms ensure model reproducibility and exchangeability as they record and categorize the semantics of model components. From the release of SBML Level 2 Version 2 (Revision 1) in the fall of 2006 to the current edition (SBML Level 3 Version 2, Revision 2[5]), the SBML format has supported annotating its components using SBO terms to unambiguously mark their semantics and extend their scope. At this point, adding general, top-level SBO terms to a model can be done automatically. However, adding precise descriptions for biochemical reactions, e.g., glycosylation or hydrolysis, remains a laborious and complicated step. After precise categorization, all terms must be determined and added individually to each occurrence. A higher-level SBO term specificity in reactions can enable new model analysis methods. For instance, similar to the gene set enrichment analysis, by counting the occurrence of SBO terms, one could easily deduce the types of over-catalyzed reactions, either for complete models or selected pathways.

Moreover, as already mentioned, the SBO terms are also considered a tool for model reproducibility. Hence, their automated assignment is of great importance. Here, we implemented an expert knowledge-driven classification scheme implemented in Python called SBOannotator, which can be easily used to assign SBO terms in a given SBML model[6] automatically.

Results

The SBOannotator workflow comprises six main steps (Figure 1). At first, all reactions found within the model are

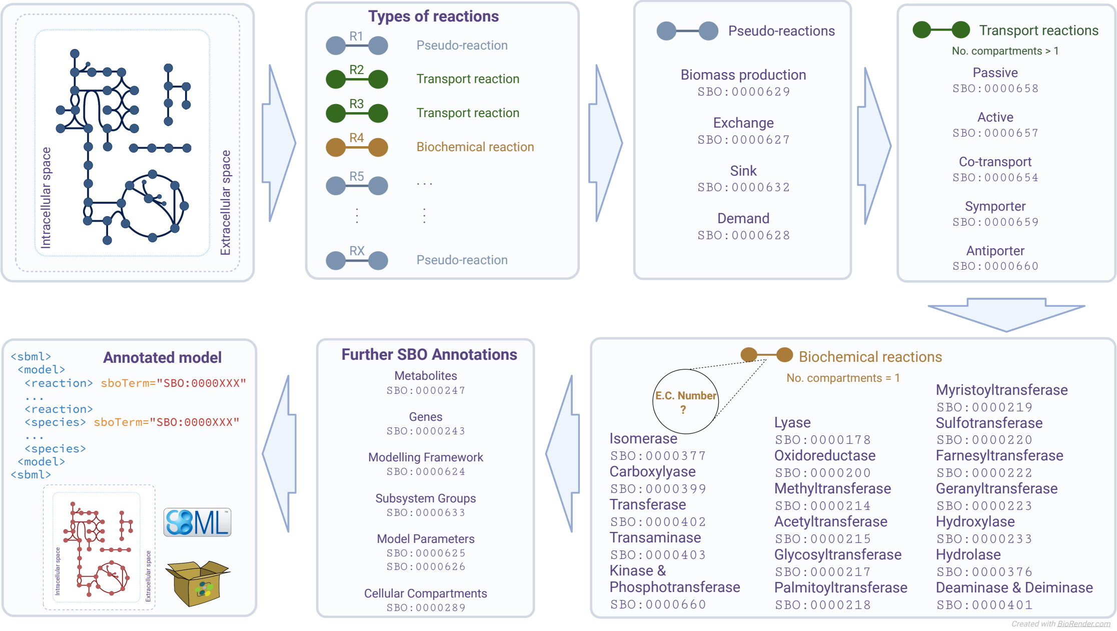

Figure 1 | Overview of SBOannotator's pipeline. The software enables the automated assignment of SBO terms to multiple model entities in a given SBML file. The main advantage is the detailed categorization of biochemical reactions and the allocation of specialized terms that precisely capture related and exchangeable information.

labeled as either (a) *transporters* that move molecules across different compartments, (b) *simple biochemical reactions* that only take place in the cytosol, or (c) *pseudo-reactions* that import or export metabolites and serve modeling purposes. Then, the pseudo-reactions are detected and analyzed. Pseudo-reactions in systems biology modeling do not correspond to any actual physical process and should not be confused with the pseudo-first-order reactions from the field of chemical kinetics. They are subdivided into demand, exchange, and sink reactions. The biomass objective function also belongs to this class. Exchange (`SBO:0000627`) and sink (`SBO:0000632`) reactions are reversible reactions that add or remove metabolites, with the latter one to be specific for intracellular compounds.

On the other hand, demand reactions (`SBO:0000628`) operate in only one direction and consume intracellular metabolites. The biomass objective function (`SBO:0000629`) is usually the optimization target reaction in modeling bacterial metabolism and simulates the organism's growth.

SBOannotator processes further by examining the transport reactions and assigning appropriate SBO terms. The classification mechanism in this step is comparably advanced since several types of transporters exist. The decision relies on the main characteristics of the different classes, such as the presence of one (passive transport) or more reaction participants, and the consumption of adenosine triphosphate (ATP) or phosphoenolpyruvate (PEP) (active transport). If reversible reactions are labeled as active transporters, a warning is printed to the user indicating that these reactions are thermodynamically infeasible. Subsequently, the total number of cellular compartments is derived to enable the distinction between symporters/antiporters and co-transporters. Reactions with metabolites from more than two compartments are characterized as co-transporters (`SBO:0000654`), while the rest is divided and either labeled as *symporter* (`SBO:0000659`, reactants, and products are from the same compartment) or *antiporter* (`SBO:0000660`, reactants, and products are from different compartments).

The remaining biochemical reactions are processed in the next step to enable more detailed labeling. For this purpose, the SBOannotator employs an Structured Query Language (SQL) database that contains mappings between Enzyme Commission (EC) numbers and the respective SBO terms. As the model's size increases, using an already-defined database accelerates the computational time needed for their annotation. To create this database, we browsed all children nodes of the `biological reaction` node in the SBO's directed acyclic graph. Our mappings could be divided into three main categories: (a) one-to-one mapping; one SBO term represents EC numbers from a single sub-subclass (e.g., transamination), (b) one-to-few mapping; one SBO term maps only a subset of EC numbers belonging in a single sub-sub-class (e.g., myristoylation), and (c) and one-to-many mapping; one SBO term covers a large subset of EC numbers within one sub-subclass (e.g., acetylation). The supplementary table S1 lists all mappings in detail. The SBOannotator assigns the general SBO term of a biochemical reaction (`SBO:0000176`) if the reaction has multiple EC numbers assigned and all are from different classes resulting in an unambiguous description. Otherwise, the SBO terms are accredited based on the respective enzyme's main class (e.g., oxidoreductases; `SBO:0000200`, transferases; `SBO:0000402`, hydrolases; `SBO:0000376`). It is important to note that a proper term that describes the ligases (EC class 6) is currently missing from the SBO graph. This would be necessary to describe, for instance, reactions involving the formation of deoxyribonucleic acid (DNA), ribonucleic acid (RNA), and protein fragments. We included an appropriate SBO term for ligases that describes the general modification of covalent bonds (`SBO:0000182`). The metabolic reactions that do not fall in any of the already mentioned cases and do not have any EC number assigned are given the general SBO term of a biochemical reaction (`SBO:0000176`). The SBOannotator is designed to handle models with or without EC numbers assigned. However, they should either be annotated with Biochemical, Genetical, and Genomical (BiGG) [7] identifiers or include any intonations. If the input model provides no EC numbers, an integrated Application Programming transfer Interface (API) call requests the necessary information from the BiGG database and adds all missing annotations into the model. Depending on the model's size, this step may increase the computational time. Hence, we recommend the prior use of an annotation tool, such as ModelPolisher[8]. We have tested the performance of SBOannotator in assigning descriptive and more precise terms to biochemical reactions using 108 metabolic models from the BiGG database. All downloaded models contained only five types of SBO annotation representing only top-level terms. Nevertheless, all biochemical reactions had a single generic term without specifying the exact type of reaction. However, our tool annotated the models with 31 different terms considering the underlying enzymatic properties (see Table S2 and Table S3). The biochemical reactions made up the largest reaction group before and after the SBOannotator. However, their coverage was reduced from 57.9% to 18.9%, meaning a large percentage of the initial reactions got a more specific term (see Fig. S1 and Fig. S2). The second most common term in the downloaded models described translocations. Our annotated models contain more specific terms based on the respective transport mechanisms. Across all models, decarbonylations (`SBO:0000400`) occurred most rarely and only ten times.

Finally, SBOannotator assigns SBO terms to the remaining model entities. These include metabolites, genes, cellular compartments, and defined parameters. If subsystem groups are declared, the SBOannotator allocates the term *subsystem* (`SBO:0000633`), while the respective modeling framework is also assigned an appropriate term. The final annotated SBML model is stored in the current working directory with the name tag `_SBOannotated` in an Extensible Markup Language (XML) format.

Conclusion

Overall, the SBOannotator is a freely available and user-friendly Python tool. It can be easily employed to rapidly annotate systems biology metabolic networks in SBML format with appropriate SBO terms, with particular emphasis on allocating precise and descriptive terms to all chemical reactions. So far, SBOannotator is a standalone application. Its integration into existing software, such as ModelPolisher[8], could be worthwhile. Lastly, the tool can be easily extended to additional terms specific to different model types, including kinetic and dynamic models.

Data availability

The SBOannotator tool, all related data, and a demo script to run the code are available in a git repository at `https://github.com/draeger-lab/SBOannotator/`. Along with this article, a supplementary table in Comma-separated Values (CSV) format is available.

Acknowledgments

N.L. and A.R. are supported by the *Deutsche Forschungsgemeinschaft* (DFG, German Research Foundation) under Germany's Excellence Strategy – EXC 2124 – 390838134 and supported by the Cluster of Excellence 'Controlling Microbes to Fight Infections' (CMFI). A.D. is supported by the German Center for Infection Research (DZIF, doi:10.13039/100009139) within the Federal Ministry of Education and Research (BMBF, German Centers for Health Research of the Federal Ministry of Education and Research (BMBF)), grant № 8020708703. The authors acknowledge the support by the Open Access Publishing Fund of the University of Tübingen (`https://uni-tuebingen.de/en/216529`). One figure was created with BioRender.com.

Competing interests:

The authors declare no conflict of interest.

References

1 R. Stevens, C. A. Goble, and S. Bechhofer. "Ontology-based knowledge representation for bioinformatics". In: *Briefings in bioinformatics* 1.4 (2000), pp. 398–414. DOI: `10.1093/bib/1.4.398`.

2 M. Ashburner, C. A. Ball, J. A. Blake, D. Botstein, H. Butler, J. M. Cherry, A. P. Davis, K. Dolinski, S. S. Dwight, J. T. Eppig, et al. "Gene ontology: tool for the unification of biology". In: *Nature genetics* 25.1 (2000), pp. 25–29. DOI: `10.1038/75556`.

3 M. Courtot, N. Juty, C. Knüpfer, D. Waltemath, A. Zhukova, A. Dräger, M. Dumontier, A. Finney, M. Golebiewski, J. Hastings, et al. "Controlled vocabularies and semantics in systems biology". In: *Molecular systems biology* 7.1 (2011), p. 543. DOI: `Controlledvocabulariesandsemanticsinsystemsbiology`.

4 A. Finney, N. Le Novère, and M. Hucka. *Systems Biology Markup Language (SBML) Level 2: Structures and Facilities for Model Definitions*. Available from COMBINE at `https://identifiers.org/combine.specifications/sbml.level-2.version-2`. 2006.

5 M. Hucka, F. T. Bergmann, C. Chaouiya, A. Dräger, S. Hoops, S. M. Keating, M. König, N. Le Novère, C. J. Myers, B. G. Olivier, et al. "The systems biology markup language (SBML): language specification for level 3 version 2 core release 2". In: *Journal of integrative bioinformatics* 16.2 (2019). DOI: `10.1515/jib-2019-0021`.

6 S. M. Keating, D. Waltemath, M. König, F. Zhang, A. Dräger, C. Chaouiya, F. T. Bergmann, A. Finney, C. S. Gillespie, T. Helikar, et al. "SBML Level 3: an extensible format for the exchange and reuse of biological models". In: *Molecular Systems Biology* 16.8 (2020), e9110. DOI: 10.15252/msb.20199110.

7 C. J. Norsigian, N. Pusarla, J. L. McConn, J. T. Yurkovich, A. Dräger, B. O. Palsson, and Z. King. "BiGG Models 2020: multi-strain genome-scale models and expansion across the phylogenetic tree". In: *Nucleic Acids Research* 48.D1 (Nov. 2019). gkz1054. ISSN: 0305-1048. DOI: 10.1093/nar/gkz1054.

8 M. Römer, J. Eichner, A. Dräger, C. Wrzodek, F. Wrzodek, and A. Zell. "ZBIT bioinformatics toolbox: a web-platform for systems biology and expression data analysis". In: *PloS one* 11.2 (2016), e0149263. DOI: 10.1371/journal.pone.0149263.