THE UNIVERSITY
of EDINBURGH

This thesis has been submitted in fulfilment of the requirements for a postgraduate degree (e.g. PhD, MPhil, DClinPsychol) at the University of Edinburgh. Please note the following terms and conditions of use:

- This work is protected by copyright and other intellectual property rights, which are retained by the thesis author, unless otherwise stated.
- A copy can be downloaded for personal non-commercial research or study, without prior permission or charge.
- This thesis cannot be reproduced or quoted extensively from without first obtaining permission in writing from the author.
- The content must not be changed in any way or sold commercially in any format or medium without the formal permission of the author.
- When referring to this work, full bibliographic details including the author, title, awarding institution and date of the thesis must be given.

# On combining collaborative and automated curation for enzyme function prediction

*Luna De Ferrari*

Doctor of Philosophy
Centre for Intelligent Systems and their Applications
School of Informatics
University of Edinburgh
2012

# Abstract

Data generation has vastly exceeded manual annotation in several areas of astronomy, biology, economy, geology, medicine and physics. At the same time, a public community of experts and hobbyists has developed around some of these disciplines thanks to open, editable web resources such as wikis and public annotation challenges. In this thesis I investigate under which conditions a combination of collaborative and automated curation could complete annotation tasks unattainable by human curators alone.

My exemplar curation process is taken from the molecular biology domain: the association all existing enzymes (proteins catalysing a chemical reaction) with their function. Assigning enzymatic function to the proteins in a genome is the first essential problem of metabolic reconstruction, important for biology, medicine, industrial production and environmental studies. In the protein database UniProt, only 3% of the records are currently manually curated and only 60% of the 17 million recorded proteins have some functional annotation, including enzymatic annotation. The proteins in UniProt represent only about 380,000 animal species (2,000 of which have completely sequenced genomes) out of the estimated millions of species existing on earth. The enzyme annotation task already applies to millions of entries and this number is bound to increase rapidly as sequencing efforts intensify.

To guide my analysis I first develop a basic model of collaborative curation and evaluate it against molecular biology knowledge bases. The analysis highlights a surprising similarity between open and closed annotation environments on metrics usually connected with "democracy" of content.

I then develop and evaluate a method to enhance enzyme function annotation using machine learning which demonstrates very high accuracy, recall and precision and the capacity to scale to millions of enzyme instances. This method needs only a protein sequence as input and is thus widely applicable to genomic and metagenomic analysis.

The last phase of the work uses active and guided learning to bring together collaborative and automatic curation. In active learning a machine learning algorithm suggests to the human curators which entry should be annotated next. This strategy has the potential to coordinate and reduce the amount of manual curation while improving classification performance and reducing the number of training instances needed. This work demonstrates the benefits of combining classic machine learning and guided learning to improve the quantity and quality of enzymatic knowledge and to bring us closer to the goal of annotating all existing enzymes.

# Acknowledgements

I would like to thank my supervisors: Stuart Aitken, Jano van Hemert and Igor Goryanin and the funding bodies that supported my research. I was funded by BBSRC Grant BB/F529038/1 of the Centre for Systems Biology at Edinburgh (CSBE) and the University of Newcastle and by a University of Edinburgh staff scholarship. Stuart Aitken was funded by BBSRC grant BB/F015976/1 and by a Wellcome Trust Value In People award. Jano van Hemert was funded by various BBSRC and EPSRC grants.

Stuart Aitken and Igor Goryanin were also funded by CSBE, a centre funded by BBSRC and EPSRC, ref. BB/D019621/1. And thanks for their support to my colleagues Anatoly Sorokin, Andrew Millar, Fiona Clark, Hongwu Ma, Liz Elliot, Ross Armstrong, Richard Adams, Shakir Ali, Stuart Moodie and Teresa Ironside (School of Informatics and Centre for Systems Biology, the University of Edinburgh).

I would also like to thank the entire Ondex team and Chris Rawlings, Catherine Canevet and Anil Wipat in particular. Many thanks also to Grigorios Tsoumakas (Mulan), the members of the Mulan mailing list, Daniel Renfro (EcoliWiki), Joanna Amberger (OMIM), David Croft (Reactome) and Alexander Pico (WikiPathways) for their collaboration. A special thank to Paolo Besana, Michelle Galea, Michael Rovatsos, Guido Sanguinetti (University of Edinburgh) and John Mitchell (University of St Andrews) for their help in clarifying this manuscript, and the ideas therein.

A big thank to all my PhDUnsocials friends: Annette Lehonard, Chris Fensch, Christophe Dubach, David Spark, Gaya Nadarajan, June Tee, Kanako Sobajima, Tom Goddy, Vera Demberg and also Adam Barker, Adela Grando, Andreia Nobre, Fiona McNeill, Ines Asceric-Todd, Jarred McGinnis, Nana Ishino, Rob Kitchen, Rosa Filgueira, Sarah McGinnis, Thomas French and the New Texas state (also known as Informatics Forum room 1.05). A big hug to my supportive family: my partner Paolo Besana, my son Daniel Besana, my mother Adriana Sangiovanni, Giorgio Sabadei, Maristella Bellosta, Alberto Besana, Laura Besana-Boledi and Luigi Boledi.

It takes a village to raise a PhD student.

# Declaration

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

(*Luna De Ferrari*)

## Publications

Parts of this thesis have appeared in the following publications:

- De Ferrari, L.; Aitken, S.; van Hemert, J. and Goryanin, I. *WikiSim: simulating knowledge collection and curation in structured wikis* WikiSym '08: Proceedings of the 4th International Symposium on Wikis, ACM, 2008, 1-2
- De Ferrari, L.; Aitken, S.; van Hemert, J. and Goryanin, I. Edmonds, B. and Gilbert, N. (Eds.) *A model of social collaboration in Molecular Biology knowledge bases* Proceedings of the 6th Conference of the European Social Simulation Association (ESSA'09), European Social Simulation Association, 2009, 47
- De Ferrari, L.; Aitken, S.; van Hemert, J. and Goryanin, I. *Multi-label prediction of enzyme classes using InterPro signatures* Machine Learning for Systems Biology Workshop (International Conference of Systems Biology), 2010
- De Ferrari, L.; Aitken, S.; van Hemert, J. and Goryanin, I. *EnzML: Multi-label prediction of enzyme classes using InterPro signatures* BMC Bioinformatics, 13:61, 2012
- De Ferrari, L.; Aitken, S.; Mitchell J. *Active and guided learning for the prediction of enzyme function* 11th European Conference on Computational Biology, 2012

To my father, his grandson and his father.

# Table of Contents

# List of Figures

# List of Tables

# Chapter 1

# Introduction

## 1.1 Motivation

In many scientific domains, the rate of data production is overwhelming not only traditional manual curation, but also computer-supported basic annotation. Open collaborative endeavours, such as Wikipedia, have shown promise and new wiki-style environments have now started to tackle more intensive annotation of scientific data. However, the numbers of entities in astronomy and biology are bound to challenge any worldwide community of human curators.

This is why, in addition to manual curation, automated annotation and prediction methods have been extensively used to enrich scientific databases. Unfortunately, automated methods are usually used in a one-off, batch mode, without leveraging the machine learning algorithm (and what it tells us about the data) for allocating curation priorities over time.

Beyond the checking of very simple spelling and typographical mistakes, automated methods are bound to introduce mistakes. The error rate of a machine learning method can be evaluated in various ways, and can even compare positively with the human error rate, but, ultimately, we do not know with certainty if an introduced annotation is erroneous until a manual check detects it. This risk of cascading down errors has to be stressed, and annotation should always be presented together with its justification. This work makes a case for a combination of machine learning and active learning to prioritise scarce manual curation towards the most informative checks.

The aim of this work is to explore techniques capable of completing the annotation of all existing enzymes, namely, of all proteins able to catalyse chemical reactions. This challenge has analogous in other domains: the advantages and challenges of using automation not only in classification tasks, but to guide the curation workflow is a little explored area of research with significant potential applications for real data.

This work explores what happens when manual curation is limited (or very limited) compared to the overall annotation task. What happens in the – often dreaded or avoided – case when automatic curation is the only provider of annotation for most data points? Can machine learning help not only in predicting and classifying, but also in prioritising curators efforts to improve overall annotation quality?

**The protein and enzyme function challenge** To examine a particular case in molecular biology, manual curation has not been able to provide a functional description for all the 17 million proteins in UniProt TrEMBL [Cons 11] (as of September 2011) found with high throughput sequencing, not to mention the hundreds of million of proteins that could be found in the estimated 3 to 10 million species not sequenced – or incompletely sequenced – yet. About a third of these proteins are enzymes. Hoping that the current horrific rate of species extinction will *not* solve the problem, the challenge will require radically new approaches.

Numerous wiki solutions have been proposed to bring the "Wikipedia" effect to biology [Wang 06]. Wikis have started to be used to share and collect knowledge in specific areas of biology, with promising wiki initiatives led by model organisms consortia. More structured wikis are evolving to cope with more structured data and these tools mediate between ease of editing and the need for machine readable formats. With suitable hardware and software infrastructure they can manage hundreds of million of records. But the specialist community of researchers, students and hobbyists of molecular biology cannot stretch indefinitely. Manual curation alone might never annotate all enzymes [Baum 07].

**Hypothesis** My main hypothesis is that knowledge quantity and quality can improve, under certain conditions, when automated and open collaborative manual curation are brought together over the same data collection and in the same curation process.

Very good methods for automated prediction of protein functions exist. Their code availability and usability varies widely though. My hypothesis is that automated methods of prediction could profit from being more directly embedded in the *data* workflow of an open, collaborative curation environment. For example, to retrain the machine learning algorithm on fresh, corrected data and, where the algorithm allows, to give more weight to manually curated data points.

In addition, this work examines the potentially positive effects that active learning could have on the *curation* workflow, in orienting and allocating the manual curation effort.

To these ends, a specific new software environment is not strictly needed. In practice, a curation process of the scale described depends on the existence of: 1. a curation environment and 2. a machine learning environment able to cope with the data volumes involved. The required software already exists to a large extent, from structured wikis to machine learning libraries able to cope with million of entries, and some are discussed and used in this work. In addition, the two environments will need to exchange basic data (the records and their annotation) and also some information usable to guide the curation process, such as a best ranking of the entries to be curated. Again, this is not a complex task for current software. As for the curation environment, only the possibility of viewing the data and editing its annotation is assumed. In practice, this work has been inspired by current wiki software and its affordances for structuring and exporting computer readable formats and for collaborative curation such as in-line editing, separate discussion space, versioning, user identification and user notifications. Hence this work does not aim to provide another tool of this kind.

## 1.2 Contributions

This work provides some elements that are still missing in order to meet the challenge of enzyme annotation: first, an initial model to simulate manual, collaborative curation, second, a very strong method for automatic prediction of enzyme function and, to bring the two together, an evaluation of the integration between manual and automated curation on real data, using active and guided learning.

In more detail, this thesis contributions are:

- A baseline model of manual collaborative curation, and its evaluation against four real molecular biology curation endeavours.
- Novel results regarding how "democratic" wikis compare to closed annotation scenarios.
- An accurate method to predict enzymatic function from sequence, and its evaluation.
- A quantitative analysis of the effects on prediction performance of 1. dataset size, 2. sequence redundancy, 3. grouping data by taxonomic domain and 4. data providers disagreements.
- A range of active and guided learning curation methods to optimise the order in which proteins are annotated, and their evaluation on real enzyme data.

A summary of the claims is found in Table 1.1 alongside their location in this thesis text and in the relevant journal or conference publications.

## 1.3 Thesis structure

This thesis contains a background, three main chapters and a conclusion chapter:

**Chapter 2 Background** describes the problem of protein function annotation and existing solutions for curation, from closed archives to wikis. This chapter also describes existing automated annotation methods and introduces active and guided learning.

| ***Component*** | ***Claim*** | ***Section*** | ***Reference*** |
|---|---|---|---|
| A model of collaborative curation | That the model can reproduce some aggregated measures (such as authorship) of knowledge collection dynamics | Section 3.4.2 on page 39 | [De F 08] |
| Model evaluation using manual curation data | That measures of "democracy" (distribution of edits per author) are unexpectedly close in wikis and non-wikis | Section 3.4.2 on page 39 | [De F 09] |
| A machine learning method to predict enzymatic function | That high accuracy and precision in predicting multiple enzymatic functions can be obtained using only the presence or absence of conserved sequences signatures.<br>That predictions do not particularly improve if data is partitioned by taxonomic domains | Section 4.2 on page 60 | [De F 10, De F 12b] |
| Evaluation of active and guided learning to prioritise manual curation | That actively selecting instances to be annotated can use less manual curation while maintaining high accuracy | Section 5.2 on page 88 | [De F 12a] |

Table 1.1: *Thesis claims. For each model or analysis component, the table gives the claims supported by the available data and results in this thesis. More details can be found in the corresponding thesis section and publication.*

**Chapter 3 A model of collaborative curation** describes an agent-based model of manual collaborative curation. A simulation strategy was adopted as it would be unfeasible to enact “curation experiments” in a sufficient number of real curation systems in a doctorate time-scale. The model is compared with the curation dynamics of four real life molecular biology knowledge bases and wikis.

**Chapter 4 Multi-label prediction of enzyme function** describes a new machine learning approach able to predict real enzyme function with very high accuracy and recall. The method is also used for the automatic side of the automatic+manual curation of protein sequences.

**Chapter 5 Active curation** measures the effect of adding an active or guided learning strategy to the overall curation process. This chapter demonstrates the advantages of guided learning in reducing the number of labelled instances needed to obtain high accuracy predictions and illustrates the method potential for curation parallelisation.

**Chapter 6 Conclusion** concludes the thesis discussing its implications for real life curation in biology and beyond, and identifies new research questions that could be addressed by future research.

## 1.4 Summary

This chapter has introduced the challenge of enzyme annotation and outlined the possibilities offered by the integration of passive and active machine learning into the manual curation process. The overall methodology could cope with curation tasks beyond what is currently possible even for a large scale collaborative effort. The following chapter introduces an initial model of manual curation and uses it to explore the dynamics of existing molecular biology curation efforts. This is an initial investigation in the direction of deciding which curation platform (open or closed) could be better suited to the introduction of an active curation workflow.

# Chapter 2

# Background

## 2.1 Big data, big curation

This work contributes to the area of *data annotation*. Here *data* is intended as a collection of real life observations. The observed instances are still material in some fields such as archaeology (artefacts) or biology (biological samples, cell lines, species specimens) but more and more they are represented and shared as digital data, for example strings of genetic sequences in molecular biology or images of astronomical objects. The known instances are often divided into classes following some attribute of interest. *Annotation* is defined here as the act of attributing one or more class values to an instance, hence declaring that the available class knowledge can be applied and extended to that instance. The term *curation* is used here in a more general sense, as a hypernym of annotation with stress on the whole process of completing and maintaining a collection of annotated exemplars.

This thesis explores a particular curation example: the annotation of enzymatic function, and the findings are relevant to other curation settings where:

1. Annotation requires significant background knowledge
2. Manual curation is scarce compared to the number of items to be annotated
3. Machine learning is able to provide curation (class prediction) of above random quality

Some curation endeavours do not fall under 1. and 3. because they require little background knowledge and yet machine learning cannot yet provide predictions of accuracy comparable to those provided by people. Astronomy has seen some curation challenges of this kind, requiring accurate, but non-specialist classifications that have been successfully opened to the general public. For example, the Moon Zoo website[1] aims at providing detailed crater counts for as much of the Moon's surface as possible. The site provides guidelines, for example, on how

[1] Moon Zoo `http://www.moonzoo.org/`

users can mark a moon crater as having boulders around it, a sign of a more powerful impact. At Galaxy Zoo[2], 80,000 astronomers and members of the public have manually classified the morphology of one million galaxies in less than three weeks. Open curation has also been applied to Biology, for example through the gamification of protein folding in the FoldIt[3] web application.

Open curation has also been sought on general topics. Among many: the Open Mind Common Sense initiative has asked users to click on common sense sentences that could be true, constructing a 450,000 assertions knowledge base [Sing 02]. The LabelMe[4] web application asks users to draw the contour of objects in complex images and classify them following a vocabulary (tree, building, ship, person etc.) [Torr 10].

In contrast, most molecular biology annotation is characterised by a high number of instances, but also the need for significant background knowledge, whether acquired through academic training or personal interest. Figure 2.1 tries to position some public knowledge collection efforts along the two problem boundaries listed above: how much data is there to be annotated and how difficult it is for a collaborator to annotate one entry. Several other positionings would be equally likely, as the total data in a domain is very difficult to estimate: how many galaxies are there in the universe, or species on Earth? Similarly, what is the amount of data that should be curated: all the galaxies that exist in all publicly available sky surveys? Should all sequenced proteins be curated?

As for the curation axis, the complexity could depend on annotation granularity or on how curation difficulty is defined (by the background knowledge or by the time needed?). In general, however, the upper panels are where annotators need to be more specialised, while the right side panels might need automatic curation methods most. In practice, Wikipedia (dashed rectangle) shows that an extended public community does not need to be unspecialised and that it can tackle a high number of entries with limited automatic support. In my opinion, the upper right panel is where integrating both open manual and automatic curation could give a solution to the maintenance and update of the billions of entries dispersed in molecular biology knowledge and databases, and this is where the results of this work may best apply.

### 2.1.1 Example problem: enzyme function annotation

Enzyme function is important for biological, medical, environmental and industrial reasons. Enzymes can cause disease when defective, can confer antibiotic resistance to micro-organisms, can enable a yeast to produce drugs or better beer. Enzymes are part of our everyday life. They are in our detergents, our bread and our contact lens solutions. The study of an enzyme starts from its sequence. Here I do not consider the technical and social challenges of full genome an-

[2] Galaxy Zoo `http://www.galaxyzoo.org`

[3] FoldIt `http://fold.it/portal/`

[4] LabelMe `http://people.csail.mit.edu/torralba/publications/labelmeApplications.pdf`

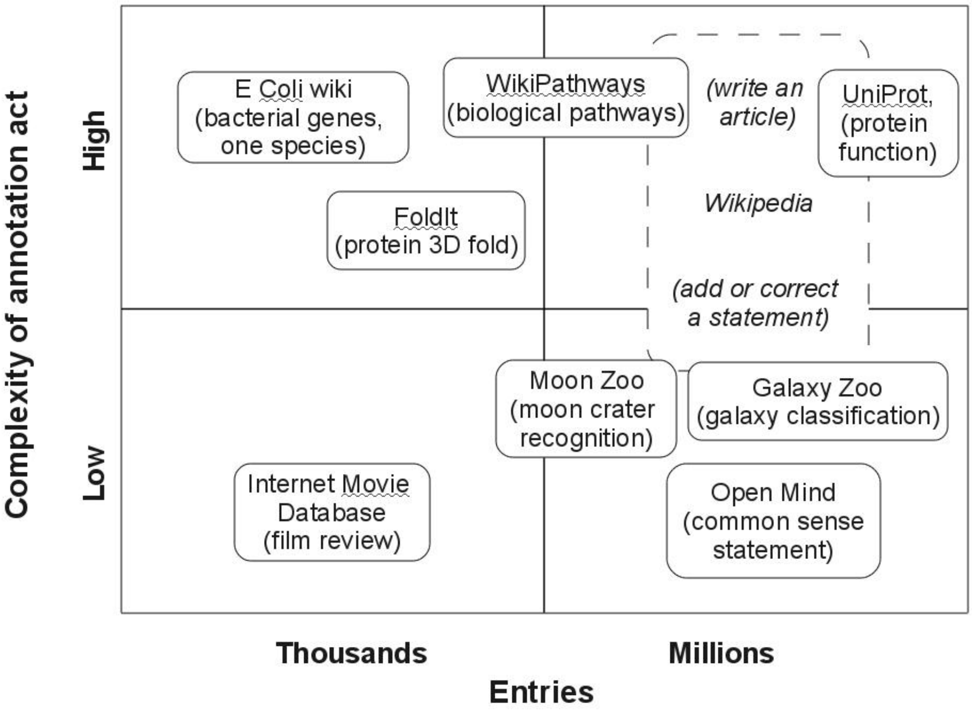

Complexity of annotation act
High
Low
E Coli wiki (bacterial genes, one species)
WikiPathways (biological pathways)
(write an article)
UniProt, (protein function)
FoldIt (protein 3D fold)
Wikipedia
(add or correct a statement)
Moon Zoo (moon crater recognition)
Galaxy Zoo (galaxy classification)
Internet Movie Database (film review)
Open Mind (common sense statement)
Thousands
Millions
Entries

Figure 2.1: *Big Data, Big Annotation or both? This figure tries to position (qualitatively) some public knowledge collection efforts along two axes: how much data is there to be collected or annotated and how difficult it is for a collaborator to annotate one entry. Several other positionings could be equally valid, depending on how the complexity of an annotation act is defined.*

notation, for example in terms of collating fragments from high-throughput sequencing, recognising genes or anonymising human sequences, but I rely on protein sequences being already recognised and publicly available. Even excluding this, there is no clear estimate of the size of the task of annotating the function of all proteins. The annotation of enzyme function is a subproblem of annotating all proteins' function. One that involves fewer categories – since many structural functions can be compacted into one "non-enzyme" function – but equally daunting.

Estimates of the number of species existing on Earth range from 3 to 30 million and are more working hypotheses than estimates, as stressed by several authors [May 88, May 92, Stor 93, Stor 07, Erwi 91]. Even the concept of species becomes less relevant for rapidly proliferating bacteria and viruses with extensive horizontal transfer of genetic material. Each species can then have from a few dozens to tens of thousands of proteins (or splice variants). As for the functional annotation, a considerable fraction of proteins can have multiple molecular functions, and hence multiple annotations, also depending on the terminology or ontology used to express the function. It is also possible for the same biochemical function to correspond to different cellular functions when occurring in different tissues or organs. The current average for a UniProt Swiss-Prot [Cons 10] protein entry is four Gene Ontology [Ashb 00] annotations [Camo 04]. These estimates generate scenarios in the range of millions of annotation acts to curate all enzymes.

On the positive side, judging from general studies and collections such as UniProt UniRef [Suze 07], protein sequences typically show extensive sequence homology. For example, UniRef100 clusters show that around 10% of proteins are identical in sequence to other existing – and possibly already annotated – proteins, allowing for safe direct transfer of annotation and a reduction in the overall annotation task. However, biological paradigms are not made to be simple for curators: cases exist where only the cellular location of a protein, and not its sequence, defines its active functionality. A well known example is the enzyme lactate dehydrogenase which catalyses the interconversion of pyruvate and lactate, but also acts as structural protein in avian and crocodilian eye lenses [Wist 87]. Other rare cases involve differential expression, changes in ligand concentration, complex formation, alternative translation or post-translational modification [Todd 02].

In addition, the task could be reduced by annotating only some representative genomes, for example only a few per taxonomic division. The Catalogue of Life [Bisb 04], compiled from more than a hundred published taxonomic databases, divides the approximately 1,400,000 species known so far into about 98,000 taxa (September 2011), and estimates – conservatively compared with other research – that this represents two thirds of existing species. So should all species or only some be annotated? And should these model organisms be equally distributed among taxonomic divisions? In fact, for historical, practical and medical reasons there is a known over-representation of mammals (mouse, rat, human, apes) and vertebrates (*Danio rerio*, *Xenopus leavis*) among model organisms and a lack of coverage for other divisions, especially invertebrates.

Considering the annotation of only one species for each of the about 100,000 taxonomic domains, an average of 1000 proteins per species (up to 25,000 for human and 100,000 for the *trychomonas* parasite, to give a few examples using UniProt as source) and three functional annotation per protein (one for each of the Gene Ontology divisions: biological process, cellular component and molecular function) we obtain a total of around 300 million protein annotations (270 million after a hypothetical elimination of identical sequences). About 30% of these sequences could be enzymes, but we do not know which.

The curation community size is also very difficult to estimate; there could be millions of untapped minds ready to help. However, to give an example, it took nine years and about 440 million edits to build the 22 million pages of all the Wikipedia projects (in different languages) existing in Jan 2011. However, only a fraction of the current 13 million registered Wikipedia users might have some molecular biology expertise, formal or informal. Obviously, potential contributors to the protein annotation task need not be current or future wikipedians. We do not know how many curators contributed to the effort, but 535,698 proteins were added to the manually curated Swiss-Prot as of April 2012, over a period of 25 years (with a notable rate increase in recent years: 40% of the total entries was annotated in the last 6 years and 25% in the last 3 years).

Protein curation challenges have been successfully extended to the community of university students through initiatives such as Metagenes[5] [Hing 08] and CACAO[6] (Community Assessment of Community Annotation with Ontologies) where student teams compete by annotating proteins with Gene Ontology terms and refuting the annotation of their adversaries. How collaborative these initiatives are will depend not simply on how many people participate (this merely makes them more parallel) but how (and by whom) the annotations are reconciled and integrated into existing knowledge bases.

## 2.2 Current solutions for manual curation

Molecular biology knowledge bases are generally organised around individual pages or entries representing biological entities (for example genes, proteins or genetic variants). In this work, I define a *knowledge base* as the result of an effort to edit and aggregate all knowledge regarding a collection of biological entries, whereas I am less concerned about biological *databases* or *archives*, defined as initiatives that archive large quantities of data without additional aggregation and manual curation.

Examples of manually curated knowledge bases include UniProt Swiss-Prot [Cons 11], KEGG (Kyoto Encyclopaedia of Genes and Genomes) [Kote 12, Kane 10] and PDBWiki [Steh 10]. UniProt Swiss-Prot is one of the richest collections of manually curated protein entries. PDBWiki allows the community to annotate the protein structures entries deposited in the Protein Data Bank [Rose 11]. Examples of archives are the Protein Data Bank [Rose 11] and the EMBL (European Molecular Biology Laboratory) Nucleotide Sequence Database [Coch 09], where 3D structures and sequences are deposited, respectively. Only the depositing authors can annotate these entries or correct them in PDB, while any registered user can edit them in PDBWiki. Archives contain the raw material for curated knowledge bases and are all growing rapidly [Coch 09], thanks also to social workflows that require depositing sequences and structures upon publication or to justify public funding [Howe 08].

The knowledge bases can also be distinguished depending on the possibility for individuals to contribute: *open* knowledge bases allow registration and editing by any non-malicious volunteer, while *closed* knowledge bases can only be edited by a given set of individuals, usu-

[5] Metageneshttp://annotathon.org/LastaccessedinMay2012.

[6] CACAO http://ecoliwiki.net/colipedia/index.php/CACAO_0.1LastaccessedinMay2012.

ally expert curators employed or invited by the data provider. These categories are not sharply defined. Archives can allow authors to change their submissions, often to correct errors, as happens, for example, in PDB and EMBL. Also, closed knowledge bases curators often accept (and sometimes seek) suggestions by other domain scientists. In general, however, open knowledge bases will provide the collaborative software support to publicly and directly correct, justify and discuss the deposited knowledge. Accordingly, I classify as closed knowledge bases those that rely only on e-mail or web forms to receive suggestions for changes and give final vetting power to internal curators.

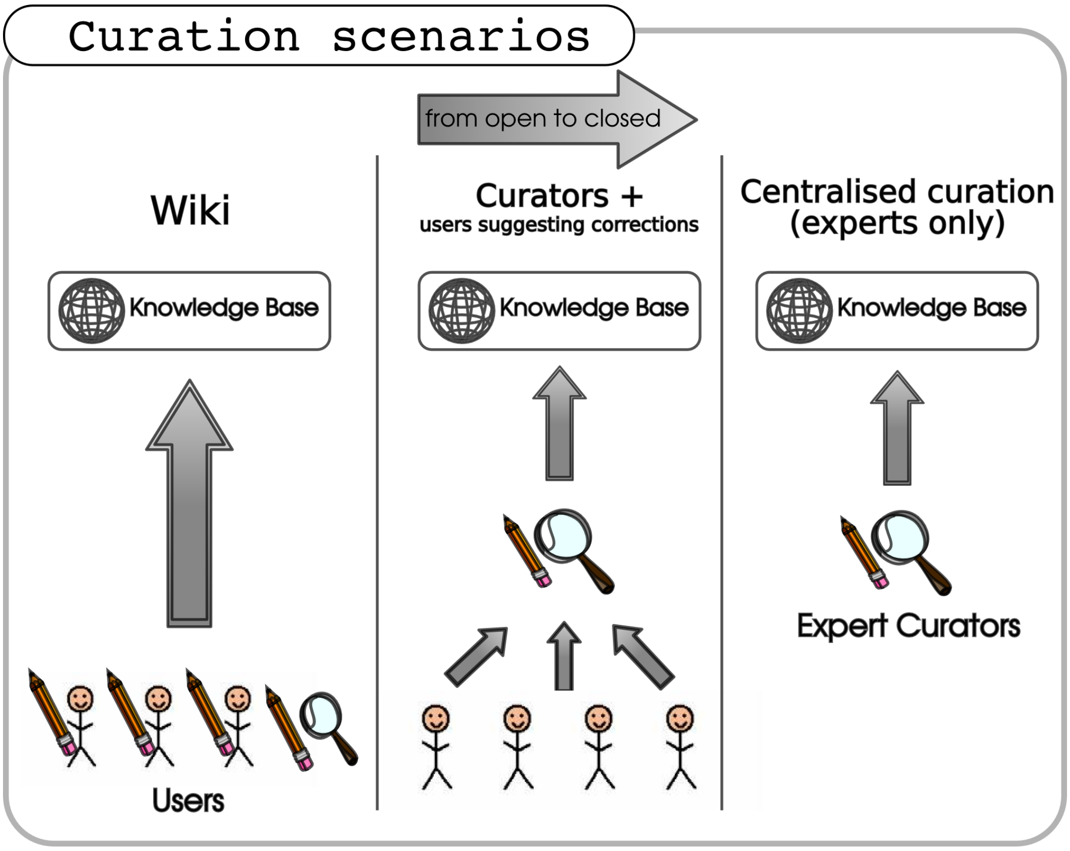


Figure 2.2: *Curation scenarios: wiki curation versus centralised curation. The left panel represents wiki style curation (*open *to all registered users), the middle panel represents* closed *curation (knowledge and edits from experts only), while the right panel represents a mixed model (edits only by expert curators but other users feeding suggestions).*

Another axis along which knowledge bases can be distinguished is "collaborativeness" of annotation, defined as the presence of opportunities for collaborative annotation of a given entry. That is, the provision of software or social support to let more than one user annotate an individual record, and to view and discuss its annotation and its justification. Most wiki software have these affordances, but they are used in varying degrees. Nothing stops a user from "adopting" an entry nobody happens to be interested in and hence see his or her annotations go unchallenged. This concept is distinct from open/closedness, the possibility for annotators to spontaneously join the annotation effort. Nothing stops a closed knowledge base from having a highly collaborative software support and social environment, *within* a closed group of hired or volunteer curators. In practice, in closed knowledge bases, this quality of the process is often not visible to outsiders. It is very difficult to attempt a classification of the collaborativeness of closed knowledge bases only based on curation manuals or similar public documentation on their web pages.

## 2.3 Wikis

Previous work has highlighted the limits of manual curation in keeping up with high-throughput molecular biology data [Baum 07]. A possible solution has been identified in open collaboration, often using wiki-like software [Salz 07, Wang 06]. Wiki software provides a collection of in-line editable web pages, with a syntax (or Graphical User Interface) that makes it easy to create and link those pages, and search them. Wikis attract collaborators through a very low technical entrance barrier [Brya 05]. This is different from saying that wikis are simple systems. What makes or breaks a wiki is its community of users. Wikis have a public image as anarchic, free-for-all communities. However, from the last five years of research, wikis (and especially Wikipedia) emerge as complex and highly regulated systems, in terms of integration of software features and social control [Stvi 08].

Effortless editing is just one of the functions of a wiki. The users cannot be at ease with reorganizing other peoples' content without the possibility to undo mistakes. For this reason, most wikis come with a full version control system. Each page has a history of past versions, which can usually be compared in a separate view, highlighting differences. Each change is stamped with the user name of the author and a time stamp. Users can also register their e-mail address to receive updates when a page is changed. The combination of these two software features is what makes Wikipedia editors able to revert vandalism in a matter of minutes [Prie 07].

### 2.3.1 Measuring wikis

Wikis have been evaluated as knowledge sources, with most research concentrating on Wikipedia. Krieger *et al.* [Krie 09] have divided Wikipedia research into five main areas: quantitative understanding of participation, connecting Wikipedia users to work, trust and article quality, analysing the policy and structure, and using Wikipedia as a corpus towards another goal (such as Natural Language Processing).

The first area of research – *quantitative understanding of participation* – exists thanks to the logs of wiki edits which, incidentally, generate a precious history of authorship, used in this and past studies to measure social collaboration [Wilk 07, Vieg 04, Adle 08]. Chapter 3 of this thesis profits from these logging features to measure, model and evaluate examples of open-collaborative wikis, comparing them with the few closed knowledge bases in molecular biology that offer analogous edit histories. Previous work has also extensively measured the growth, authorship and edit distributions of various language editions of Wikipedia [Orte 07b, Voss 05]. In particular, authors have tried to gauge how "democratic" Wikipedia is in terms of how dominated it is by few major contributors. Kittur *et al.* [Kitt 07] suggests a changing picture, with Wikipedia driven by "elite" users early on and a more recent shift towards "common" user participation. In contrast, Ortega *et al.* [Orte 08] finds a greater level of inequality (with less than 10% of authors responsible for more than 90% of contribution) in all Wikipedia language editions and throughout their recent history, with a more mixed, participative situation only at the very beginning, in line with the results in Chapter 3.

The literature only compares Wikipedia with itself at different times or different language versions, or with open source initiatives. Chapter 3 will introduce closed system into the comparison, finding unexpected similarities between open and closed systems on many "democracy" measures, suggesting either that closed systems are more democratic than expected or that new and better defined metrics are needed to discriminate between open and closed systems. The findings also suggest that the observed long-tail distributions of edits might be a more general characteristic of human knowledge collections.

### 2.3.2 Wikipedia and knowledge quality

The main point of collaboration is that it could improve knowledge quality, that is, if we believe the empirical Linus law of software development: "given enough eyeballs, all bugs are shallow" also defined as: "Given a large enough beta-tester and co-developer base, almost every problem will be characterized quickly and the fix will be obvious to someone."[7] (Linus Thorvald).

If the quantity of wiki articles is easy to measure, much less so is the *quality* of wiki content. The most quoted and commented article is the Wikipedia vs Britannica comparison[8] [Gile 05] (and Britannica's refutation[9] [Brit 06]). This article suffers from clear limitations, the most obvious being that it was not peer-reviewed and compares only 42 entries in the two encyclopaedias. More revealing is the immediate media storm it raised and the pride of the Wikipedia community in having immediately corrected the factual errors highlighted[10]. The open community accent was on currency as much as on factual correctness. Wikis' stress on perfectibility of knowledge chimes well with scientific thought, but their current stress on cur-

[7] http://en.wikipedia.org/wiki/Linus\%27_Law Last accessed Sep 2011

[8] http://www.nature.com/nature/journal/v438/n7070/full/438900a.html Last accessed Sep 2011

[9] http://corporate.britannica.com/britannica_nature_response.pdf Last accessed Sep 2011

[10] http://en.wikipedia.org/wiki/Wikipedia:External_peer_review/Nature_December_2005 Last accessed Sep 2011

rency of information also make it problematic to cite [Wate 07] and evaluate wikis content over time. The perfect balance between stable but obsolete information and current but uncapturable is certainly difficult to achieve.

Much effort has been devoted to finding information quality metrics specific for Wikipedia [Stvi 05, Stvi 08]. This is a challenging area of research as Wikipedia has, paradoxically, grown beyond the possibility of independent and manual evaluations of a significant portion of its total content. Wilkinson and Huberman [Wilk 07] have tried to relate Wikipedia article quality with edit numbers: more popular articles receive a higher number of edits and hence generally become richer in content and more polished. Nielsen [Niel 07] has tried to use the number and impact factor of scientific quotations as a proxy for article quality, finding frequency of citation similar to scientific literature. Evaluation is more practical for specific content areas. Medical sciences have widely participated in Wikipedia and wikis evaluation, particularly to assess their utility in teaching [Boul 06], but also as an information source for patients. Clauson *et al.* [Clau 08] find existing drug information in Wikipedia mostly correct (80 factual elements were checked), but not very complete. That is, when information existed it was correct, but many entries lacked particular clinical indications. However, it found the score significantly improved three months after. Similar high quality, but not complete coverage, was found in the comparison of 25 biographical entries [Rose 06]. However limited in scope some of these studies might seem, it is important to highlight that comparable quality evaluations for molecular biology knowledge bases have not been extensive. Chapter 4 of this thesis will show that disagreement between data sources can be dramatic.

### 2.3.3 Wiki viability

In addition to quality, quantity has its importance. Another main point in favour of openness is that it can lead to mass collaboration. One of the most referenced examples is Nupedia, the closed, professionally-curated ancestor of Wikipedia that stopped activity after collecting only twenty articles in eighteen months of activity, where Wikipedia collected million of articles in an equivalent time scale. Initial research on what makes a wiki thrive has started with a one-point-in-time analysis of several thousand wikis [Roth 07], observing interesting relations such as that while activity directly scales with content size, the number of users does not seem to. In fact, user density (number of users per content page) might *reduce* the growth of content [Roth 08]. Roth [Roth 07] also notes that, to attract voluntary content, topic appeal is obviously important, which might confer an advantage to all-purpose projects such as Wikipedia. An analysis of 360 medium size (400 to 20,000 users) MediaWiki-based wikis in time [Roth 08] notes that “user activity correlates very strongly with wiki growth, not only in terms of content production (which is to a certain extent unsurprising) but also new member recruitment. The effect becomes stronger with wikis that are initially populated to a significant extent: the more users are actively editing, the more a wiki grows in content and population." The model described in Chapter 3 could be the basis for experimenting on the parameters that make a wiki thrive. This work, though, does not aim at directly contributing to this very interesting area of research, I limit myself to assume that mass collaboration is possible given certain conditions.

### 2.3.4 Modelling wikis

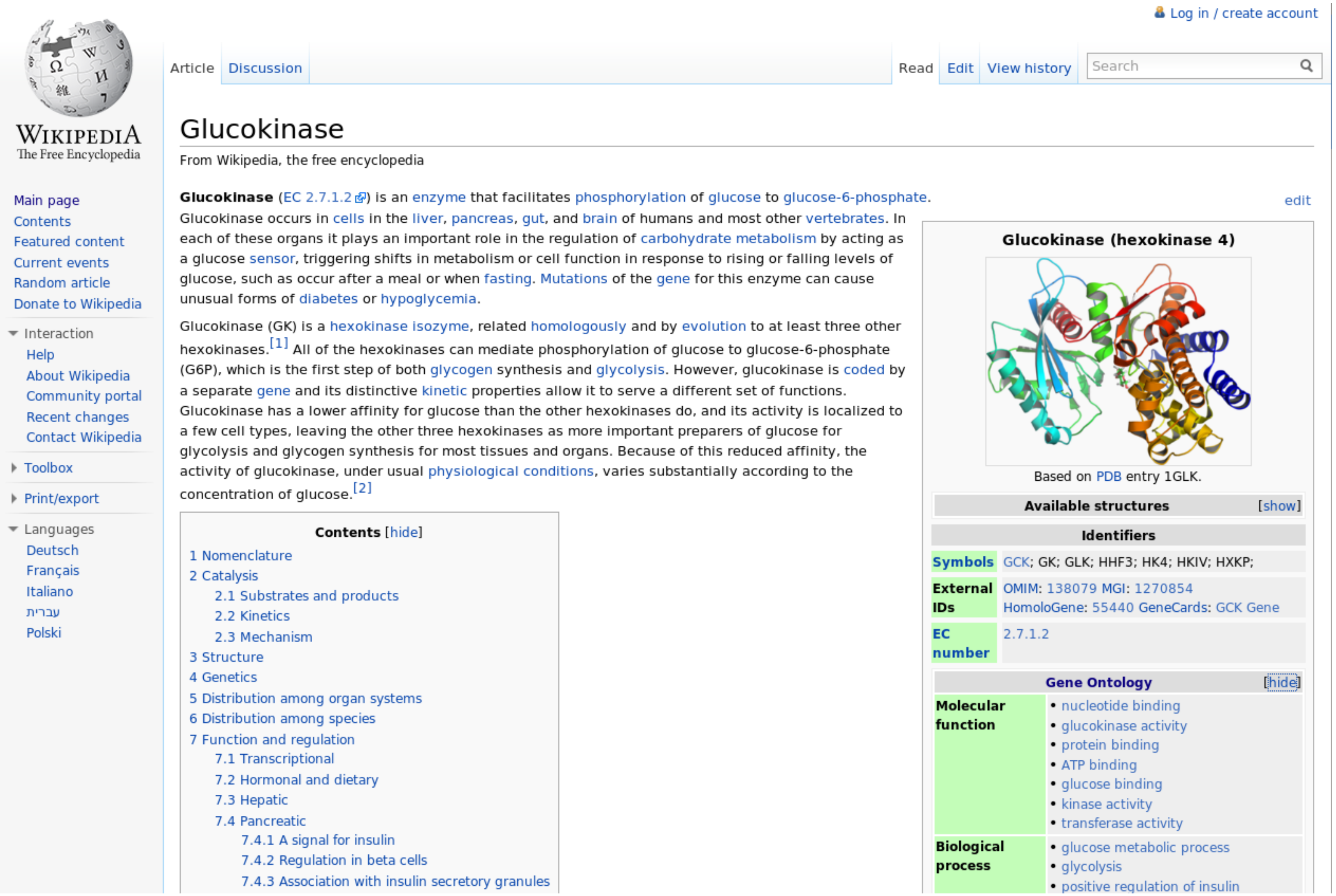
There are few publications on modelling and simulating wikis. Authors such as Stuckman and Purtilo [Stuc 09] advocate the modelling of wikis given that so many wikis present the same distributions of editing and authorship. Troitzsch [Troi 08] developed an agent based simulation of Wikipedia that models collaborative writing of text, authority, plagiarism and how different regimes emerge when agents comply to or violate social norms. Crandall *et al.* [Cran 08] created a simulation using Wikipedia historical data to quantify how shared interactions (editing of the same article) could predict participants further actions. They find that similarity of interests is a better predictor of future behaviour of a given user than direct social interaction, that is, a person is more likely to share a future pattern of edits with people who have similar interests than people who edited the same pages. This result may reflect the fact that the majority of edits on Wikipedia are minor corrections which do not necessarily qualify a person as interested in the page they have just edited, while belonging to a particular Wikipedia community or portal gives a clearer indication of what they will edit next.

Xu *et al.* [Xu 08] represent "facts" as strings of varying length as opposed to true/false values and they only evaluate their model visually, in the style of [Vieg 04] History Flow, against a few Wikipedia articles. Xu *et al.* [Xu 08] also explore the impact of vandalism in Wikipedia not only in simulation, but by the perplexing (to say the least) act of vandalising real Wikipedia articles and then timing how long it took for someone to revert them. Apparently, thanks to their efforts, Wikipedia users looking for "Spallation" in 2008 were exposed for an entire day to a vandalised text while the researchers were waiting for someone to restore it.

### 2.3.5 Molecular biology wikis

Wikis have become popular in molecular biology for online writing of documentation and laboratory protocols, such as in OpenWetWare[11] with 20,000 pages and over 61 million views, and their popularity has extended to data curation. Popular off-the-shelf wiki software such as MediaWiki (or variants adapted to the task at hand) are being used to create editable knowledge bases or to re-annotate archive data. Curation efforts using wikis are currently under way in many molecular biology projects. Some are organism specific such as EcoliWiki for *Escherichia coli* [Ecol 11] (Figure 2.4) , Xanthusbase for *Myxococcus xanthus* [Arsh 07] or the

[11] `http://openwetware.org/wiki/Main_Page` Last accessed Sep 2011



Log in / create account

Article Discussion Read Edit View history Search

# Glucokinase

From Wikipedia, the free encyclopedia

Main page
Contents
Featured content
Current events
Random article
Donate to Wikipedia

Interaction
Help
About Wikipedia
Community portal
Recent changes
Contact Wikipedia

Toolbox

Print/export

Languages
Deutsch
Français
Italiano
עברית
Polski

edit

**Glucokinase** (EC 2.7.1.2) is an enzyme that facilitates phosphorylation of glucose to glucose-6-phosphate. Glucokinase occurs in cells in the liver, pancreas, gut, and brain of humans and most other vertebrates. In each of these organs it plays an important role in the regulation of carbohydrate metabolism by acting as a glucose sensor, triggering shifts in metabolism or cell function in response to rising or falling levels of glucose, such as occur after a meal or when fasting. Mutations of the gene for this enzyme can cause unusual forms of diabetes or hypoglycemia.

Glucokinase (GK) is a hexokinase isozyme, related homologously and by evolution to at least three other hexokinases.[1] All of the hexokinases can mediate phosphorylation of glucose to glucose-6-phosphate (G6P), which is the first step of both glycogen synthesis and glycolysis. However, glucokinase is coded by a separate gene and its distinctive kinetic properties allow it to serve a different set of functions. Glucokinase has a lower affinity for glucose than the other hexokinases do, and its activity is localized to a few cell types, leaving the other three hexokinases as more important preparers of glucose for glycolysis and glycogen synthesis for most tissues and organs. Because of this reduced affinity, the activity of glucokinase, under usual physiological conditions, varies substantially according to the concentration of glucose.[2]

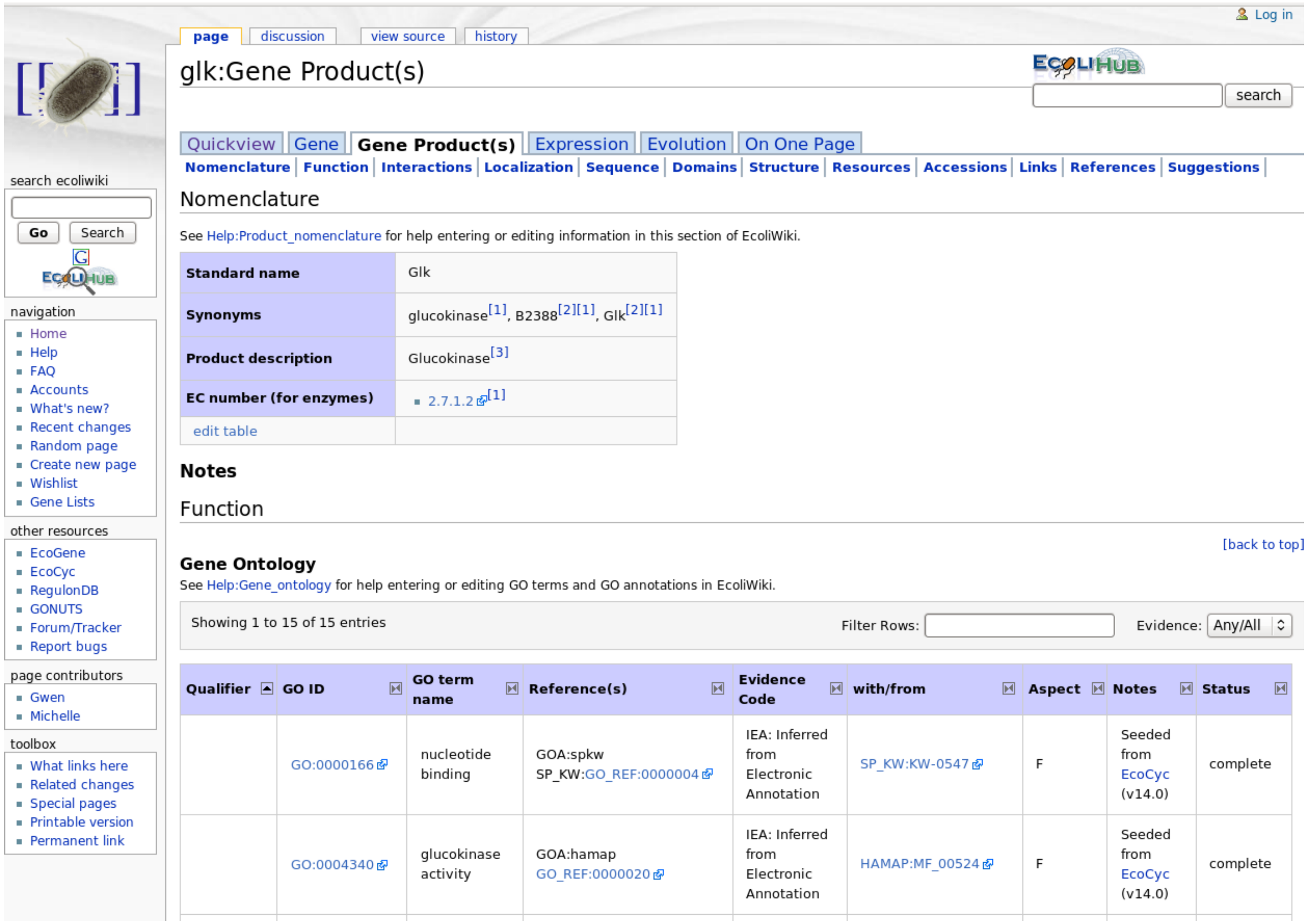
Contents [hide]

**Glucokinase (hexokinase 4)**

Based on PDB entry 1GLK.

| Available structures [show] | |
|---|---|
| **Identifiers** | |
| **Symbols** | GCK; GK; GLK; HHF3; HK4; HKIV; HXKP; |
| **External IDs** | OMIM: 138079 MGI: 1270854<br>HomoloGene: 55440 GeneCards: GCK Gene |
| **EC number** | 2.7.1.2 |
| **Gene Ontology** [hide] | |
| **Molecular function** | • nucleotide binding<br>• glucokinase activity<br>• protein binding<br>• ATP binding<br>• glucose binding<br>• kinase activity<br>• transferase activity |
| **Biological process** | • glucose metabolic process<br>• glycolysis<br>• positive regulation of insulin |

Figure 2.3: *The human glucokinase article from the Gene Wiki portal in Wikipedia. The structured section on the right side of the page contains, among other data and identifiers, the Enzyme Commission code for the protein reaction (EC 2.7.1.2) and Gene Ontology terms to annotate the protein function.*



glk:Gene Product(s)

Nomenclature

See Help:Product_nomenclature for help entering or editing information in this section of EcoliWiki.

| | |
|---|---|
| **Standard name** | Glk |
| **Synonyms** | glucokinase[1], B2388[2][1], Glk[2][1] |
| **Product description** | Glucokinase[3] |
| **EC number (for enzymes)** | 2.7.1.2[1] |

Notes

Function

Gene Ontology

See Help:Gene_ontology for help entering or editing GO terms and GO annotations in EcoliWiki.

Showing 1 to 15 of 15 entries

| Qualifier | GO ID | GO term name | Reference(s) | Evidence Code | with/from | Aspect | Notes | Status |
|---|---|---|---|---|---|---|---|---|
| | GO:0000166 | nucleotide binding | GOA:spkw<br>SP_KW:GO_REF:0000004 | IEA: Inferred from Electronic Annotation | SP_KW:KW-0547 | F | Seeded from EcoCyc (v14.0) | complete |
| | GO:0004340 | glucokinase activity | GOA:hamap<br>GO_REF:0000020 | IEA: Inferred from Electronic Annotation | HAMAP:MF_00524 | F | Seeded from EcoCyc (v14.0) | complete |

Figure 2.4: *A screen shot of the* E. coli *glucokinase article on EcoliWiki. The upper nomenclature table contains the Enzyme Commission code for the protein reaction (2.7.1.2). In the bottom half of the page, the protein function is annotated using not only Gene Ontology terms, but their provenance and evidence code too: the reason for assigning the annotation, usually a wet lab experiment or the prediction of a computational method.*

GeneWiki portal in Wikipedia for human genes [Huss 08, Huss 10] (Figure 2.3). Other wikis tackle molecule resources, such as PDBWiki, the wiki branch of the PDB database, publicly curating 3D protein structures [Steh 10]. In general, wikis that are extensions of existing scholarly communities or established knowledge bases have fared better than top-down approaches such as WikiProteins [Mons 08]. WikiProteins pre-populated a task-specific wiki with millions of concepts from PubMed articles, EMBL, Swiss-Prot and Gene Ontology, but has not seen much activity in the last three years according to the number of edits and editors in its logs.

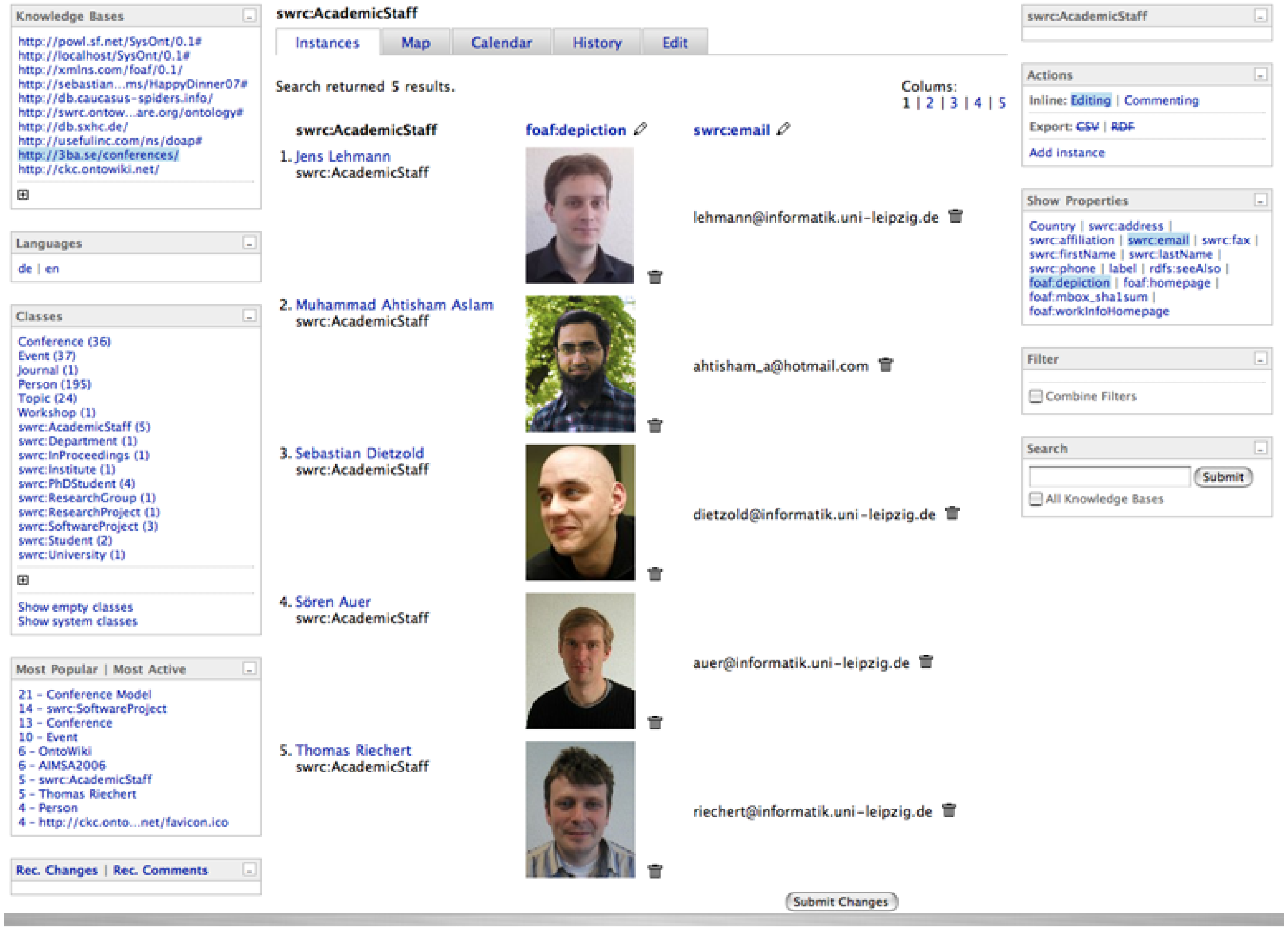
Wikis support data structuring using templates and embedding data tables in traditional wiki articles, generating more easily machine-readable data, as shown in Figure 2.3 for the Gene Wiki portal [Huss 08, Huss 10] a Wikipedia collection of pages on human genes and proteins. Figure 2.4 shows how EcoliWiki has customised the MediaWiki software to structure the data over several page tabs and multiple tables. WikiPathways [Pico 08] has customised it further to allow in-line editing of graphical blob-and-arrow diagrams representing biological networks (Figure 2.6). WikiPathways also supports the import and export of networks in GenMAPP format [Dahl 02, Salo 07] through PathVisio. The use of standard formats is good practice that can help integrating manual and automated curation in wikis.

Semantic wikis offer improved support for storage and editing of structured information. Some initiatives have extended popular existing software, such as Semantic MediaWiki for MediaWiki [Krot 06]. Other, like OntoWiki [Auer 06], have developed dedicated software: a semantic wiki which supports fine grained data collection and in-line editing, shown in Figure 2.5. These wiki software have potential for collaborative curation, but have yet to be widely adopted in molecular biology.

## 2.4 Current solutions for automated curation

Biology has seen extensive application of machine learning techniques to predict biological function, profiting from the fact that evolutionary mechanisms (sequence conservation) make inference of biological function possible across species. Obviously, inference of functional conservation across species is not completely reliable. Here I concentrate on supervised machine learning techniques, where algorithms are trained on known examples, instances associated with zero or more functional classes, and then evaluated on how well they can predict other instances. Supervised learning must train on known data: knowledge bases and structured wikis are perfectly positioned to provide machine-readable examples for automatic learning.

Most of these automated methods are evaluated and applied to batches of examples. For example, in a closed system such as UniProt, the InterPro2GO method is applied to every UniProt release (about once per month) to predict the Gene Ontology functional classification



Knowledge Bases

http://powl.sf.net/SysOnt/0.1#
http://localhost/SysOnt/0.1#
http://xmlns.com/foaf/0.1/
http://sebastian...ms/HappyDinner07#
http://db.caucasus-spiders.info/
http://swrc.ontow...are.org/ontology#
http://db.sxhc.de/
http://usefulinc.com/ns/doap#
http://3ba.se/conferences/
http://ckc.ontowiki.net/

Languages

de | en

Classes

Conference (36)
Event (37)
Journal (1)
Person (195)
Topic (24)
Workshop (1)
swrc:AcademicStaff (5)
swrc:Department (1)
swrc:InProceedings (1)
swrc:Institute (1)
swrc:PhDStudent (4)
swrc:ResearchGroup (1)
swrc:ResearchProject (1)
swrc:SoftwareProject (3)
swrc:Student (2)
swrc:University (1)

Show empty classes
Show system classes

Most Popular | Most Active

21 - Conference Model
14 - swrc:SoftwareProject
13 - Conference
10 - Event
6 - OntoWiki
6 - AIMSA2006
5 - swrc:AcademicStaff
5 - Thomas Riechert
4 - Person
4 - http://ckc.onto...net/favicon.ico

Rec. Changes | Rec. Comments

swrc:AcademicStaff

Instances | Map | Calendar | History | Edit

Search returned 5 results.

Colums:
1 | 2 | 3 | 4 | 5

| swrc:AcademicStaff | foaf:depiction | swrc:email |
|---|---|---|
| 1. Jens Lehmann<br>swrc:AcademicStaff | | lehmann@informatik.uni-leipzig.de |
| 2. Muhammad Ahtisham Aslam<br>swrc:AcademicStaff | | ahtisham_a@hotmail.com |
| 3. Sebastian Dietzold<br>swrc:AcademicStaff | | dietzold@informatik.uni-leipzig.de |
| 4. Sören Auer<br>swrc:AcademicStaff | | auer@informatik.uni-leipzig.de |
| 5. Thomas Riechert<br>swrc:AcademicStaff | | riechert@informatik.uni-leipzig.de |

Submit Changes

swrc:AcademicStaff

Actions

Inline: Editing | Commenting
Export: CSV | RDF
Add instance

Show Properties

Country | swrc:address | swrc:affiliation | swrc:email | swrc:fax | swrc:firstName | swrc:lastName | swrc:phone | label | rdfs:seeAlso | foaf:depiction | foaf:homepage | foaf:mbox_sha1sum | foaf:workInfoHomepage

Filter

□ Combine Filters

Search

Submit
□ All Knowledge Bases

Figure 2.5: *An OntoWiki screen shot showing a contacts management page. The pencil and trash bin icons link to in-line editing of each entry. The left and right frames contain automatically generated links to the ontology classes and properties for the entries.*

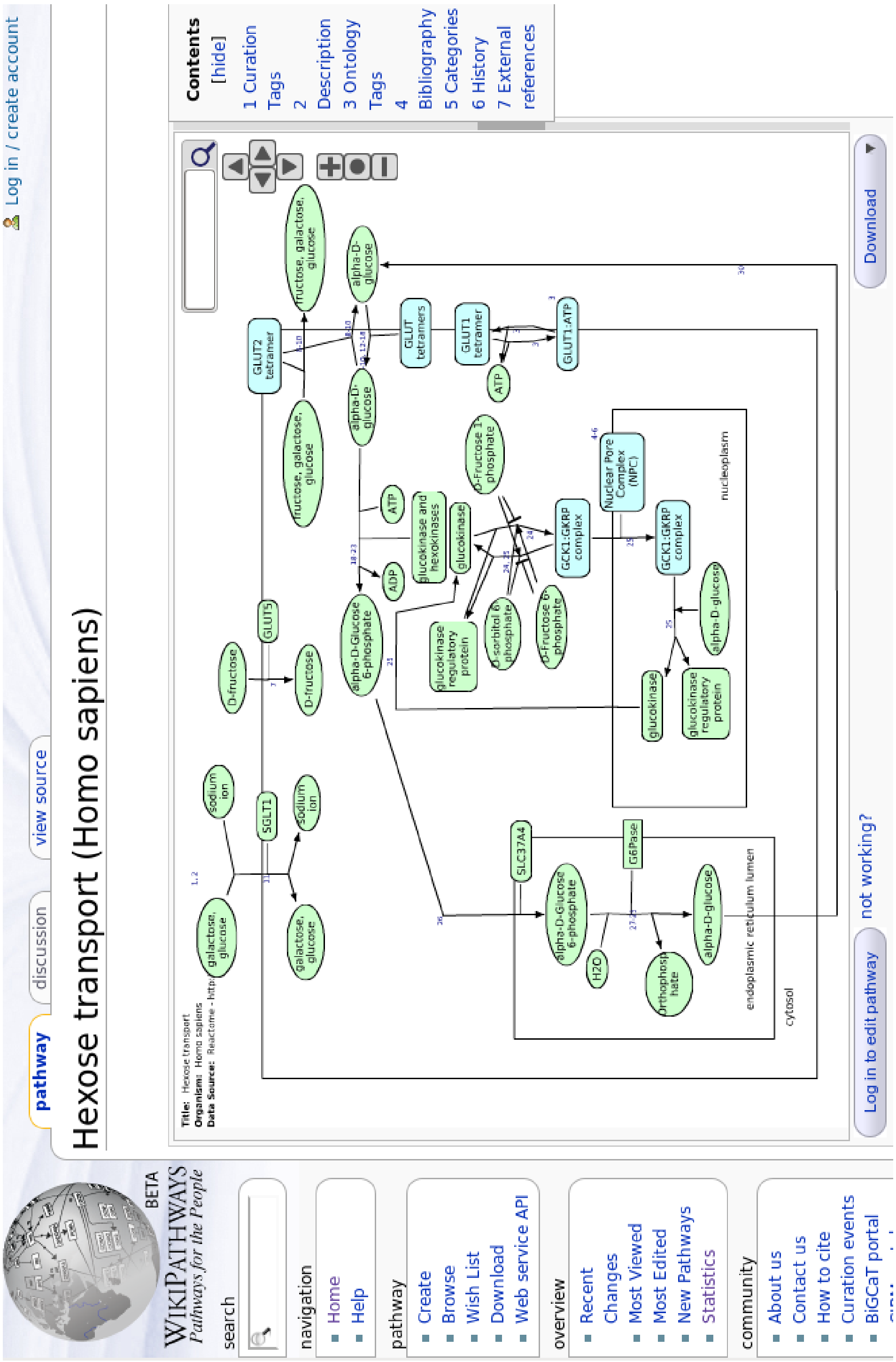


Figure 2.6: *A screen shot of the human hexose transport pathway in WikiPathways. The network nodes and connections can be added, deleted or edited in the web page and exported in various semantic (.owl), image (.svg, .png) or text formats (.txt, .pdf).*

of all proteins [Burg 12]. EcoCyc and MetaCyc have similar procedures to try to complete metabolic pathways [Kese 11, Late 12]. This is the typical workflow also used in chapter 4, with the machine learning applied to a batch of protein examples. Chapter 5 of this thesis explores how to integrate the machine learning method more into the curation workflow.

The resulting predictions can be used as they are or as the basis of further manual annotation. However, in closed knowledge bases, but also in wikis, there is a general lack of visual indication of provenance: external and internal links appear in the same URL style, there are few clues about which elements in a page are manually curated, automatically curated or linked from other databases.

Open systems also have automatic scripts for basic error checking, data import or link updating (wiki bots), but, to my knowledge, they do not include more complex machine learning methods. The integration of machine learning methods, and their community acceptance, would require clever methods to present and preserve manual curation alongside automatic prediction, as well as the justification and provenance of the predictions. This interesting area of research would require very good human-computer interaction design.

### 2.4.1 Predicting enzymatic function

Assigning enzymatic function to the proteins in a genome is one of the first essential steps of metabolic reconstruction, important for biology, medicine, industrial production and environmental studies. Without precise annotation of the reactions a protein can perform, the subsequent pathway assembly and verification becomes problematic [Pitk 10]. Metabolic flux studies that aim to understand diseased states or biomass production become almost impossible. Unfortunately, at the current rate of genome sequencing and manual annotation, manual curation will never complete the functional annotation of all available proteomes [Baum 07].

Tetko *et al.* [Tetk 08] used principal component analysis to show that the highest contributors to the performance of various protein function prediction methods were InterPro signatures. InterPro is an extensive database of conserved sequence signatures and domains [Hunt 09], an umbrella database including twelve other signatures sources: Pfam, PRINTS, PROSITE, SMART, ProDom, PIRSF, SUPERFAMILY, PANTHER, CATH-Gene3D, TIGRFAMs and HAMAP. Sequence signatures can have different lengths and tolerance for mutations, ranging from short catalytic sites with a stringent requirement in terms of amino acid type to entire protein domains composed of hundreds of amino acids. InterPro also provides a publicly available browser based, web service and offline software to match signatures to any genetic sequence (InterProScan [Muld 07]).

*Multi-label classification*, that is, the direct association of *multiple* functions with each protein is particularly important or enzymes. A single enzyme can perform different reactions, either due to the presence of multiple catalytic sites, or because of substrate promiscuity, or by regulation of a single site, and can hence be associated with multiple EC numbers.

Despite some known limitations, such as some inconsistencies between the rules set by the nomenclature committee and the actual EC number definitions [Egel 10], the NC-IUBMB Enzyme Commission (EC) nomenclature is used to define enzymatic reactions, as it is the current standard for enzyme function classification. The EC nomenclature uses a four digit code, such as EC 1.2.3.4 to represent an enzymatic reaction. These four digit codes defined by the Enzyme Commission will be referred to from now on as "EC numbers"[12]. The first three digits of an EC number represent an increasingly detailed definition of chemical reaction catalysed, while the last digit represents the accepted substrates.

Multi-label learning can take multiple EC numbers, and their hierarchical relationship, into account more coherently and effectively than creating an individual classifier for each class.

### 2.4.2 Sequence based methods

Other sequence based methods for the prediction of EC numbers include EFICAz [Tian 04], ModEnzA [Desa 11] and PRIAM [Clau 03]. PRIAM uses a set of position-specific scoring matrices (profiles) specific for each EC number to predict the existence of a given EC function somewhere in a fully sequenced genome. EnzML, ModEnzA and EFICAz assign EC numbers to individual protein sequences or fragments. ModEnzA builds hidden Markov model profiles of positive and negative sequences specific for each four digit EC numbers, but partial or multiple EC numbers cannot be assigned. EFICAz can assign multiple EC numbers of exactly three or four digits by weighting information from four sequence based prediction methods using functionally discriminating residues for enzyme families, pairwise sequence comparison, Pfam enzyme families and Prosite patterns (EFICAz2 [Arak 09] is enhanced using Support Vector Machine methodology). EFICAz, ModEnzA and PRIAM are further discussed and quantitatively compared with EnzML in Section 4.2.7 on page 65.

### 2.4.3 Multi-label prediction

Multi-label learning has been successfully applied to predict FunCat protein functions in yeast [Clar 02], GO functions in yeast [Baru 06], CYGD functions in yeast [Lanc 04], FunCat and GO functions in yeast and plants [Schi 10] and other species [Vale 08], but had not yet been extensively applied to the prediction of enzyme functionality. A multi-label support vector machines methodology was used in the past to predict EC numbers but only up to the second EC digit (e.g.. EC 1.2.-.-) and only on 8,291 enzymes [Cai 04]. Hierarchical classification was also applied to about 6,000 enzymes from KEGG, obtaining over 85% accuracy in predicting

[12] It is important to note that "EC class" is often used in the literature to indicate the first digit of an EC number. In this work the term "EC class" will not be used to avoid confusion with the term "class" (class label) used in machine learning.

four digit EC numbers [Asti 08].

## 2.5 Active learning

Active learning is an umbrella definition for all methods that allow the machine learning algorithm to choose the data from which it learns. The resulting methods can sometimes achieve greater accuracy with fewer training labels, in particular for problems where unlabelled data are cheap and abundant, but labels are expensive to obtain, for example because they require an expert ("oracle" is the term sometimes used in the area to refer to either a human expert or a computational system used as a source of reliable labels).

In molecular biology, basic information about instances, including gene and protein sequence, or even protein structure, is relatively easy to obtain. However, functional annotation requires at best expert annotators and at worst carefully designed and expensive wet laboratory experiments. For the enzyme curation scenario the two potential advantages of active learning are to improve the speed and predictive performance of the machine learning, and to avoid redundant manual curation effort. Active learning for large scale, multi-label prediction problems is still an open area of research, to which this thesis will contribute a real life example.

### 2.5.1 Pool-based scenario

Settles [Sett 09] describes three possible scenarios of learner querying: 1. Membership query synthesis, 2. Stream-based selective sampling and 3. Pool-based sampling. *Membership query synthesis* requires the *generation* of artificial instances. For example, in character recognition, this would include examples of partial or stretched characters. This would be complex to execute for protein sequences and awkward for annotators, for which such artificial proteins would hardly have concrete meaning.

In *stream-based selective sampling*, instances arrive as a stream, for example, in a sequential order from a sensor or from a text (in part-of-speech recognition). The learner has to decide on the spot whether it is worth requesting the label for that example or not, accounting for the possibility that better examples might come along later on. The assumption is that the instances are uniformly distributed and the shape of their distribution is known in advance. In our domain, proteins' examples do appear and are annotated in time, but their distribution is non-uniform and getting a new example might entail a wait of anything between minutes and days.

In *pool-based sampling* a large pool of unlabelled examples is assumed to be available at once (also called *Selective sampling* [Lind 04]). The basic assumption of pool-based active learning is that there are numerous unlabelled examples in the domain (the pool), all with attribute values that are easy to calculate. However, the labelling of an instance incurs a significant cost and labels can only be provided by a trained annotator. The pool instances are then evaluated by some informativeness measure and ranked in the best order to be added to the training set.

Many molecular biology curation problems fit this scenario, as much raw data is available (for example, genetic sequences) but not annotated. In our enzyme domain, UniProt TrEMBL contains 13 million unlabelled proteins, versus only half a million manually curated proteins in UniProt Swiss-Prot. The sequence attributes (InterPro domains) are either already calculated or easy to obtain with the InterProScan service [Muld 07]. However, labelling a protein with its enzymatic class can take a curator from hours to weeks, including the time needed to read the literature, examine the protein sequence and similar sequences or structures.

Pool-based active learning is a strategy where an algorithm first trains on a set of labelled examples and then actively asks for the labels of other examples taken from the unlabelled pool. Ideally, the method should choose the new example in such a way as to optimise the predictivity of the trained algorithm and hence improve predictions on the remaining unlabelled examples. This is done by calculating a utility or informativeness metric for each instance. This utility metric is usually recalculated at each active learning step to account for the change in the dataset (at each cycle one instance moves from the unlabelled to the labelled pool). This limits the parallelisation of the curation task to one instance at a time. Thus until the most informative instance has been labelled, the informativeness of the remaining unlabelled instances cannot be recalculated and hence active learning cannot proceed (the "next-best" instance might be suboptimal).

In more detail:

1. Input: a set of labelled ($L$) and unlabelled ($U$) instances ($L$ can initially be empty).
2. For each instance $u \in U$: calculate the utility metric
3. Find the unlabelled instance with best utility metric (usually the maximum): the *query* instance $q \in U$
4. Label $q$: $q \notin U, q \in L$
5. Train on $L$
6. Test on $U$ and emit evaluation metrics
7. Go to step 2

### 2.5.2 Query strategy frameworks

Different strategies can be used to decide the informativeness of each example and hence find the best *query* instance, the one that will hopefully improve the trained learner the most. The categories below are further detailed in [Sett 09].

**Uncertainty Sampling** [Lewi 94] presents a framework where the active learner queries the instances it is *least certain* how to label. The best instance query $x^*$ becomes:

$$x^* = argmax\,[-\sum_i P_\theta(y_i \mid x) log P_\theta(y_i \mid x)]$$

where $y_i$ ranges over all possible labellings and $P_\theta$ is the posterior probability under the model $\theta$. This approach falls into the category of "confidence" based (or "uncertainty" based) active learning, where the general strategy is to calculate a confidence for the prediction of unlabelled instances, and then take the instances with lowest confidence to be added next. The underlying assumption is that low confidence instances will add more information to the next learning run. That is, if the classifier cannot confidently predict a certain example, it means the classifier lacks information about the particular combination of attributes contained in that example. Hence, providing a label for that example (that set of attribute values), will improve the classifier performance on similar instances.

For binary classification, this is equivalent to selecting and labelling the instance with class prediction confidence closest to 0.5, that is, the least confident in either discarding (0) or accepting (1) the label. This approach is natural with probabilistic classifiers and is also applicable to the K-Nearest Neighbours algorithm [Fuji 98, Lind 04]. In K-Nearest Neighbours the principle becomes to allow each neighbour to vote on the class labels of $x$, with the proportion of these votes representing the posterior label probability. In fact, the same neighbours vote method is used in Mulan BR-kNN algorithm [Spyr 08] to emit a confidence for each predicted label.

**Query-By-Committee** [Seun 92] involves maintaining a number of competing models (the committee) and using the instance on which they most disagree as query. This is particularly suitable for algorithms such as Support Vector Machines. This method is also computationally intensive as it can grow exponentially with the size of the training set [Haus 89], so the committee size would become problematic with a dataset of the size of all existing enzymes.

**Expected Model Change** [Sett 08c, Sett 08a] select the instance that would most change the current model *if we knew its label* as the query. This method is also computationally very intensive as it involves calculating over all possible labels, or combinations of labels where there are multiple labels.

**Expected Error Reduction and Variance Reduction** are similar to the above, but measure how much the generalization error (or variance) of a method would change if an instance were to be added. This also requires averaging over all possible labels for each unlabelled instance.

**Density-Weighted methods** Density-Weighted methods are now discussed in more detail as they overcome a problem of Uncertainty Sampling; what if the most uncertain instance is a one of-a-kind and not representative of many other instances? Such an instance is unlikely to improve a classifier greatly. For enzymes, where the class distribution is very skewed, there is a high risk of choosing as best query instances one-of-a-kind examples. Expected Model Change, Expected Error Reduction and Variance Reduction implicitly avoid the problem by using the unlabelled pool $U$ when estimating future error and output variance, but this is computationally expensive. In addition, the input distribution can also be modelled explicitly.

Information density, well described in [Sett 08c, Sett 08b], uses the principle that instances should be selected not only because they are uncertain, but also because they are representative of the underlying distribution of unlabelled instances:

$$x^*_{ID} = argmax\,(\phi_A(x) \times \frac{1}{U}\sum_{u=1}^{U} sim(x, x^{(u)})^{\beta})$$

where $\phi_A(x)$ is the informativeness of $x$ according to the same base query strategy $A$, for example, uncertainty sampling. The second term weights the informativeness of $x$ by its average similarity to all other instances in the input distribution. The $\beta$ parameter controls the relative importance of this last density term. Settles [Sett 08c] also shows that careful caching of the information density can make this approach as fast as basic uncertainty sampling, an essential aspect for datasets in the order of million of protein instances.

Fujii *et al.* [Fuji 98] adapted this strategy for the K-Nearest Neighbours algorithm to select the instance 1. least similar to the labelled instances and 2. most similar to the remaining unlabelled instances. However, their algorithm is not immediately applicable to the enzyme example. Their approach was developed for text mining and makes use of the role of a word in a sentence, a role for which there is no obvious analog for a protein in the domain of biological function. Various other K-Nearest Neighbours algorithms have been proposed for active learning. Lindenbaum *et al.* [Lind 04] have worked on binary problems with a limited number of examples (maximum 1000) and attributes (maximum 35). In contrast, Jain and Kapoor [Jain 09] have used K-Nearest Neighbours to tackle multi-class active learning on more substantial datasets (hundreds of classes with about 40 to 800 instances per class). This is still significantly smaller than our challenge with four thousand classes and million of instances. Jain and Kapoor consider the Euclidean distance as too simplistic and not reflecting the knowledge we have about the data and the Mahalanobis distance too expensive for large multi-label problems, so they define their own distance metric. However, any distance more complex than Euclidean has the potential to become a bottleneck on million of instances. Chapter 5 will present some methods trying to profit from the ideas described while remaining applicable to very large multi-label problems.

### 2.5.3 Active learning in biology

Active learning has very recently started to be applied to biological problems. Most of the methods that have entered the arena come from the more mature areas of text-mining or image processing. For example, generic text mining methods have been customised to support the annotation of biological papers [Tsur 08, Wall 10]. Active learning for image recognition has been cleverly applied to microscopy, to help researchers define which areas of, for example, a tissue slice, are most informative for classification, and hence reduce the image scanning time and the cell toxicity caused by the chemicals used to highlight the structures of interest [Jack 09].

Osmanbeyoglu *et al.* [Osma 10] applied active learning to identify trans-membrane proteins whose structure, if revealed experimentally, would be maximally predictive of others, an important prediction given that trans-membrane proteins are extremely difficult and hence expensive to crystallise. They use a neural network algorithm they developed previously and apply self organising maps to cluster the unlabelled instances. They then compared four different active learning approaches: random instance selection, node coverage (size of the self organising map node the instance belongs to), maximal entropy (proteins whose feature vectors fall in the nodes with maximum confusion between trans-membrane and non trans-membrane labels are selected) and an ensemble method which alternates node coverage and maximal entropy selection at each iteration. Their domain is binary and has probably less issues with the lack of parallelisation that active learning entails, as not many groups will tackle the crystallisation of a difficult protein at the same time.

A random forest algorithm is used by Mohamed *et al.* [Moha 10] to predict which protein-protein interactions should be experimentally tested to maximally improve the *Homo sapiens* interactome. The results show that with as few as 500 protein-pairs labels selected actively, the classifier achieved a higher precision and recall than with 3000 randomly chosen real protein-pairs. Two of the active learning approaches they used (density based and uncertainty based) are particularly interesting because they attempts to label “batches” of instances. The density based approach applies K-means clustering and then gets a batch of desired instances in proportion to the clusters sizes. The uncertainty based method picks the 250 instances with maximum informational entropy. This method appears to be sensitive to how the first instance is chosen (either randomly or by density). Their problem is defined as binary though, with couples of proteins either interacting or not, so it is not immediately applicable to the multi-label enzyme scenario.

## 2.6 Guided learning

Active learning has two main disadvantages for enzyme curation [Atte 10b, Atte 10a, Atte 11], the first is that most methods serve one instance at a time and hence are not easy to parallelise, and the second is that most methods have serious limitations in finding rare classes, that is, most EC numbers in our scenario.

Attenberg *et al.* [Atte 10a] propose an alternative to active learning to tackle class imbalance. They show that under class skew asking experts to look for examples of the minority class provides much better accuracy than classic active learning strategies such as uncertainty based and query committee. They use the example of classification of web pages in acceptable or objectionable content for online advertisers. They have a problem we do not have though, namely, having to rebuild classifiers depending on advertisers preferences (objectionable could be adult content for one advertiser and children content for another). In biology, experts generally work towards one shared classifier. Also, the task is binary (page objectionable can be true or false) while our problem is multi-class. However, enzymes certainly have a high number of rare classes. Attenberg *et al.* [Atte 10a] use examples of highly skewed classes with ratio of positive examples over negative examples varying between $1/80$ and $1/10^7$. In Swiss-Prot, out of 2,958 represented EC numbers, 2,806 have less than $1/1,000$ examples and 2,037 have less than $1/10,000$ examples. Hence strategies to tackle class skew are most welcome to obtain high accuracy and recall of rare EC numbers as well. In terms of evaluation metrics, multi-label frameworks such as EnzML can produce the macro average recall metric which is particularly sensitive to efficiency in predicting rare classes.

The guided learning approach might superficially seem at odds with active learning. Was it not the aim to rank informative instances *without* knowing their classes? Guided learning starts from the pragmatic intuition that class definitions do exist (such as the description of EC number chemical reactions) and that an expert should be able to find one example, any example, of that class. Most EC numbers have at least one example in UniProt, and for the EC numbers without any example at all (or very few examples), it becomes even more important to quantify the advantage of a guided learning strategy aggressively directed towards the coverage of minority classes.

Attenberg *et al.* [Atte 10a] reports that guided learning dominates active learning even when search cost is eight times the labelling cost. In other words, in their example scenario, it would be more informative for a curator to spend eight minutes finding a positive example of an undesirable web page than it would be to spend one minute labelling each of the eight pages suggested by an active learning strategy. Turning again to the molecular biology scenario, the effort necessary to an expert to find an example of an EC number decided a priori might be equivalent or even lower than the time needed to label eight unknown protein sequences provided by an active learning strategy. In terms of work organisation, guided learning would make it easier to recruit experts who are most knowledgeable or interested in a given class. It would also fit better with the self-assigning of tasks common in research or in collaborative knowledge curation.

Another idea to note is that, if reliable class distributions are not available, guided learning can also be applied to attributes (or InterPro attributes set for enzymes). In our scenarios, attributes are always available for any instance and so they could be used to build a strategy that covers the maximum number of features, either in random order or in order of their frequency in the dataset. Whatever the choice of guided learning (over classes as in Attenberg, or over attributes, as in my approach) it is important to know whether the order of addition of the labelled examples matters, to overcome the active learning limitation of only one ordering leading to an optimum build up of accuracy.

## 2.7 Summary

This chapter summarises the challenge of annotating molecular biology entities, the social and software environments currently supporting curation and the automated methods to push it further. The next chapter will use modelling and real curation logs to explore in more detail what defines a basic curation environment.

# Chapter 3

# A model of collaborative curation

## 3.1 Introduction

This chapter describes and evaluates a quantitative model of knowledge collection and curation. The aim is to compare whether open and closed knowledge bases have different properties and whether one or the other could be more suited to molecular biology curation. The basic model, described in Section 3.2, represents the knowledge base as a set of true or false values, added and edited by users following a fixed set of rules. This Boolean model, where an annotation is represented as a true or false value is particularly suited to represent structured knowledge bases or wikis. The model developed here is only one of many possible models. However simplistic, it still makes the parameters under study explicit and highlights questions on what is an editable knowledge base (or wiki), what makes it a functioning collaboration and what role it has in a curation effort.

The second section (Section 3.3) evaluates the model against real molecular biology knowledge bases, to understand what could be the cause of some of the dynamics emerging from the individual work of independent volunteers. The evaluation builds on recent literature analysing quantitative measures of existing wikis, especially Wikipedia as detailed in the background Section 2.3. I have compared the distributions of user edits in my simulations with those of Wikipedia [Voss 05] first, and then four other open or closed knowledge bases. The model is shown to reflect real curation scenarios and it shows some of the emergent properties of wiki-like systems. The validated model is then used to try and find a quantitative reason for the particular long tail shape of authorship. The model tries to reproduce in a simple way the collection process of the real systems considered, and approximates some aggregate measures well, but I would like to stress that the model does not exclude more sophisticated causative models or dynamics.

Since the model was inspired by wiki-like systems, I called the model implementation and simulation engine *WikiSim*, for *wiki sim*ulation. The software implementation tracks the system states and all user (agent) actions during the simulation, allowing collection of aggregated measurements or potentially very detailed analysis of user actions and edit wars in the style of graphical history flows [Vieg 04]. The software has been implemented in Java. A MySQL database provides storage for the simulation results. All the code, tests and documentation is available under the University of Edinburgh GNU Public License on our Systems Biology centre wiki[1] and on the SourceForge website[2].

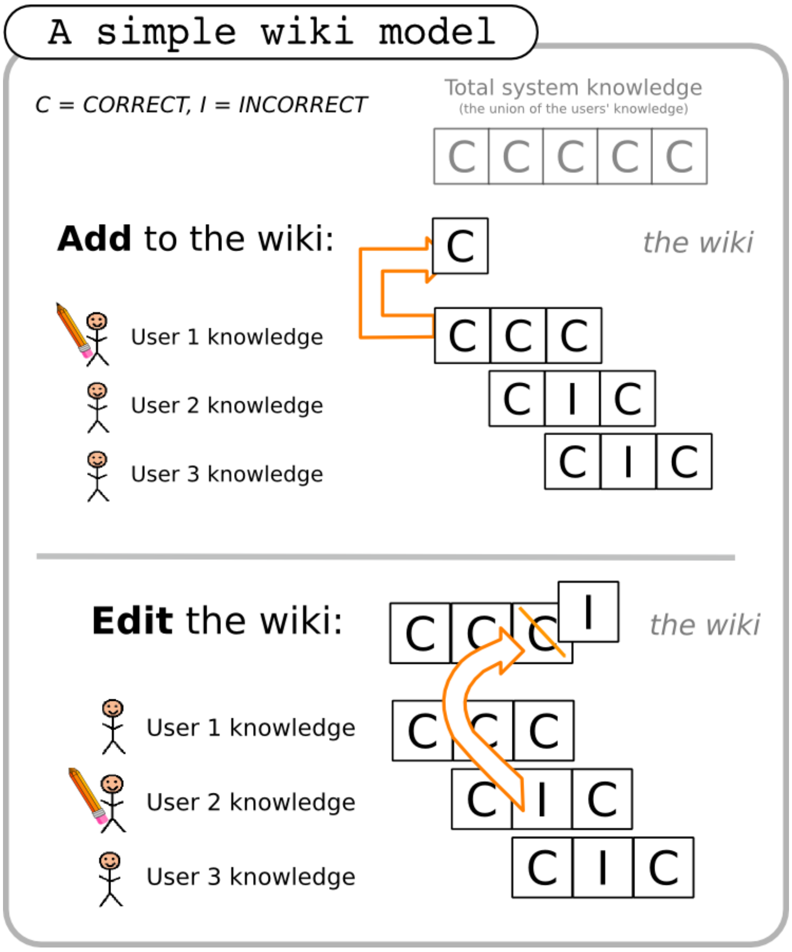


Figure 3.1: *The wiki model: adding and editing elements. In the top panel user 1 has added her first knowledge element (which is correct) to the wiki. In the bottom panel user 2 has edited the third knowledge element to reflect her internal knowledge, thereby introducing an error in the wiki.*

[1] CentreforSystemsBiologywikihttp://mook.inf.ed.ac.uk/twiki/bin/view.cgi/PublicCSB/WikiSim

[2] SourceForgehttp://sourceforge.net/projects/wikisim/

## 3.2 Model

The basic model of curation described here contains an editable knowledge base and some software agents (called users from now on) which add elements to the wiki guided by a fixed action plan sketched in Figure 3.1. The main elements of the curation model are defined below.

**Knowledge** A knowledge element $k$ is a Boolean, having either a true ($T$) value – meaning that the element annotation is correct – or a false ($F$) value – meaning the annotation is incorrect – There is a function attributing a value to each $k$, and the value can only be true or false:

$$\rho : k \rightarrow v \;\; k, \; v = \{T, F\} \tag{3.1}$$

The system knowledge $S$ is a finite set of knowledge elements. They represent the items in the world that can be annotated:

$$S = \{k_1, ..., k_n\} \tag{3.2}$$

$S$ is the pool from which the elements are taken to create the user knowledge. The universal knowledge $K$ would be the set of all the system elements, all true in value (correct annotations):

$$K = \{k_1, ..., k_n\} \; \forall k \in K : k_i = T \tag{3.3}$$

In practice, the maximum correct system knowledge achievable at any simulation round is the sum of the current user knowledge (see the section Community below on page 34).

**Users** A user $u$ belongs to the set of users $U = \{u_1, ..., u_m\}$. Each user $u_j$ has a personal set of knowledge elements $KU_j$ taken from $S$. Since it is very difficult for a single person to know everything in an extensive knowledge domain, each user knowledge is a subset of the system knowledge:

$$KU_j \subset S \tag{3.4}$$

Since user knowledge in the real world can be incorrect, the knowledge elements of the users can have true or false value (knowledge mistakes).

This, in addition to the fact that the knowledge of any two users can overlap:

$$\exists i, j \, : \, KU_j \cap KU_i \neq \emptyset, \; j \neq i \tag{3.5}$$

means that users can “disagree” on the value of knowledge elements.

**User energy** Each user has a given energy $EU_j$, to represent two real life facts: 1. users only have a limited time to spend on a wiki and 2. users enter and exit the curation community over time. Each action (search, add and edit) consumes a fixed amount of energy. When the user energy falls to zero, the user stops acting on the wiki.

**Community** The sum of the user knowledge is the community knowledge $C$:

$$C = \sum_{j=0}^{m} KU_j \tag{3.6}$$

where $m$ is the number of users. The sum here represents the result of the Boolean OR operator over the $k_i$ values (true or false) held by each user.

This is the community's *potential* maximum knowledge, if all the true values could be summed seamlessly. In practice, the curation process is the collection of actions trying to bring the users' knowledge to the knowledge base. The knowledge base is where this knowledge is collected and can be edited. Each user knowledge (and hence the community's knowledge) are set at the beginning of the simulation and cannot be changed. This is a strong assumption because biological knowledge does change over time, the change being more substantial for increasing time scales.

**Wiki** The knowledge base (or wiki) $W$ is a set of knowledge elements that the users can add or edit. The wiki is initially empty:

$$W = \{\} \tag{3.7}$$

The wiki cannot become better than the sum of all the users' correct knowledge (the community's knowledge). In the best scenario, the wiki will get closer and closer to the community's knowledge over time because the users add to and edit the knowledge therein:

$$\lim_{t \to \infty} W = C \tag{3.8}$$

**User actions** A user can add elements to the wiki or edit the elements therein, and he spends energy in doing so. An add operation adds an element $k$ to the set, it represents the addition of both a knowledge element $k$ and its value (annotation). An edit operation only changes the value of a knowledge element already in the set (from true to false or from false to true).

### 3.2.1 Community and curation

We can partition the model parameters into *community* and *curation* parameters. *Community parameters* model those aspects which depend on the user community, but not on the particular software or curation methodology used, such as: the number of users, how much the user knowledge overlaps and the distribution of incorrect knowledge in the population, potentially producing annotation mistakes. *Curation parameters* depend on the methodology or software used for curation, such as the cost of each annotation action. If we split an annotation into a "search" action and then either an "add" action if the annotation is missing, or an "edit" action if the entry exists (but the annotator wants to change its value) then a paper and pen method could be as quick as a computerised data entry method to write down one annotation, but the cost of searching through thousands of records will usually be higher on paper.

As an extreme example, if data were annotated by hand and kept on small paper cards, there might be no space available for edits, so that each edit might require recopying an entire card. In this case the cost of an edit could be as high (or even higher) than the cost of adding a new annotation. At the other end of the spectrum, in a curation software searching is fast and editing an existing entry generally takes less time than adding a completely new one, so in the model an "edit" action should cost less than an "add" action. Different curation methods can have different parameter values to represent the relative cost of each annotation action component.

### 3.2.2 Probability distributions

The distribution of knowledge among users plays a crucial role in making the model realistic. People's knowledge and favourite topics tend to be long tail distributed in many real world contexts [Ande 06]. Qualitatively, this corresponds to saying there are things everybody tends to know (or is interested in) and then a long tail of more and more specialist knowledge. Describing it in the style of Bentley *et al.* [Bent 09], users mostly copy other users' preferences while, more rarely, they "innovate" and dedicate their attention to more obscure topics.

This in turn could shape biological knowledge. For example, enzyme knowledge appears to be long tail distributed with most enzymes belonging to few classes, this could be caused by evolutionary reuse of protein domains, but it could also be in part an effect of the way humans cultivate and extend knowledge. We start by re-examining and extending the known and more rarely we tackle the relatively unknown. Contributions to Wikipedia tend to be long tail distributed too, with less than 3% of the users responsible for more than 50% of the edits [Capo 06, Voss 05]. In the model evaluation (Section 3.3) we will see that this seems to be confirmed for molecular biology knowledge bases as well.

In more detail, probability distributions (uniform or power-law, compared in Section 3.5) are used in this model to represent:

1. How users' knowledge overlaps, and hence the uniform or power law shape of the community's knowledge (compared in Section 3.3).

2. How errors in the user knowledge are distributed (*uniform*, in the absence of better esti-

mates from literature)

3. The order in which the users act on the wiki (*uniform*). However, the users are uniformly picked from the set of users *still having energy*. In case of a power-law distribution of user energy, this means that some users have more energy than others and will continue acting long after the other users' energy is spent.

4. Which knowledge element – from his or her internal knowledge – the user acts on (*uniform*, in the absence of better estimates from literature).

5. How the user energy is distributed among the population of users (*uniform or power-law*, compared in Section 3.3)

## 3.3 Model evaluation

| | ***Type*** | ***First edit*** | ***Last edit considered*** | ***Active users*** | ***Entries*** | ***Edits*** | ***Avg authors per entry*** | ***as % of tot authors*** | ***Avg entries per author*** | ***as % of tot entries*** | ***Avg edits per author*** | ***as % of tot edits*** |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **OMIM** | C | 1986 | Jan 2009 | 89 | 20,125 | 72,077 | 2.2 | 2.5% | 472 | 2.3% | 809 | 1.1% |
| **Reactome** | C | 2002 | Nov 2008 | 100 | 219,728 | 10,247 | 1.1 | 1.1% | - | - | 102 | 1.0% |
| **EcoliWiki** | O | 2005 | Feb 2009 | 139 | 56,053 | 19,666 | 1.3 | 0.9% | 48 | 0.1% | 141 | 0.7% |
| **Wiki Pathways** | O | 2007 | Feb 2009 | 420 | 10,852 | 4,192 | 1.2 | 0.3% | 5.4 | 0.05% | 10 | 0.2% |

Table 3.1: *Summary of the data and history of the knowledge bases under analysis (from creation to 2009). O=open (editing open to everyone) and C=closed system (expert curators only)*

### 3.3.1 Simulation parameters

This section describes the different model parameters that are implemented in the simulations. All the simulation parameters, such as the number of users, the user energy, the percentage of errors or the action costs can be varied at will. This basic, parametrised model can already quantify how the parameters affect knowledge quality, for example by measuring the number of correct items present in the wiki during or at the end of the simulation.

#### 3.3.1.1 Basic model parameters

In the simulations described in this chapter, the following parameters were used, unless otherwise stated:

- Number of simulation steps: 1000
- Number of users: 10
- Knowledge elements per user: 20% of the 100 system elements (either normally distributed or power-law distributed using a discrete Pareto distribution with shape parameter=2.0 and scale parameter=1.0, truncated at 1,000)
- Percentage of mistakes in the user knowledge: 20% (5% to 80% for Figure 3.2)
- User energy: power-law distributed (using a discrete Pareto distribution with shape parameter=2.0 and scale parameter=1.0, truncated at 1,000)
- Action costs: add=3 energy units, edit=2 units, search=1 unit.
- Results are averaged for 100 simulation runs.

A power law function can be written as:

$$y(x) = kx^{\alpha} \tag{3.9}$$

where $k$ is the scale parameter and $\alpha$ is the shape parameter.

**Data: open and closed molecular biology knowledge bases** An analysis is executed on 36 years of knowledge collected by 738 authors in two molecular biology wikis (EcoliWiki [McIn 12] and WikiPathways [Pico 08]) and two closed knowledge bases (OMIM [Ambe 09, Ambe 11] and Reactome [DEus 11, Crof 11]). This data is then compared with the WikiSim model simulations.

Table 3.1 summarises the age and size of the knowledge bases. The closed systems considered have had longer lives (OMIM: 23 years, Reactome: 9 years) than the open systems (EcoliWiki: 4 years, WikiPathways: 2 years), since wikis have only recently entered molecular biology. The knowledge bases range from 10,000 to 220,000 entries. An *entry* in a wiki system corresponds to a wiki article or page: in EcoliWiki an entry is an *E. coli* gene; in WikiPathways a biological pathway; in Reactome a molecule, reaction or pathway; in OMIM an entry corresponds to a human gene and its genetic diseases. An *edit* here is defined as a unique act of creation, editing or revision, marked by an author name, a time stamp and a reference to the edited entry. Wherever edits have been marked as minor or were executed by an automatic script they have been excluded from the analysis.

## 3.4 Results

### 3.4.1 User errors

The simulation results shown in Figure 3.2 exemplify the potential of the model. It shows how an increasing percentage of user mistakes affects the number of correct entries accumulated in the wiki over time. In the figure, the 100% mark represents correct values for all knowledge items assigned to the users in that simulation. The data on individual user actions gives a possible explanation of why the wiki never reaches more than 60% of correct elements. The reason is that the users spend most of their energy in edit wars over the most popular topics. The model does not provide a way for users to reach consensus (or to become annoyed and give up quarrelling).

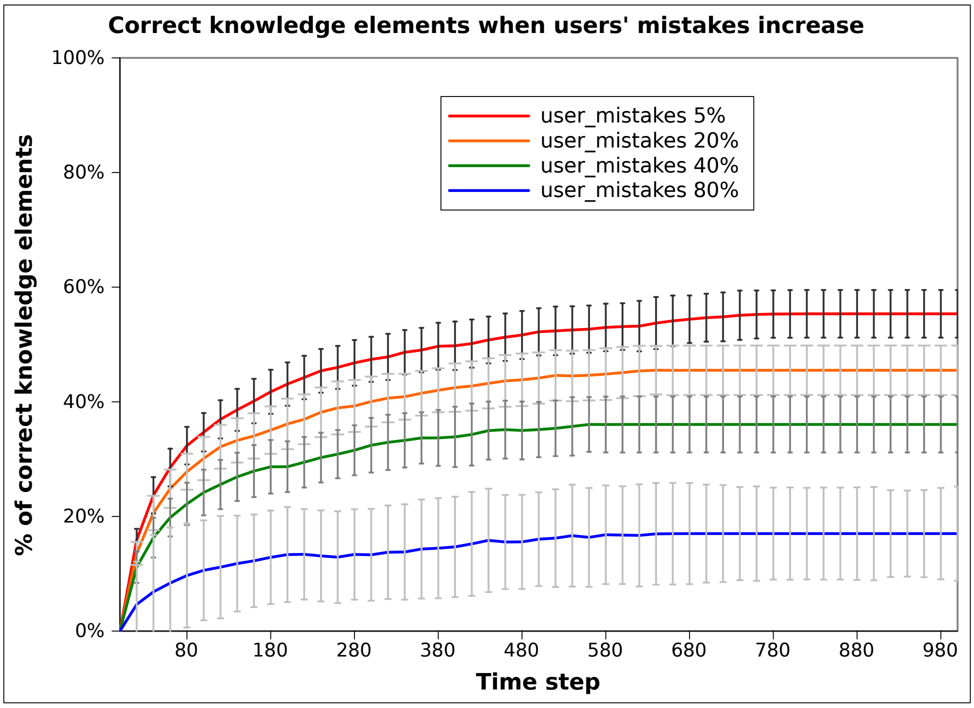


Figure 3.2: *Percent of correct knowledge elements accumulated over the course of the simulation. The different series represent increasing percentages of mistakes in the user knowledge. The vertical bars represent the standard deviation over 100 simulations.*

### 3.4.2 Authorship

This section of the model evaluation concentrates on two metrics which have been extensively discussed for wikis, Wikipedia and other collaborative systems. The first metric is the *number of edits per author*, that is, the number of knowledge base edits (text creation, change or revision) a user has authored. In Wikipedia, it has been measured that the distribution of number of edited pages per author follows a long tail distribution, with few authors contributing to a high number of pages and many authors only contributing to one or two pages (excluding automated wiki-bots and minor edits) [Voss 05, Orte 07a]. This metric is seen as a measure of how dominated a wiki is by a small core of heavy contributors.

The second metric is the *number of authors per entry*, that is, the number of authors that have edited a given knowledge base entry. In Wikipedia, the distribution of the number of authors per page also follows a long tail distribution, with few pages having a very high number of collaborating authors and many pages having only one (the creator) or two authors [Voss 05]. This could be viewed as a measure of how "collaborative" a knowledge base is or how certain topics command more attention.

**WikiSim vs Wikipedia** Figure 3.3 shows the number of authors in the model versus the number of distinct articles they have authored. The same measure is presented by Voss [Voss 05] for the 2004 German Wikipedia. Both the data and the simulation show the expected long tail of authorship common to many collaborative endeavours. The power-law shape parameter is 1.3 in the simulation versus 1.5 in Wikipedia and could be used to further refine the model parameters, if the objective were to model Wikipedia in particular.

**WikiSim vs molecular biology knowledge bases** An overview of the growth of the four knowledge bases is given in Figures 3.4 and 3.5 which show, respectively, the edits and the active authors over time (authors responsible for at least one edit in the month considered). In more detail, Figure 3.5 shows that the four systems considered have comparable numbers of active contributors, excluding a peak at about 100 authors for WikiPathways. This is quite a feat for wikis which rely on unpaid contributors and have had relatively short lives. Also, despite the different life spans, Figure 3.4 shows a rather regular increase in edits over time (that is, close to the diagonal of the plot), with the exception of OMIM, which saw little development for almost half of its lifetime before editing took off, possibly because of funding issues.

The aggregate authorship metrics – authors per entry and edits per author – which are long tail distributed in Wikipedia, are found to be long tailed also in the four knowledge bases under consideration (Figures 3.6, 3.7 and 3.8). In more detail, Figure 3.6 shows the number of authors per entry while Figures 3.7 and 3.8 show the number of edits per author for the four systems. The data for the number of *entries* (instead of edits) per author give very similar distributions

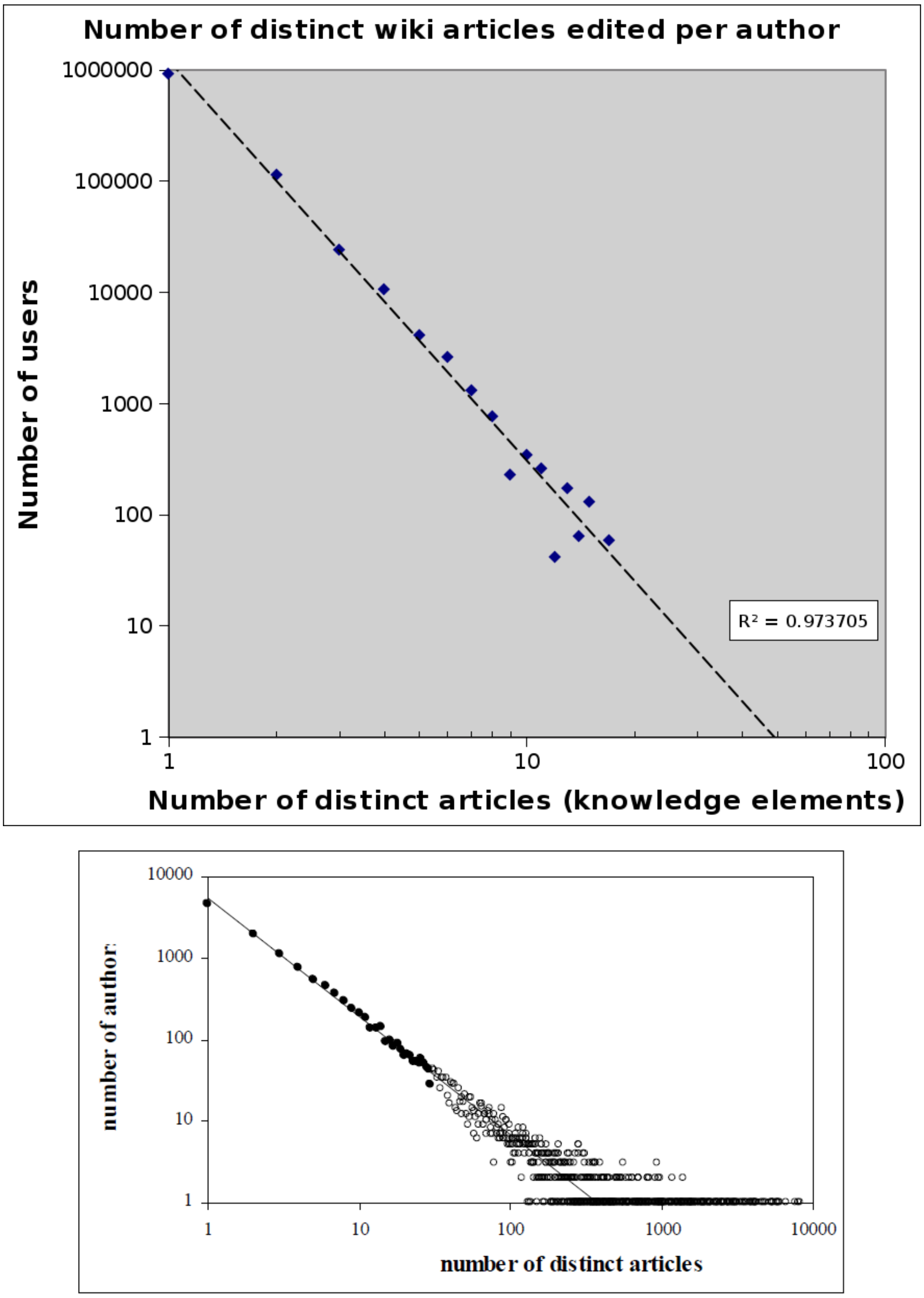


Figure 5: Number of distinct articles edited per author ($\gamma \approx 1.5$)

Figure 3.3: *Articles per author in WikiSim and Wikipedia. On the top plot: number of distinct articles edited per author from a simulation with parameters as in Section 3.3.1.1. On the bottom plot: the same measure for Wikipedia presented in Voss [Voss 05] (reproduced with permission). The power-law exponential factor is about 1.3 for the top plot, about 1.5 for the bottom plot. $R^2$ is a measure of the regression fit.*

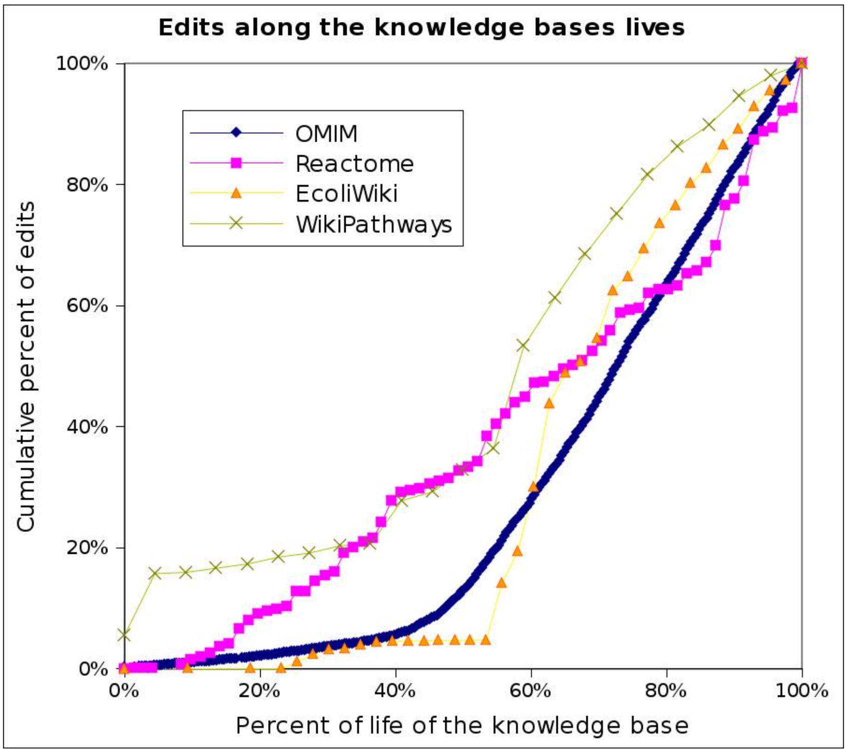


Figure 3.4: *Edits in time in the open and closed curation systems under analysis. The x axis (time) is represented as a percentage of each knowledge base life to ease the comparison. Zero represents the time of the first edit and 100% the time of the last edit done on each knowledge base in the period considered.*

(data not shown). The cumulative percentage plot in Figure 3.8 is marked with lines at 80% for the authors and 20% for the edits to compare the systems with the so called 80-20 law (or Pareto principle), which indicates that in many systems around 80% of the contributions come from the top 20% contributors. In the systems considered the distribution seems even more skewed, towards a 90-20 ratio.

These results complement the current discussion on the democracy of wiki-like systems [Orte 07b, Orte 08, Kitt 07] by showing unexpected similarities on basic metrics between open (editing open to everyone) and the closed systems (expert curators only).

This is the first time this style of analysis has been applied to non-wiki systems. The simulation results in Section 3.5 suggest that knowledge overlap among authors can drive the number of authors per entry in these systems (Figure 3.10), while the distribution of the time the users spend on the knowledge base drives the number of contributions per author (Figure 3.9).

**Is authorship data power-law distributed?** Power law distributions are common in many social, economical and biological data, with variants known in different disciplines as Pareto

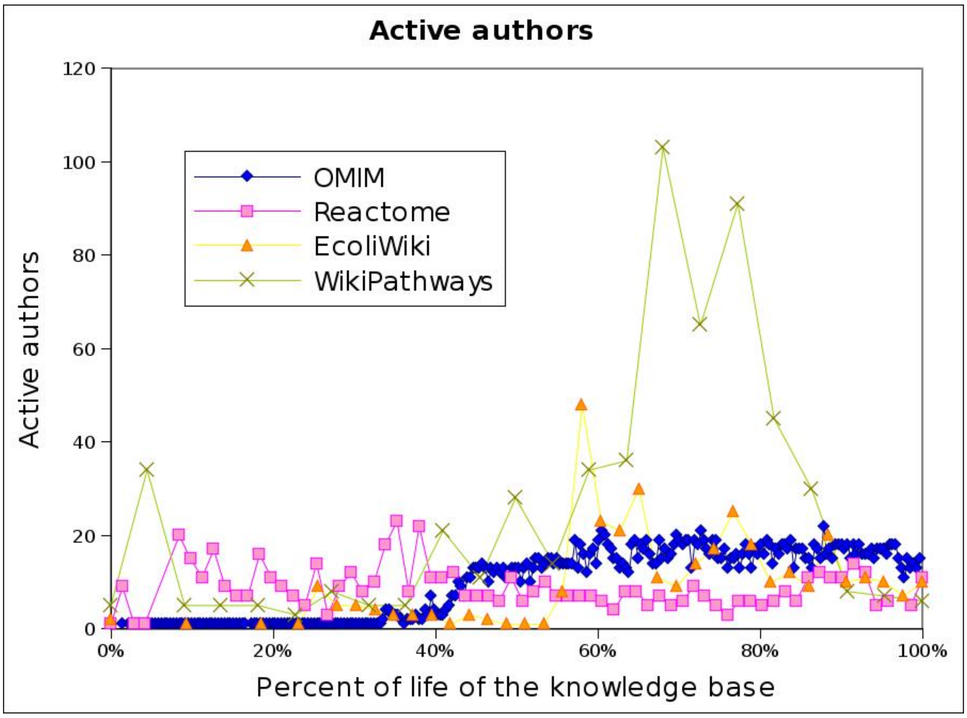


Figure 3.5: *Active authors over time in the open and closed curation systems under analysis. The x axis (time) is represented as percentage of each knowledge base total life time to ease the comparison.*

distribution, Lotka's law, Zipf's law, Bradford's law, scale-free or simply long tail [Ande 06]. When the frequency of an event varies as a power of some attribute of that event, the frequency is said to follow a power law:

$$y(x) = kx^{\alpha} \tag{3.10}$$

where $k$ is the scale parameter and $\alpha$ is a shape parameter that influences the length of the typical "tail" of these functions.

I used the method in [Clau 09] to test whether the metrics are distributed in a way consistent with a power-law distribution (Matlab scripts by Aaron Clauset) or a log-normal distribution (Matlab scripts by Ohad Gal). Tables 3.2 and 3.3 show that, in short, the authorship data could potentially fit both a power-law or a log-normal distribution.

In general, we can never completely confirm that some data comes from a power-law distribution and a low number of data points can make the analysis more difficult, especially for high $x_{min}$ (the lower data point for which the power-law distribution holds). In addition, the same causative model can generate both a power-law or a log-normal distribution depending on subtle differences [Mitz 04]. For this work the key aspect is that these metrics are either power law or log normally distributed and not, for example, linearly or normally distributed, so the data is generically defined here as long tailed.

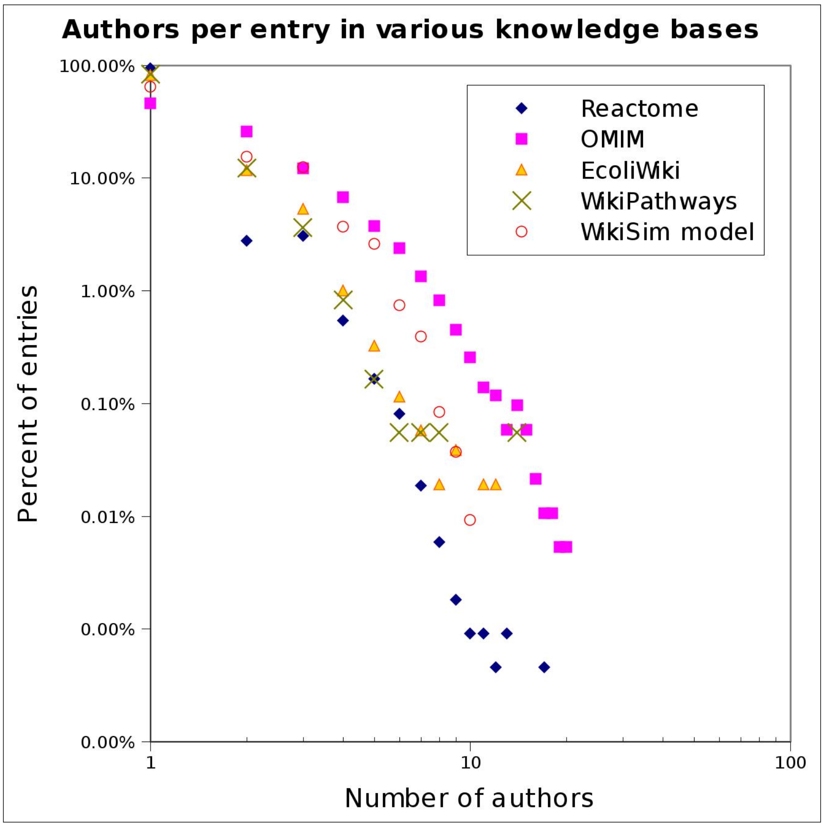


Figure 3.6: *Number of authors per entry in the open and closed curation systems under analysis (double logarithmic plot). The entries have been represented as percentage of total entries to ease comparison between knowledge bases of different sizes. The WikiSim model was initialised with the parameters in Section 3.3.1.1.*

## 3.5 Model refinement

This section describes how the use of power-law distributions in the model allows for a better fit to real data and the discrimination between potential driving forces in the authorship distributions. Different parametrisations of the model were used to compare different hypotheses regarding what causes the authoring metrics to be long tail distributed.

**What drives the long tail authorship distribution?** There has been much debate in wiki research about whether the authorship distribution supported a more democratic view of Wikipedia (written by many minor contributors) or a more closed view (Wikipedia is dominated by the edits of a few authors). The data shows that both closed and open systems seem to have a long tail distribution of edits.

The model is used here to discriminate between two hypothesis on why the authorship is skewed. The first hypothesis is that the cause of the long tail distribution is the difference in time that volunteers spend on wikis: some spend long hours, other only minutes on it. The

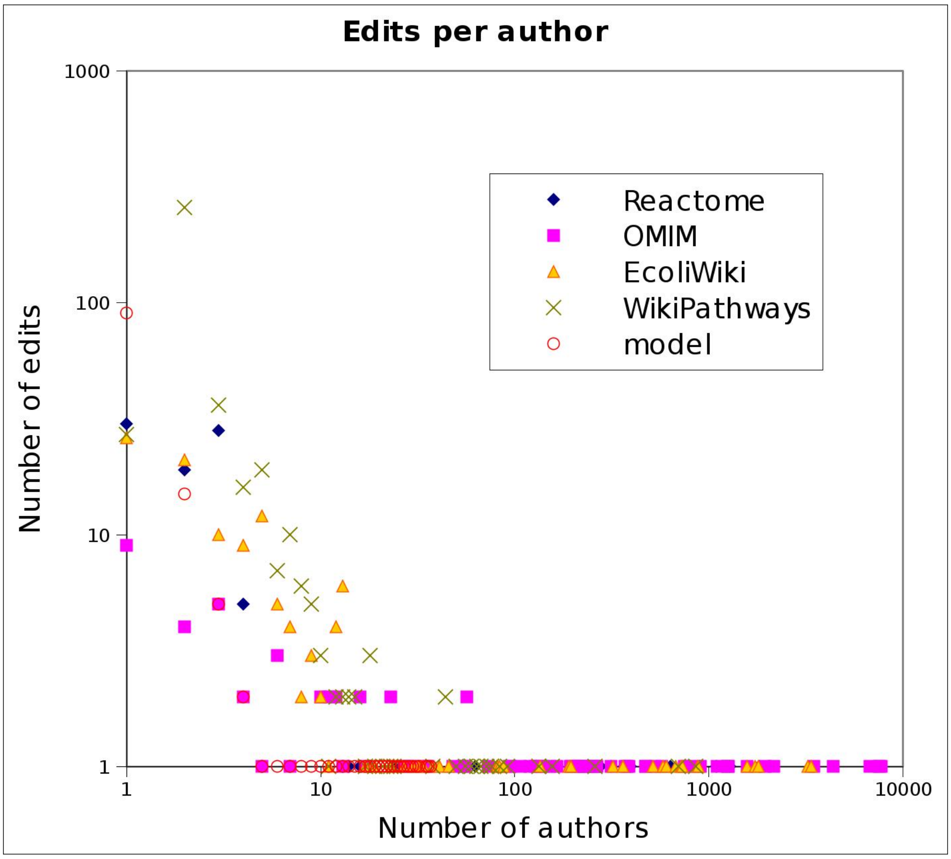


Figure 3.7: *Edits per author in the open and closed curation systems under analysis, as a double logarithmic plot. The WikiSim model was initialised with the parameters in Section 3.3.1.1.*

second hypothesis is that the cause might be the overlap of user knowledge: many users have overlapping interest in common topics, followed by a long tail of very specialist topics that fewer and fewer experts (or hobbyists) are interested in editing.

Using the model we can compare these two causative mechanisms, but other mechanisms cannot be excluded.

The comparison in Figure 3.10 shows that the long tail distribution of the *number of authors per entry* is mainly driven by a long tail distributed *user knowledge*. If the user knowledge were uniform, the authors per entry would become linear and not long tailed. In contrast, the distribution of time spent on the wiki (the user energy in my model) does not seem to affect the number of authors per entry.

So the number of authors per entry depends on how knowledge (or interest for certain knowledge topics) is distributed in the population. It could be an artefact of the way we label categories: a rare, one-gene-dependent disease will require fewer edits than a broad category such as “cancer”. This becomes less likely though when each entry corresponds to an individual protein in a compact genome (EcoliWiki). The stress researchers put on some classes is more

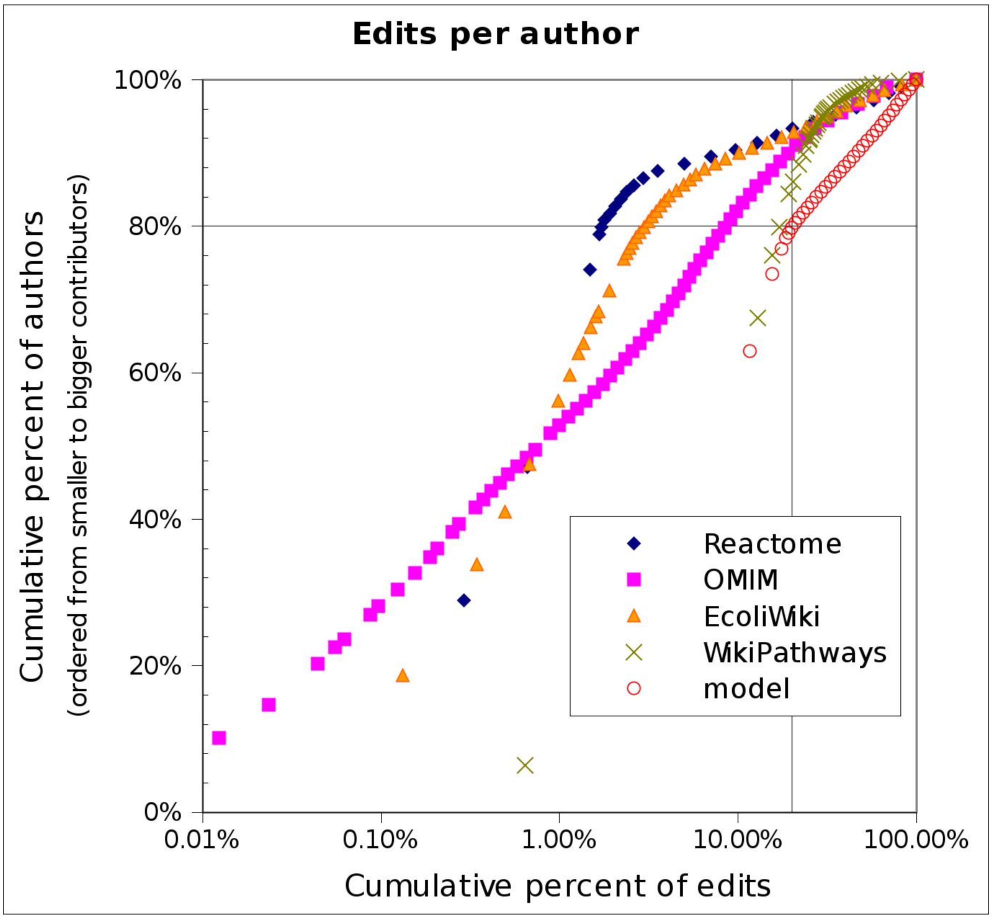


Figure 3.8: *Edits per author, as cumulative percentage, in the open and closed curation systems under analysis. The black line perpendicular to the y axis marks the 80% of authors, while the line on the x axis indicates the 20% of edits mark. A series following exactly the 80-20 Pareto principle would cross the two lines where they meet (as happens for the WikiSim model and WikiPathways). The WikiSim model was initialised with the parameters in Section 3.3.1.1.*

likely linked to applications (human or crop disease, industrial interest) or ease of study of the given protein.

The opposite is true in Figure 3.9; the long tail distribution of the *entries per author* is mainly driven by a long tail *user energy*. If the user energy is uniform, the number of entries per author become approximately Gaussian, and not long tailed. Compared with the first plot, the knowledge distribution among the users does not seem to have any impact on the entries per author.

The existence of heavy contributors seems to be a property of different engagement of user (whether paid or volunteers). Do all human populations contain a minority of more generous contributors? It could also be a property of the intersection between homogeneous user engagement and its interest for that wiki: for all we know a contributor of only one EcoliWiki

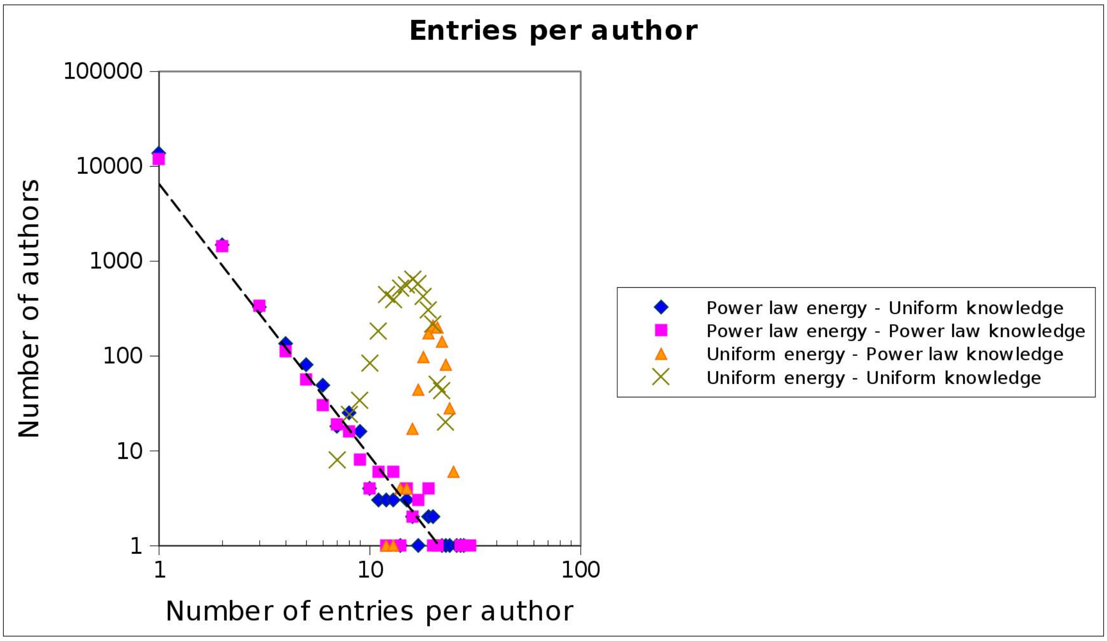


Figure 3.9: *Double logarithmic plot showing the number of entries per author depending on the user energy distribution used in the model. The WikiSim models were all initialised with the parameter values described in Section 3.3.1.1, but differ by the use of either a uniform or a power-law distribution (shape parameter* $\alpha = 2.0$*, scale parameter* $x_{min} = 2.0$*) to assign the user energy or user knowledge, as specified in the legend.*

entry may have contributed hundreds to Wikipedia (or vice versa).

In conclusion, to obtain a plausible simulation, both the user knowledge and energy have to be long tail distributed.

## 3.6 Discussion

### 3.6.1 Community vs. system knowledge

In a given WikiSim simulation, if no user has the correct value for a certain knowledge item, the maximum achievable community knowledge will be one correct element less than the system knowledge. In this case, should the knowledge base content at any given time be compared against the community or system knowledge? In other words, if no one knows better in the community, should an incorrect annotation be deemed the best achievable annotation? If the curation community comprised all human beings knowing about the topic and for a consistent span on time, this might be acceptable. However, this is rarely going to be the case or be provable. Hence, in these simulations, correctness has been compared to the system (ideal) knowledge, that is, all annotations should be correct (true).

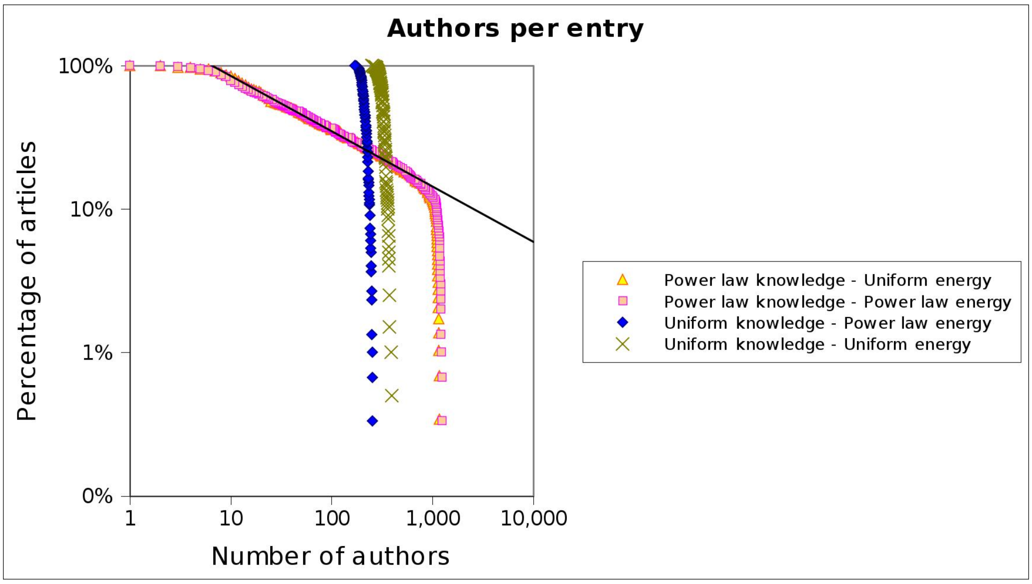


Figure 3.10: *The double logarithmic plot shows the number of authors per entry depending on which probability distribution was used to assign the system knowledge to the users. The WikiSim models were all initialised with the parameters in Section 3.3.1.1, but differ by the use of either a uniform or a power-law distribution (*$\alpha = 2.0$, $x_{min} = 2.0$*) to assign the knowledge elements per user or the user energy, as specified in the legend.*

Technically, the knowledge base can reach its maximum knowledge in the course of a simulation. In practice, this is not a realistic assumption: I know of no existing molecular biology wiki or knowledge base that has suspended activity after being declared “complete” because all possible contributors have transmitted all their knowledge. This could be the case though for certain groups of records or very limited areas of research.

### 3.6.2 Similarity of closed and open systems

Purely from the dynamics analysed (authors per entry, edits per author, edits in time and active authors in time) I could not find significant differences between open and closed knowledge bases, even if the two wikis have gained the same, if not higher, number of contributors and linearly growing edits in a relatively short time compared to the closed systems. This could mean that the differences between the two categories is not sharp: wikis can have a core of highly committed editors generating edit dynamics similar to the full time curators in closed systems, with especially visible impact in young systems such as the wikis analysed here (EcoliWiki and WikiPathways). In the same way, closed knowledge bases such as OMIM and Reactome can accept suggestions from external authors or briefly employ specialists which affect the edit dynamics in the same way as the occasional contributors in wikis.

| | ***Authors per entry*** | | | | ***Edits per author*** | | | |
|---|---|---|---|---|---|---|---|---|
| ***Power law fit*** | ***Fit*** | $\alpha$ | $x_{min}$ | ***% of data points above*** $x_{min}$ ***(number)*** | ***Fit*** | $\alpha$ | $x_{min}$ | ***% of data points above*** $x_{min}$ ***(number)*** |
| **Reactome** | 0.46 | 1.23 | 1 | 100% (13) | 0.67 | 5.62 | 1097 | 21% (5) |
| **OMIM** | 0.26 | 1.36 | 11 | 60% (12) | 0.34 | 1.62 | 151 | 56% (36) |
| **EcoliWiki** | 0.5 | 1.32 | 2 | 82% (9) | 0.82 | 2.17 | 50 | 37% (14) |
| **Wiki Pathways** | 0.63 | 1.69 | 66 | 33% (3) | 0.55 | 1.69 | 3 | 26% (10) |
| **WikiSim model** | 0.24 | 1.26 | 4 | 80% (8) | 0.13 | 1.73 | 1 | 100% (35) |

Table 3.2: *Power law fitting. A fitness >0.1 is consistent with the hypothesis that the data comes from a power-law distribution.* $\alpha$ *is the exponent (shape parameter) of the power-law distribution fitted on the data.* $x_{min}$ *is the minimum data point from which the power-law distribution holds. If more than 50% of the data points fit the power-law distribution, the results have been underlined. The WikiSim model, shown for comparison in the last row, was initialised with the parameter values described in Section 3.3.1.1, but used power-law distributions with shape parameter* $\alpha = 1.5$ *and scale parameter* $x_{min} = 2.0$ *for user energy and user knowledge.*

Another possible explanation is that the long tail distributions of authorship can be a general phenomenon caused by preferential attachment [Capo 06, Mitz 04] in the knowledge. Successful research topics receive more funding, attention and investigation, leading to more revision and editing of the corresponding entries in the knowledge bases, and more availability of experts and expert knowledge. Heavy tail distributions could also be caused by a “hidden” variable, usually linked to time, which here could be the different age of the entries [Mitz 04].

An elegant solution is given by Bentley *et al.* [Bent 09] which formulates a general model to produce long tail distributions, of which the preferential attachment is a special case. Adapting the terminology to our domain: their model adds $n$ new authors at each time step. A new author has a certain probability $(1-\mu)$ of copying the choice of past authors (that is, to edit the same wiki article or structured entry) or else, with a probability $\mu$ the author “innovates” and creates a new entry. This reflects well the wiki authorship dynamics where authors join and leave the knowledge base ranks over time. WikiSim does not explicitly model the turnover in the authors pool, but represents it by giving authors a long tail energy (so that some authors stop acting sooner) and long tail overlap of knowledge (so authors have a certain probability of “copying”

| | ***Authors per entry*** | | | | | ***Edits per author*** | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| ***Log normal fit*** | $\mu$ | $\sigma$ | ***CRB*** $\mu$ | ***CRB*** $\sigma$ | ***RMS of error*** | $\mu$ | $\sigma$ | ***CRB*** $\mu$ | ***CRB*** $\sigma$ | ***RMS of error*** |
| **Reactome** | 4.4 | 14.0 | 1.1 | 30.1 | 1.5E-08 | 5.1 | 4.6 | 0.1 | 0.7 | 3.6E-07 |
| **OMIM** | 4.0 | 7.6 | 0.4 | 5.8 | 4.0E-06 | 4.3 | 5.6 | 0.2 | 2.6 | 4.4E-05 |
| **EcoliWiki** | 2.8 | 7.7 | 0.7 | 10.9 | 2.2E-05 | 3.2 | 2.1 | 0.1 | 0.2 | 1.7E-04 |
| **Wiki Pathways** | 2.3 | 6.8 | 0.8 | 10.3 | 2.0E-04 | 0.8 | 1.7 | 0.0 | 0.2 | 6.7E-03 |
| **WikiSim model** | 4.7 | 7.4 | 0.7 | 10.9 | 5.3E-06 | 0.3 | 0.8 | 0.02 | 0.04 | 8.0E-02 |

Table 3.3: *Log-normal fitting. $\mu$ and $\sigma$ are the estimated parameters for the log-normal distribution fitted on the data. CRB is the Cramer-Rao lower bound for the parameter (ideally $\mu \geq CRB\, of\, \mu$ and $\sigma \geq CRB\, of\, \sigma$). RMS is the root mean squared error. If each CRB is lower than its corresponding parameter and the RMS of the error is lower than 0.1 the results have been underlined as probably log-normal distributed. The WikiSim model shown for comparison in the last row was initialised with the parameters in Section 3.3.1.1, but used power-law distributions with shape parameter $\alpha = 1.5$ and scale parameter $x_{min} = 2.0$ for user energy and user knowledge.*

other authors choice of the knowledge element to edit). In [Bent 09] authors would enter the system over time in shifts, but never leave. In WikiSim authors are all there at the beginning but leave the action one by one at different times. Both are approximations of real life author turnover where authors can enter and leave at different times.

### 3.6.3 Data availability

There is a wealth of open questions in the area of wiki research. This initial exploration led me to discover that, unfortunately, some of the more extensive and intensively manually curated knowledge bases of enzymatic function do not provide information regarding individual authorship. At the same time, while wikis would provide the necessary data granularity, no current wiki curates primary enzyme annotation. Enzyme Commission numbers and Gene Ontology terms are simply imported or linked from primary knowledge bases such as Swiss-Prot, KEGG or Reactome.

## 3.7 Summary

This chapter has used an initial model of manual curation to explore the distribution of edits and authorship in real molecular biology knowledge bases. The results show a surprising similarity between open and closed systems in some of the measures related to “democracy” of content. The model also demonstrates that users’ knowledge and users’ energy must be long-tail distributed to represent real knowledge bases. The next chapter will progress in the direction of augmenting such manual annotation processes with automated prediction of enzyme function. It will present a combination of data schema and multi-label machine learning algorithm that offers high accuracy and can scale to million of proteins.

# Chapter 4

# Multi-label prediction of enzyme function

Manual annotation of enzymatic functions cannot keep up with automatic genome sequencing. In this chapter I explore the use of InterPro [Hunt 09] sequence signatures to predict enzymatic function. The method described – called EnzML from now on – applies multi-label classification to enzymes represented as sets of InterPro signatures. Multi-label classification can efficiently account for proteins with multiple enzymatic functions: 50,000 such annotations exist in UniProt.

EnzML was evaluated using a standard set of 300,747 proteins for which the manually annotated Swiss-Prot [Cons 11] and KEGG [Kane 10] databases have agreeing Enzyme Commission (EC) [Bioc 99] annotations. EnzML achieved more than 98% subset accuracy (exact match of *all* correct Enzyme Commission classes of a protein) for the entire dataset and between 87 and 97% subset accuracy in re-annotating eight entire proteomes: human, mouse, rat, mouse-ear cress, fruit fly, the *S. pombe* yeast, the *E. coli* bacterium and the *M. jannaschii* archaebacterium. To understand the role played by the dataset size, I compare the cross-evaluation results of smaller datasets, either constructed at random or from specific taxonomic domains such as archaea, bacteria, fungi, invertebrates, plants and vertebrates. The results were confirmed even when the redundancy in the dataset was reduced using UniRef100, UniRef90 or UniRef50 clusters [Suze 07].

Also, multi-label machine learning is feasible in reasonable time (30 minutes to train on 300,747 instances with 10,852 attributes and 2,201 class values) using the MULAN Binary Relevance K-Nearest Neighbours algorithm implementation (BR-kNN) [Spyr 08]. InterPro signatures emerge from this exploration as a compact and powerful attribute space for the prediction of enzymatic function.

## 4.1 Methodology

### 4.1.1 Data sources

The protein sequence and Enzyme Commission (EC) [Bioc 99] annotation data was taken from UniProt Knowledge Base [Cons 11] release 2010_12 (Nov-2010) consisting of Swiss-Prot release 2010_12 and TrEMBL release 2010_12, InterPro [Hunt 09] release 30.0 (Dec 2010), KEGG [Kane 10] release 57.0 (Jan 2011). The InterPro release used contains 21,591 signatures, 21,178 of which are present in UniProt. The complete set of 5,222 EC numbers and their status (active, deleted or transferred) was downloaded from ExPASy's ENZYME database [Gast 03] (11-Jan-2011 release). All annotations using "deleted" EC numbers were removed from the data; "transferred" EC numbers were substituted with their newly assigned EC number(s). The data was further processed using Ondex [Koeh 06, Lyse 09] and MySQL. The data source content of EC and InterPro annotation is summarised in Figure 4.1.

The overlap between UniProt and KEGG is schematically represented in Figure 4.2, which shows that the manually curated section of the UniProt Knowledge Base (Swiss-Prot) only contains about half a million entries, versus the over twelve million entries awaiting manual annotation in TrEMBL. The taxonomic breakdown shows an overall dominance of bacterial annotation, in addition to a certain over-representation of vertebrates and under representation of invertebrates, considering their estimated number of species in the tree of life. This distribution is not an artefact of the intersection, it is due to the underlying distribution of Swiss-Prot and KEGG data.

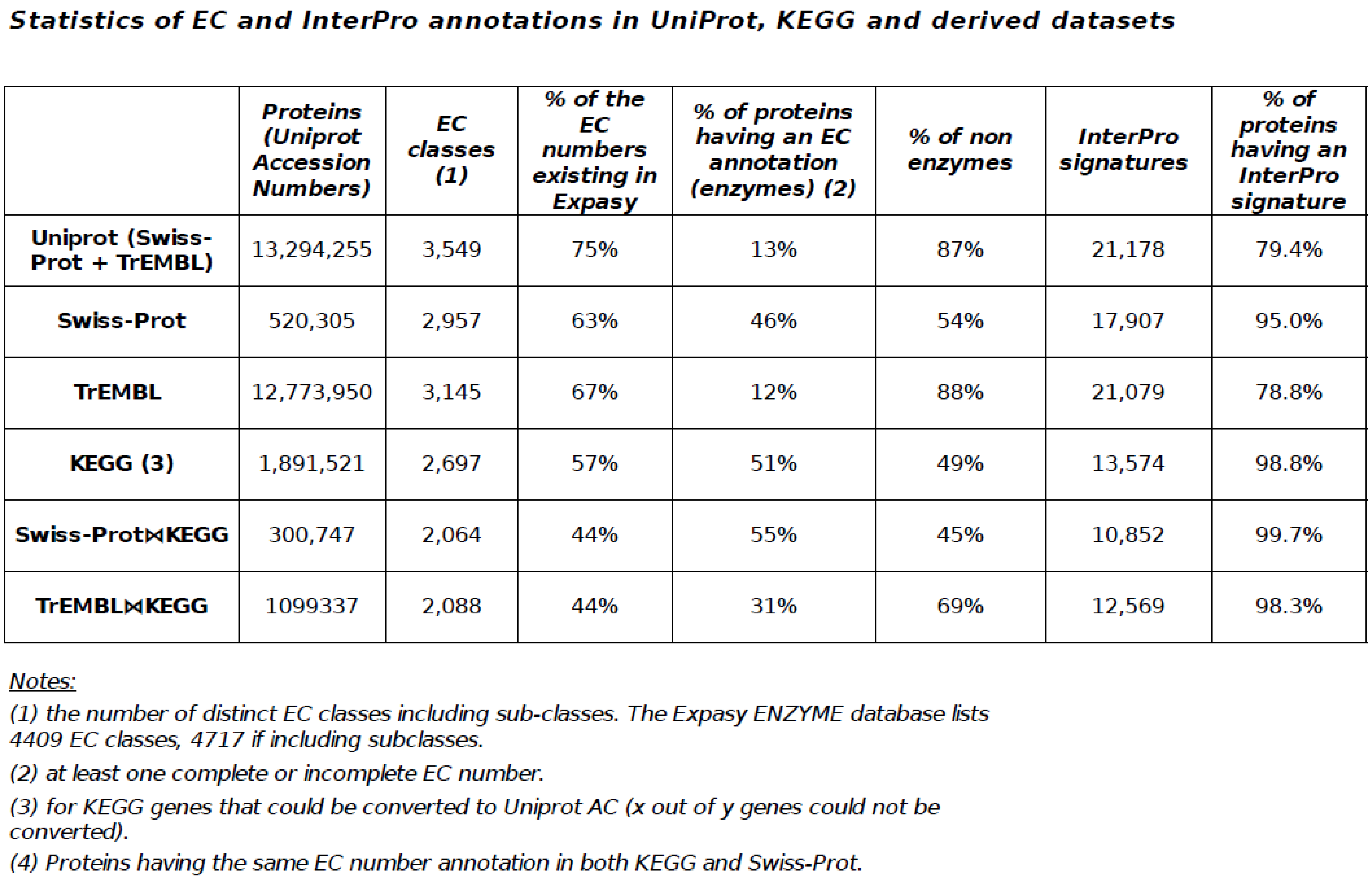
### 4.1.2 Datasets

The EnzML data schema is shown in Figure 4.3, where each instance represents a protein (identified by a UniProt Accession Number). Each protein can have zero or more class labels in the form of Enzyme Commission (EC) numbers. Each instance can also have zero or more attributes (features), each representing the presence or absence of one or more InterPro signatures (protein domains, catalytic sites, sequence repeats etc.).

In order to execute the different evaluations presented in the Results section (Section 4.2, a number of datasets have been created. The data format consists of a sparse Weka ARFF (Attribute-Relation File Format) file supplemented by a MULAN XML file containing the class label hierarchy. I made the Java code publicly available on SourceForge[1].

The *Swiss-Prot* $\bowtie$ *KEGG* dataset consists of all EC annotations *agreeing* in both Swiss-Prot and KEGG, an annotation being a couple of the form [UniProt Accession Number, EC number]. The set includes 300,747 proteins, 55% enzymes and 45% non-enzymes (see below for a definition of "non-enzyme"). The *Swiss-Prot* $\bowtie$ *KEGG* dataset has thus been submitted

[1] `https://sourceforge.net/projects/enzml/`



**Statistics of EC and InterPro annotations in UniProt, KEGG and derived datasets**

| | Proteins (Uniprot Accession Numbers) | EC classes (1) | % of the EC numbers existing in Expasy | % of proteins having an EC annotation (enzymes) (2) | % of non enzymes | InterPro signatures | % of proteins having an InterPro signature |
|---|---|---|---|---|---|---|---|
| **Uniprot (Swiss-Prot + TrEMBL)** | 13,294,255 | 3,549 | 75% | 13% | 87% | 21,178 | 79.4% |
| **Swiss-Prot** | 520,305 | 2,957 | 63% | 46% | 54% | 17,907 | 95.0% |
| **TrEMBL** | 12,773,950 | 3,145 | 67% | 12% | 88% | 21,079 | 78.8% |
| **KEGG (3)** | 1,891,521 | 2,697 | 57% | 51% | 49% | 13,574 | 98.8% |
| **Swiss-Prot⋈KEGG** | 300,747 | 2,064 | 44% | 55% | 45% | 10,852 | 99.7% |
| **TrEMBL⋈KEGG** | 1099337 | 2,088 | 44% | 31% | 69% | 12,569 | 98.3% |

*Notes:*

*(1) the number of distinct EC classes including sub-classes. The Expasy ENZYME database lists 4409 EC classes, 4717 if including subclasses.*

*(2) at least one complete or incomplete EC number.*

*(3) for KEGG genes that could be converted to Uniprot AC (x out of y genes could not be converted).*

*(4) Proteins having the same EC number annotation in both KEGG and Swiss-Prot.*

Figure 4.1: *A summary of the EC and InterPro content of UniProt, KEGG and other datasets used in this chapter.*

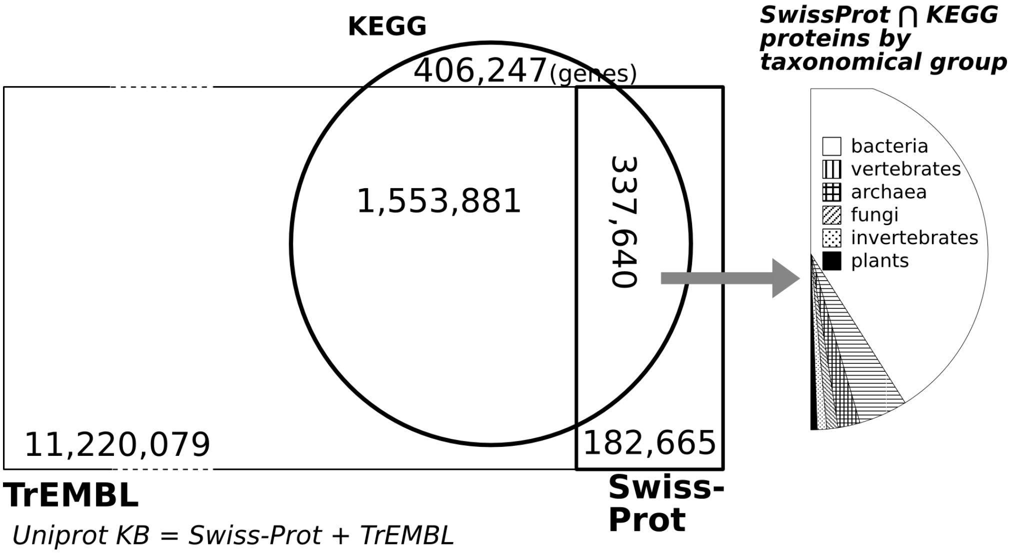


Figure 4.2: *The shared protein content of UniProt and KEGG. The circle represents KEGG, the right rectangle represents Swiss-Prot (manually curated), while the left rectangle represents TrEMBL (mostly automatically curated). The two rectangles together constitute the UniProt Knowledge Base. The intersection between Swiss-Prot and KEGG has been further expanded to show the distribution of taxonomic groups. For legibility, the areas in the pseudo Venn diagram are not exactly proportional to the number of proteins.*

to *two manual curations*, in which none of the authors were involved. The join symbol (⋈) of relational algebra has been used to represent that the set contains only annotation in agreement between the two databases.

In the same way, the *TrEMBL* ⋈ *KEGG* dataset includes all annotations agreeing between UniProt TrEMBL and KEGG. The *TrEMBL* ⋈ *KEGG* dataset is very extensive and varied, but it has not been manually curated in TrEMBL. This dataset has been included in the analysis not for the purpose of method evaluation, but to review EnzML performance on a large dataset and to judge the internal consistency of *TrEMBL* ⋈ *KEGG* itself. The protein instances have surprisingly few features, having an average of 3.55 InterPro signatures (attribute values) and 3.97 EC numbers (class labels, including incomplete EC numbers) per protein.

The proportion of proteins with no EC annotations ranges from 45% of the *Swiss-Prot* ⋈ *KEGG* dataset to 69% of the *TrEMBL* ⋈ *KEGG* dataset. These sets include proteins that have been extensively studied and do not carry enzymatic activity (especially in the *Swiss-Prot* ⋈ *KEGG* dataset) as well as proteins not yet characterised as enzymes or belonging to still unknown enzymatic classes (more probable in the *TrEMBL* ⋈ *KEGG* dataset). Due to the difficulty of distinguishing between these cases, the "non" and "not yet" EC proteins are treated

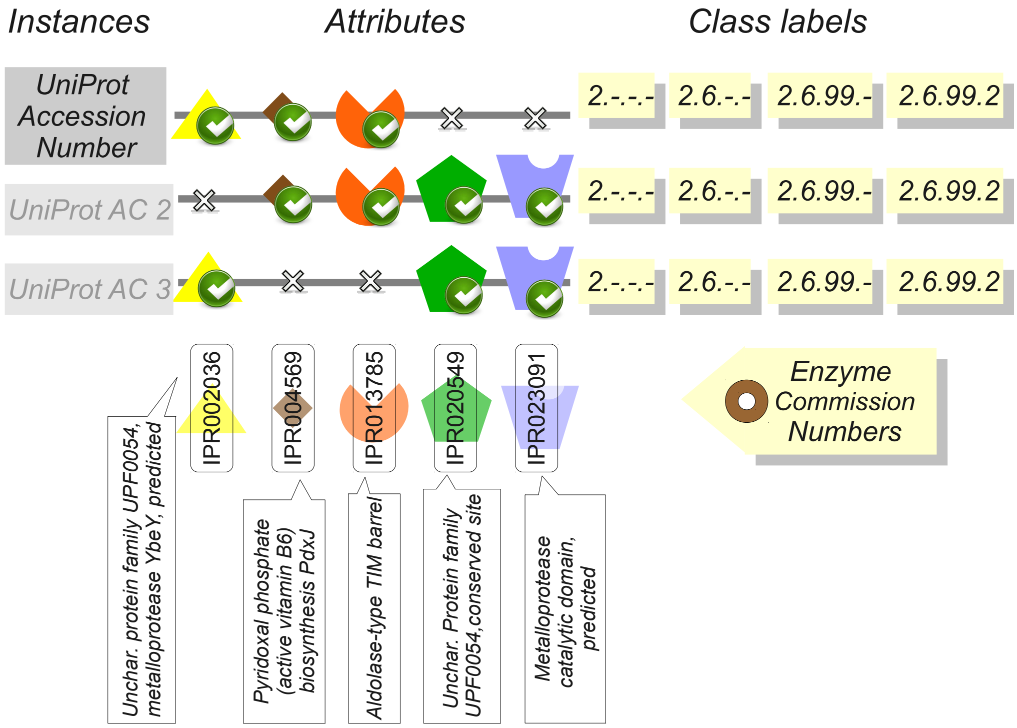

Instances
Attributes
Class labels
UniProt Accession Number
UniProt AC 2
UniProt AC 3
2.-.-.-
2.6.-.-
2.6.99.-
2.6.99.2
IPR002036
IPR004569
IPR013785
IPR020549
IPR023091
Enzyme Commission Numbers
Unchar. protein family UPF0054, metalloprotease YbeY, predicted
Pyridoxal phosphate (active vitamin B6) biosynthesis PdxJ
Aldolase-type TIM barrel
Unchar. Protein family UPF0054,conserved site
Metalloprotease catalytic domain, predicted

Figure 4.3: *Data schema: protein instances, InterPro attributes, EC numbers. In the data schema used each row represents one UniProt protein. An attribute value is the presence or absence of an InterPro signature, here shown as a geometrical shape. The class labels are one or more EC numbers, either accessible to the learning algorithm (for training) or invisible (for testing and predicting). The example shows the InterPro signatures associated with EC number 2.6.99.2 in UniProt (pyridoxine 5'-phosphate synthase, vitamin B6 pathway). These three combinations of five signatures compactly represent the 1,108 UniProt proteins having function 2.6.99.2.*

as one class. This allows EnzML to emit a cumulative “no EC” prediction as an alternative to the prediction of one or more EC numbers. A protein predicted as “no EC” could thus be either a non-enzyme or a not yet characterised enzyme or belonging to a not yet characterised enzyme class. For simplicity, I refer to this class as “non-enzyme” from now on. The EnzML method can accept instances with an empty set of attributes, which account for 0.3% of the *Swiss-Prot* ⋈ *KEGG* dataset and 1.7% of the *TrEMBL* ⋈ *KEGG* dataset. These proteins are processed normally, but they are generally predicted as “non-enzymes” due to the fact that most proteins without InterPro signatures do not have EC annotations. The datasets used also include (and hence the method predicts) incomplete EC numbers, such as EC 1.-.-.- , EC 1.2.-.- or EC 1.2.3.-.

The independence of the UniProt and KEGG curation cannot be determined by the annotations alone due to a lack of provenance meta-data. Curators in both institutions use a variety of primary (experimental data and literature) and secondary (other databases) sources to assign an EC annotation. However, out of the 1.8 million proteins annotated in both UniProt and KEGG, about 30% have a disagreeing annotation, showing that the two knowledge bases creators have different scientific opinions in many cases.

In order to evaluate the impact of the dataset size and taxonomic content on prediction, the *Swiss-Prot* $\bowtie$ *KEGG* dataset has been partitioned into taxonomic domains: archaea, bacteria and eukarya, the latter further divided into fungi, invertebrates, plants and vertebrates. For each taxonomic domain I have investigated the individual proteome having most proteins in the *Swiss-Prot* $\bowtie$ *KEGG* set: *Methanocaldococcus jannaschii* for archaea, *Escherichia coli* (all strains) for bacteria, *Schizosaccharomyces pombe* for fungi, *Drosophila melanogaster* for invertebrates, *Arabidopsys thaliana* for plants, *Homo sapiens* for vertebrates. *Mus musculus* and *Rattus norvegicus* were also included in the analysis as second and third most represented species overall (the first is *Homo sapiens*).

To examine the performance on each EC main class, the *Escherichia coli* dataset was further divided into seven datasets each containing exclusively either the no enzyme annotation (*Ecoli_NoEC*) or EC annotations starting with a different main EC class (*Ecoli_EC1*, *Ecoli_EC2*, ..., *Ecoli_EC6*).

As an alternative to machine learning, EC labels could be directly assigned from InterPro domains: the InterPro2GO file associates individual InterPro signatures with GO terms, which in turn are mapped to EC numbers in the EC2GO file. To understand if EnzML is more accurate than this simple transitive assignment, a dataset was created containing all the *Swiss-Prot* $\bowtie$ *KEGG* entries annotated using the InterPro2GO and EC2GO lists provided by the UniProt FTP website (the dataset is named *InterPro2GO2EC*).

### Sequence redundancy

To analyse the performance of EnzML at different levels of sequence similarity I generated other datasets using UniRef clusters. UniRef100 is a database of clusters of UniProt proteins that are 100% identical in sequence (UniRef90 90% similar, UniRef50 50% similar in sequence). Each cluster has a representative (reference) protein sequence and a group of other sequences similar to it. To measure the effect of sequence redundancy on the method, the *Swiss-Prot* $\bowtie$ *KEGG* dataset was reduced to only its UniRef representative sequences (UniRef100 from *Swiss-Prot* $\bowtie$ *KEGG*, UniRef90 from *Swiss-Prot* $\bowtie$ *KEGG* and UniRef50 from *Swiss-Prot* $\bowtie$ *KEGG* datasets) and cross evaluated.

### 4.1.3 EC number distribution

Enzymatic classes are long-tail distributed in the main data sources, that is, some EC numbers are very frequent among proteins while most EC numbers only rarely occur. The distribution is very skewed (Figure 4.4), with roughly a 80-10 ratio: 80% of EC numbers annotate only about 10% of UniProt enzymes, while the remaining 20% most common EC classes annotate 90% of UniProt enzymes (excluding the 45% of proteins with no EC annotation). The 2,825 most rare EC numbers (80% of the total) only annotate 185,634 enzymes (about 10% of UniProt), and 731 EC numbers have less than 5 protein examples in UniProt. 277 EC numbers only have one protein example in UniProt, which makes them non-predictable (either the unique exemplar is only in the training set or it is only in the test set).

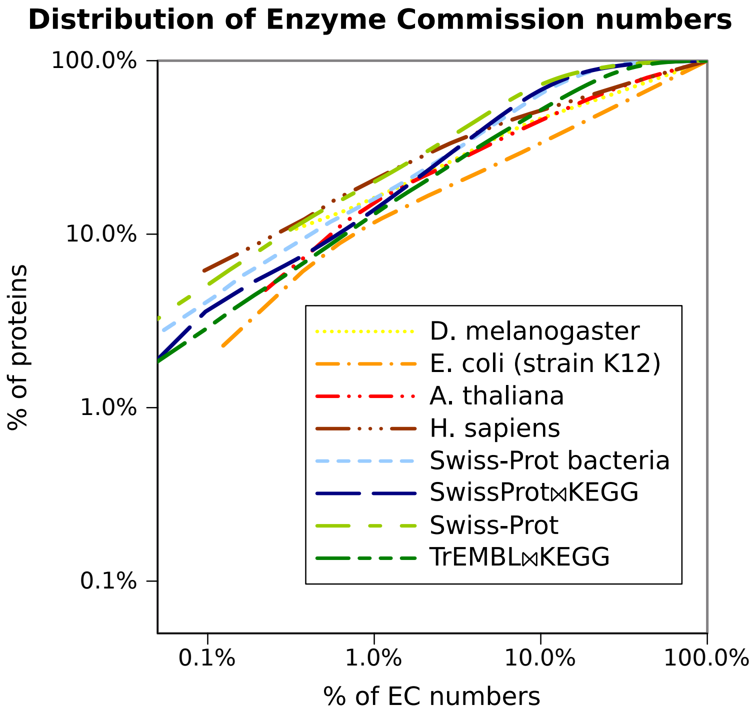


Figure 4.4: *Distribution of Enzyme Commission numbers among proteins. To compare datasets of different sizes, the distribution is represented as cumulative percentage, starting with the most frequent EC number. The bottom left EC number is the one with most proteins in the data set. The x and y axes are logarithmic. The datasets in the legend are in descending order of size. If each EC number were to annotate exactly the same proportion of proteins, the distribution would follow a diagonal from the bottom left to the top right corner of the plot.*

### 4.1.4 Algorithm

The algorithm used throughout this work is BR-kNN [Spyr 08]. BR-kNN is a multi-label adaptation of the traditional K-Nearest Neighbours using Binary Relevance. Binary Relevance transforms the original dataset into as many datasets as the existing labels, each example being labelled as label=true if the label existed in the original example and label=false otherwise (also called one-against-all or one-versus-rest approach). The MULAN version 1.2.0 implementation of BR-kNN [Spyr 08] used in EnzML makes sure the (Euclidean) distance between neighbours is calculated only once, with considerable time savings on large datasets.

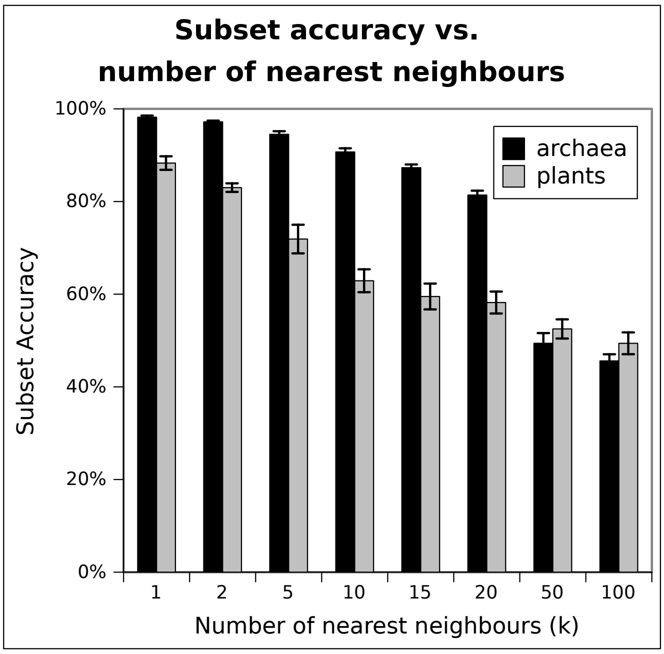


Figure 4.5: *Impact of the number of neighbours on the accuracy of the BR-kNN algorithm. Examples for the archaea and plants datasets, chosen since they are the best and worst performing small dataset by accuracy, respectively.*

The best choice for the number of neighbours was k=1 as shown in Figure 4.5. BR-kNN is fast on the data used: less than 30 minutes per fold of a 10-fold cross evaluation of 300,747 instances, on a dedicated machine with 2 GHz CPU and 4 GB RAM (14 hours to predict over a million instances). As baseline the Zero Rule algorithm was used, which assigns the majority class (non-enzyme) to every instance [2].

[2] A Zero Rule classifier observes the class attribute values and outputs the label that is most commonly found in the given dataset. It does take into consideration the non-class attributes and it will predict the mean for numeric labels or, in case of nominal labels such as EC numbers, the mode.

### 4.1.5 Evaluation metrics

The evaluation metrics are either based on a single round of evaluation (train-test) or, for cross evaluation, they are the average of a number of cross-evaluation rounds. After examining the standard deviations, the datasets with less than 40,000 proteins were submitted to two rounds of 10-fold cross evaluation, training on 9/10 of the data and testing on the remaining unseen 1/10 (one round of cross evaluation for bigger samples). Among the performance metrics presented, the average value of *subset accuracy* is particularly significant: it is a strict measure of prediction success, as it requires the predicted set of class labels to be an *exact match* of the true set of labels [Tsou 07]. For example, if a protein has these four EC class labels: [EC 1.-.-.-, EC 1.2.-.-, EC 1.2.3.- and EC 1.2.3.4], and it is assigned as prediction only the three first labels: [EC 1.-.-.-, EC 1.2.-.-, EC 1.2.3.-], this prediction would be considered as *completely* incorrect, because it misses the last label.

Where computable, also *micro* and *macro* metrics are reported. In this context *micro* averaging (averaging over the entire confusion matrix) favours more frequent EC numbers, while *macro* averaging gives equal relevance to both rare and frequent EC numbers. Hence a protein will affect the macro-averaged metrics more if it belongs to a rare EC class. *Example based* metrics consider how many correct EC predictions have been given to each individual protein example. The full mathematical form of all metrics is defined by Tsoumakas *et al.* [Tsou 10, Tsou 07]. The best achievable value of all these measures is 100% when all instances are correctly classified. Where averaged, the metrics are presented with plus and minus standard deviation marks.

### 4.1.6 Statistical significance

To judge the difference between sets of results, the p-value at 5% confidence was used and calculated as follows. If the t-statistic is:

$$t = \frac{\frac{X-M}{sd}}{\sqrt{n}}$$

where $X$ is the average (and $sd$ the standard deviation) of the reference set of samples, $M$ is the average of the other set of samples to be compared and $n$ is the number of samples in both sets, the p-value becomes:

$$p - value = tdist(abs(t), r, tails)$$

where $r$ are the degrees of freedom (equal to $n-1$). Here a two tailed hypothesis is considered, so $tails$ equals 2. $tdist$ returns the probability density function for the t-distribution, calculating:

$$\frac{\Gamma((r+1)/2)}{\sqrt{\pi r}\Gamma(r/2)}\left(1+\frac{t^2}{r}\right)^{\frac{-r+1}{2}}$$

where $\Gamma$ is the Gamma function and $r$ are the degrees of freedom. If the $p-value$ is lower than 5%, the confidence that the samples come from different underlying distribution is higher than 95% and hence the two samples are declared significantly different.

## 4.2 Results

### 4.2.1 Whole, taxonomic and random datasets

The first set of experiments address the ability of EnzML to predict EC numbers using InterPro signatures by cross evaluation. The cross-evaluation results are summarised in Figure 4.6 (additional metrics in Figure 4.11 on page 67). The total dataset *Swiss-Prot* ⋈ *KEGG* achieves 98% (±0.1% standard deviation) subset accuracy (perfect match of all enzymatic classes of a protein). For comparison, the Zero Rule algorithm achieves 45% ±0.2% subset accuracy.

To understand whether taxonomically related proteins were better at predicting proteins in their own taxa, the *Swiss-Prot* ⋈ *KEGG* dataset has been subdivided into archaea, bacteria and eukarya (further divided into fungi, invertebrates, plants or vertebrates). The average classification accuracy after cross evaluation of each taxonomic dataset was then compared with sets of the same size as each taxonomic set, but comprising proteins picked at random from *Swiss-Prot* ⋈ *KEGG*.

The results in Figure 4.6 show that the prediction accuracy generally increases as the dataset size increases. Excluding distantly related species does not seem to dramatically improve results: only the archaea and bacteria sets significantly outperform a random set of the same size, but they cover a reduced set of enzymatic functions compared to the full set. The plants, invertebrates, fungi and vertebrates sets are not significantly different from a random set of the same size, while the eukarya's dataset accuracy is significantly different but lower.

### 4.2.2 Sequence redundancy reduction

To evaluate the impact of the sequence redundancy reduction on the method, a cross evaluation was executed on the three sets of proteins derived from *Swiss-Prot* ⋈ *KEGG* by keeping only the UniRef reference entries (*Swiss-Prot* ⋈ *KEGG* from UniRef100, *Swiss-Prot* ⋈ *KEGG* from UniRef90 and *Swiss-Prot* ⋈ *KEGG* from UniRef50). Hence the *Swiss-Prot* ⋈ *KEGG* UniRef50 dataset contains only one representative sequence per each 50% similarity cluster. When the dataset is submitted to 10-fold cross evaluation, the nine tenths of sequences that make up the training set are all less than 50% similar to the sequences in the test set (the remaining 10th).

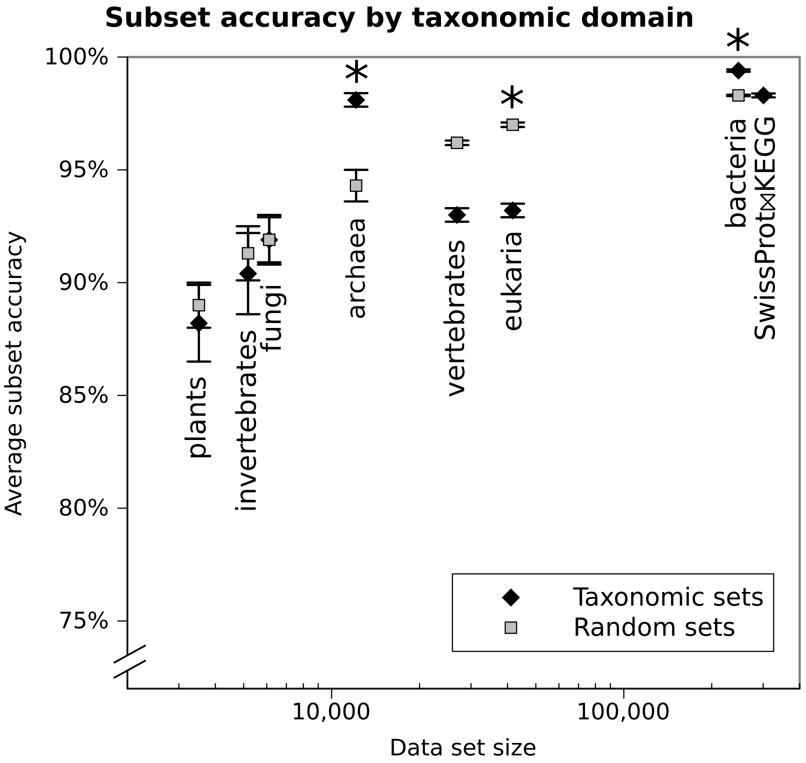


Figure 4.6: *Cross evaluation over taxonomic and random datasets. The plot compares the subset accuracy between taxonomic datasets and random sets of the same size. The rightmost point of the diagram is the whole Swiss-Prot ⋈ KEGG dataset. The y axis (accuracy and recall) starts at 70%. An asterisk indicates significant difference in accuracy (with p-value at 5%) between the taxonomic and random datasets below.*

The cross-evaluation results of the three sets of reference proteins (UniRef100, UniRef90 and UniRef50) derived from *Swiss-Prot* ⋈ *KEGG* are shown in Figure 4.7. The results are robust and *not significantly affected* by the reduction to UniRef100 sequences, not even when clustering at 50% sequence similarity, despite losing 80% of the sequences. This might be because, in spite of the dramatic reduction in the number of sequences in the set, only 4% of the EC numbers and 3% of the InterPro signatures are lost, as shown in Figure 4.8.

### 4.2.3 Proteome re-annotation

The performance obtained by cross evaluating the entire *Swiss-Prot* ⋈ *KEGG* dataset is representative of the success that can be expected on a metagenomic sample, especially one with a high bacterial content, as suggested by the high bacterial content in Figure 4.2. I hence executed another set of experiments to evaluate the performance of EnzML on annotating *individual* proteomes. Each experiment: 1. excluded the chosen species from the *Swiss-Prot* ⋈ *KEGG* dataset, 2. trained on the remaining data, 3. re-annotated that species' proteome (as if it were

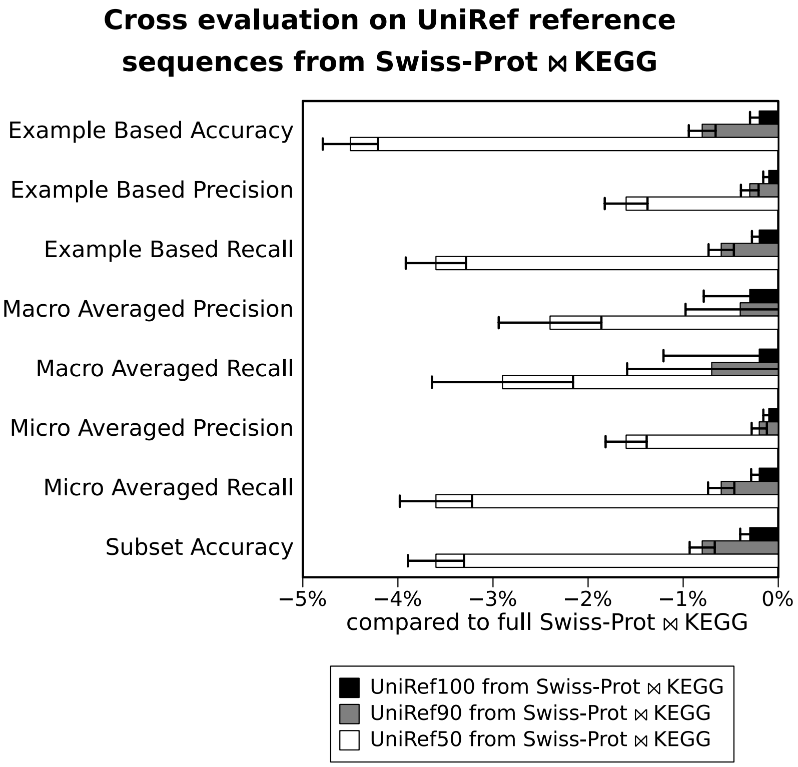


Figure 4.7: *Cross evaluation on UniRef reference sequences. The reference sequences are derived from Swiss-Prot ⋈ KEGG using UniRef100, UniRef90 or UniRef50 clusters. The values are shown as difference to the corresponding value for the entire Swiss-Prot ⋈ KEGG dataset.*

from a newly sequenced genome), and 4. compared the predictions with the existing annotations. This is sometimes referred to as jackknife evaluation. Figure 4.9 shows that EnzML can re-annotate an entire proteome with subset accuracy starting at 87% for *A. thaliana* and reaching 97% for *E. coli*.

To gauge the predictive power of a single species, the inverse was also attempted: to re-annotate the entire *Swiss-Prot ⋈ KEGG* dataset based on a single proteome. This inverse exercise (Figure 4.10) shows that up to 88% of proteins, and more than a third of the EC numbers, can be re-annotated correctly in the *Swiss-Prot ⋈ KEGG* dataset (minus *E. coli*) if the training occurs on possibly the most studied species in molecular biology, *E. coli*. This suggests a high level of evolutionary conservation of core metabolism across species.

### 4.2.4 Comparison with InterPro2GO2EC and TrEMBL

EC labels could also be directly assigned from InterPro domains using the InterPro2GO and EC2GO lists. As shown in Figure 4.11, this method has much lower accuracy (80%) than EnzML (97%) on the same *Swiss-Prot ⋈ KEGG* dataset.

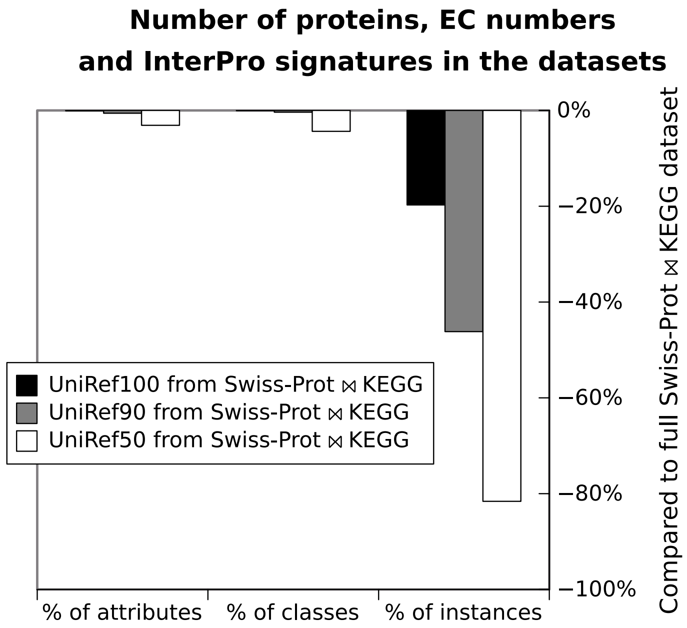


Figure 4.8: *UniRef datasets statistics. Reduction in the number of protein instances, InterPro attributes and EC numbers when the Swiss-Prot ⋈ KEGG dataset is reduced to its UniRef representative sequences. The values are shown as difference to the corresponding value for the entire Swiss-Prot ⋈ KEGG dataset.*

### 4.2.5 Testing on an independent dataset

To assess computational performance, EnzML was also trained on *Swiss-Prot ⋈ KEGG* (the right semicircle in Figure 4.2 on page 54 on page 54) and tested on the diverse and extensive (1,099,321 proteins), but not intensively manually curated, *TrEMBL ⋈ KEGG* dataset (the left semicircle in Figure 4.2 on page 54). Figure 4.11 compares the performance on the *TrEMBL ⋈ KEGG* dataset with the cross-evaluation performance on *Swiss-Prot ⋈ KEGG* and the cross evaluation on the whole of Swiss-Prot.

The loss of subset accuracy on the *TrEMBL ⋈ KEGG* dataset is not due to a limitation in EnzML, but more to the sheer variety and low internal consistency of *TrEMBL ⋈ KEGG*. The loss of accuracy on the *TrEMBL ⋈ KEGG* set cannot be accounted for by loss of rare EC numbers: classes existing in *TrEMBL ⋈ KEGG* but not in *Swiss-Prot ⋈ KEGG* only affect about 7,600 proteins out of over a million. However, *Swiss-Prot ⋈ KEGG* only contains half of the InterPro domains existing in *TrEMBL ⋈ KEGG* (see Figure 4.1). InterPro domains not present in the training set (*Swiss-Prot ⋈ KEGG*) cannot help in predicting proteins in the test set (*TrEMBL ⋈ KEGG*). This affects about 76,000 *TrEMBL ⋈ KEGG* proteins and could account for the reduction of subset accuracy from 98.3% (*Swiss-Prot ⋈ KEGG* cross evaluation) to 90.7% (train on *Swiss-Prot ⋈ KEGG* and test on *TrEMBL ⋈ KEGG*).

To further investigate the quality of the predictions emitted for *TrEMBL ⋈ KEGG* I com-

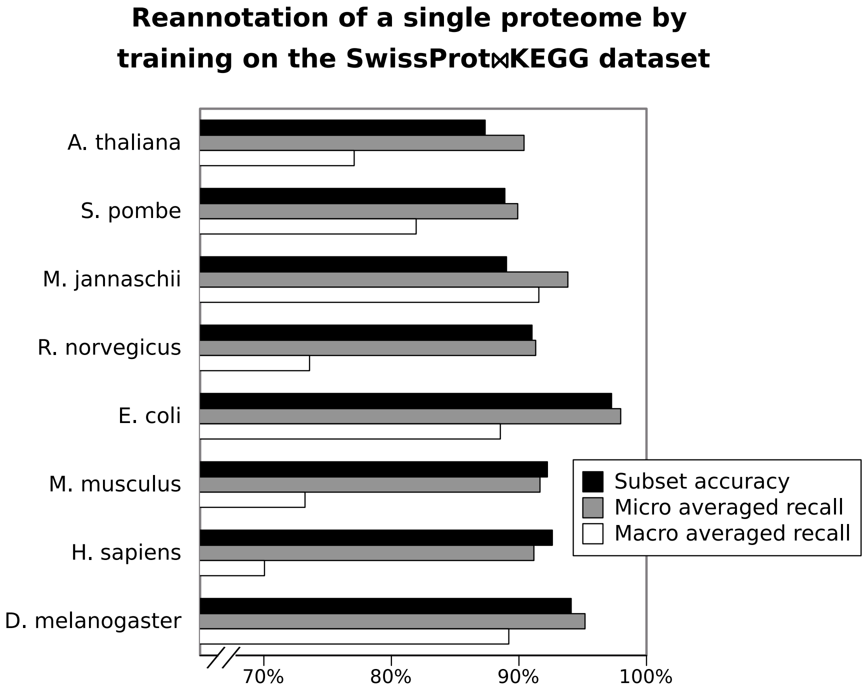


Figure 4.9: *Reannotation of individual species' proteomes. The classifier is trained on the Swiss-Prot ⋈ KEGG dataset (minus the species to be predicted) and then used to predict each species' proteome. The x axis (accuracy and recall) starts at 65%. There are no standard deviation bars since no randomisation is involved: each value represents one experiment (one species excluded from training and used as the test set).*

pare the number of EC digits of the predictions with the number of digits in the correct EC number annotations (Figure 4.12). As desirable, the proportion of predicted four digit EC numbers appears to be in line with their proportion in the true dataset. The higher the number of digits, the more specific the prediction, for example: EC 1.-.-.- only provides a generic enzymatic classification (oxidoreductases), while EC 1.2.3.4 defines the catalytic functionality down to the substrate. The third EC number digit (sub-subclass) defines the type of substrates (with oxygen as acceptor). The fourth EC number digit specifies the exact substrate as oxalate.

### 4.2.6 Prediction errors

A more detailed analysis of the prediction errors was executed using the *E. coli* dataset. The predictions were obtained by training on the *Swiss-Prot* ⋈ *KEGG* dataset minus all *E. coli* strains (286,938 instances) and predicting all strains of *E. coli* (13,800 instances). Figure 4.13 represents the predictive accuracy by main EC class. The highest accuracy is achieved for classes EC 6 and EC 2, while the lowest accuracy is recorded for classes EC 1 and EC 4. Thus the errors are not homogeneously distributed among EC numbers. Classes EC 5 and 6 (and no

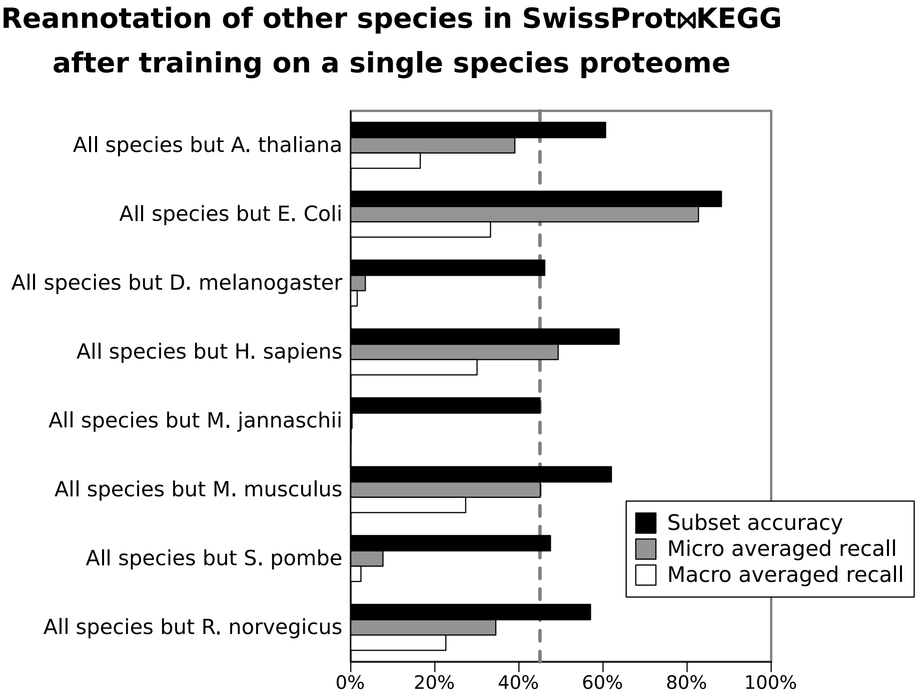


Figure 4.10: *Re-annotation of the entire Swiss-Prot ⋈ KEGG dataset training on individual species' proteome. The classifier is trained on a single proteome and then used to predict all the other species. The dashed line at 45% of accuracy represents the baseline of subset accuracy than would be obtained if all proteins were simply classified as non-enzyme. There are no standard deviation bars since no randomisation is involved: each value represents one experiment (one species is used for training and all other species are used as the test set).*

EC) are not affected by a high number of errors despite being the most frequent, while classes EC 1 to EC 4 are more affected.

Table 4.1 shows the most common mistakes by main destination EC (wrong EC) and source EC (true EC). Out of the predictions of the 13,800 *E. coli* proteins, the most common mistake is the attribution of a class EC 3 to 85 proteins that are in fact non-enzymes. When all four EC digits are considered, the most common error for the classifier is the classification as non-enzymes of sixteen EC 3.6.3.33 enzymes (Vitamin B12-transporting ATPase). The second most common error was to classify as EC 2.5.1.18 eleven proteins that are in fact non-enzymes in Swiss-Prot and KEGG. However, the vast majority of errors is spread across a variety of EC numbers. There is no higher incidence of mistakes in less frequent classes.

### 4.2.7 Comparison with other EC prediction methods

This sections presents a quantitative comparison with some well known methods to predict enzymatic function: PRIAM [Clau 03], ModEnzA [Desa 11] and EFICAz [Tian 04, Arak 09].

| *Wrong EC* | *True EC* | *Errors* | *% of errors* |
|---|---|---|---|
| EC 3 | no EC | 85 | 24% |
| EC 2 | no EC | 78 | 22% |
| no EC | EC 3 | 52 | 14% |
| EC 1 | no EC | 42 | 12% |
| EC 4 | EC 5 | 16 | 4% |
| no EC | EC 1 | 13 | 4% |
| EC 6 | EC 2 | 12 | 3% |
| EC 4 | no EC | 12 | 3% |
| EC 4 | EC 2 | 11 | 3% |
| EC 2 | EC 3 | 7 | 2% |
| no EC | EC 2 | 6 | 2% |
| EC 3 | EC 5 | 4 | 1% |
| EC 5 | no EC | 4 | 1% |
| EC 4 | EC 3 | 4 | 1% |
| EC 2 | EC 4 | 4 | 1% |
| EC 3 | EC 4 | 4 | 1% |
| EC 1 | EC 5 | 3 | 1% |
| no EC | EC 6 | 2 | 0.6% |

Table 4.1: *Most frequent errors in the classification of* E. coli *proteins by their main EC class (true and predicted).*

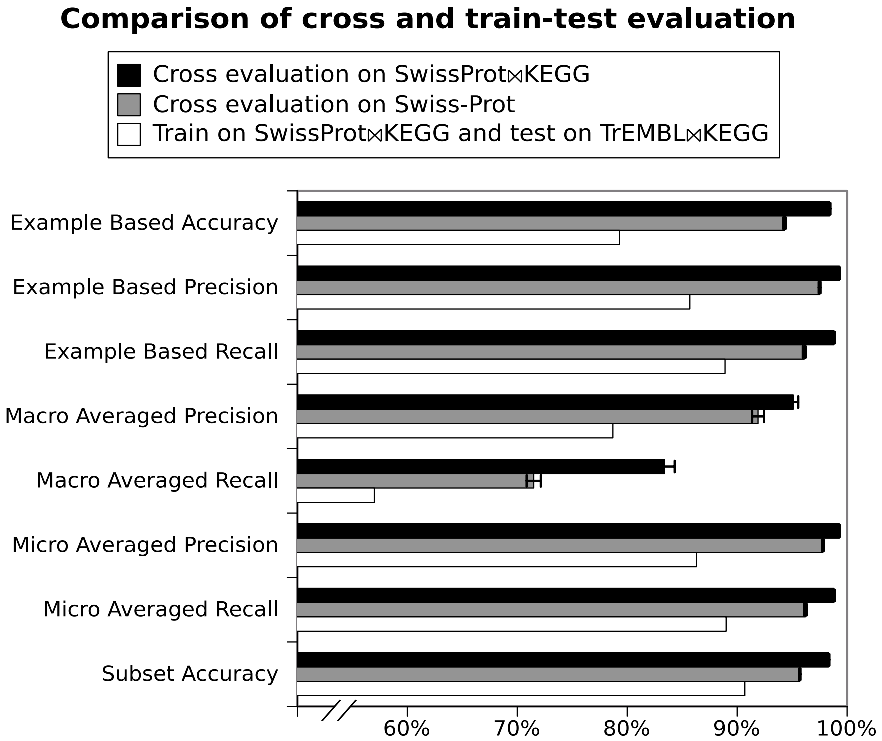


Figure 4.11: *The results of the internal cross evaluation of the entire Swiss-Prot ⋈ KEGG and Swiss-Prot datasets are compared with the direct transitive annotation using InterPro2GO and GO2EC lists. The results of training on the Swiss-Prot ⋈ KEGG dataset and testing on the TrEMBL ⋈ KEGG dataset are also included. The x axis (accuracy, precision, recall) starts at 40%.*

#### 4.2.7.1 PRIAM

PRIAM [Clau 03] was designed to predict the overall metabolism of an organism, indicating whether particular enzyme functionalities were encoded in the genome, rather than assigning functions to individual genes. A gene-oriented version of PRIAM was introduced in 2006 to address this task. In contrast, EnzML is designed to associate EC numbers with individual genes or gene fragments.

Table 4.2 compares the performance of PRIAM and EnzML in recognising EC numbers in a dataset. EnzML improves on PRIAM results on both recall (sensitivity) and true negative rate (specificity), for all genomes but *E. coli*, where specificity is higher, but recall is the same as PRIAM and lower than the KEGG Orthology.

**ModEnzA and EFICAz** EnzML improves on ModEnzA [Desa 11] by supporting the prediction of multiple EC numbers for a protein, and on EFICAz [Tian 04] by being able to assign multiple EC numbers of any number of digit. EFICAz2 [Arak 09] improves the precision of EFICAz on test sequences having less than 30% similarity to the training set, and has not been

| | **KEGG Orthology (from [Clau 03])** | | **PRIAM jackknife (from [Clau 03])** | | **EnzML jackknife** | | |
|---|---|---|---|---|---|---|---|
| *Genome* | *Specificity* | *Sensitivity* | *Specificity* | *Sensitivity* | *Macro-Averaged Specificity* | *Macro-Averaged Recall (Sensitivity)* | *Macro-Averaged Precision* |
| *B. aphidicola* | 87% | 80% | 86% | 91% | 99% | **98%** | 98% |
| *E. coli* | 89% | **91%** | 92% | 88% | 99% | 88% | 93% |
| *H. influenzae* | 88% | 93% | 84% | 91% | 99% | **95%** | 97% |
| *M. genitalium* | 93% | 95% | 86% | 87% | 99% | **96%** | 94% |
| *M. pneumoniae* | 91% | 95% | 85% | 87% | 99% | **96%** | 93% |

Table 4.2: *Comparison between PRIAM, KEGG Orthology and EnzML. The data for columns 1-5 (PRIAM jackknife and KEGG Orthology) are taken from [Clau 03], where the original caption reads: "Specificity and sensitivity of PRIAM-based enzyme detection in five complete genomes, using SWISS-PROT annotation as a standard. The RPS-BLAST E-value was set at 10–30. Jackknife analysis was performed with PRIAM profiles in which sequences from the corresponding genome were omitted. Specificity and sensitivity of KEGG Orthology assignments (retrieved from http://www.genome.ad.jp/kegg/kegg2.html; 10,25) were calculated similarly against SWISS-PROT for comparison." Columns 6-7 contain EnzML results from leave-one-proteome-out experiments (jackknife) as detailed in Table 4.9 with the addition of Haemophilus influenzae and Mycoplasma genitalium data. The highest specificity value for the row is underlined. The highest sensitivity value is in bold.*

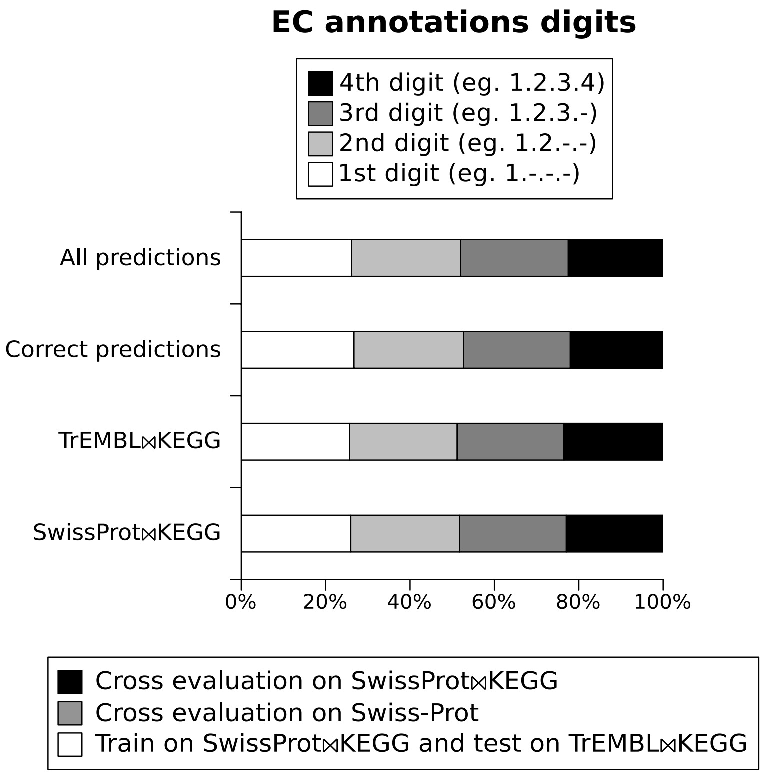


Figure 4.12: *Comparison of the EC digits in the predicted and actual EC numbers for the TrEMBL ⋈ KEGG dataset. All predictions = all the EC annotations emitted by training on Swiss-Prot ⋈ KEGG and predicting the unlabelled TrEMBL ⋈ KEGG (true positives, true negatives, false positives, false negatives). Correct predictions = only the predictions corresponding to true, correct annotations existing in TrEMBL ⋈ KEGG (true positives and true negatives). Wrong predictions = false positives and false negatives.*

evaluated separately from EFICAz.

Tables 4.3 (continued in Tables 4.4 and 4.5) compare the recall (sensitivity) and true negative rate (specificity) of various methods in predicting EC numbers and assigning EC numbers to sequences. The prediction performance published by Desai *et al.* [Desa 11] for one eukaryotic and three bacterial genomes are compared to the corresponding EnzML results.

EnzML shows higher recall (sensitivity) for all genomes presented, the only exception being the recall of *EC numbers* (EC number sensitivity row) in *E. coli* where ModEnzA Tier I+II = 91.1% while EnzML is only 88.5%. However, for the same species, the recall/sensitivity of *E. coli* sequences has EFICAz=86% and ModEnzA=89-92% while EnzML has a much higher 97%. EnzML also exhibits a very high true negative rate (specificity). For *E. coli* sequences: EFICAz 81%, ModEnzA 85-87%, EnzML 99.9%, in addition to high overall precision (98%) and accuracy (97%). In addition, despite acting on multi-label data, ModEnzA and EFICAz use metrics appropriate for binary class data, further complicating the comparison [Soko 09].

| ***Methods*** | ***EFICAz (from [Desa 11])*** | ***ModEnzA (Tier I) (from [Desa 11])*** | ***ModEnzA (Tier I+II) (from [Desa 11])*** | ***EnzML*** |
|---|---|---|---|---|
| ***Annotation Benchmark*** | KEGG (Nov 2010) | KEGG (Nov 2010) | KEGG (Nov 2010) | Swiss-Prot (Dec 2010) $\bowtie$ KEGG (Jan 2011) |
| ***Training Data*** | Swiss-Prot, TrEMBL, KEGG (2004) | Swiss-Prot, ENZYME (Jan 2010) | Swiss-Prot, ENZYME (Jan 2010) | Swiss-Prot (Dec 2010) $\bowtie$ KEGG (Jan 2011) |
| ***E. Coli*** | | | | |
| ***Sequences*** | *856 (1051)* | *892(1021)* | *919 (1082)* | **13,800** |
| Sensitivity | 86.11 | 89.73 | 92.45 | **97.8 (A)** |
| Specificity | 81.44 | 87.36 | 84.93 | **99.9 (B)** |
| Precision | | | | 98.1 (C) |
| Accuracy | | | | 97.3 (D) |
| ***EC numbers*** | *647 (728)* | *648 (697)* | *683 (775)* | **927** |
| Sensitivity | 86.26 | 86.4 | **91.06** | 88.5 (E) |
| Specificity | 88.87 | 92.96 | 88.12 | **99.9 (F)** |
| Precision | | | | 93.0 (G) |

Table 4.3: *Comparison between methods to predict EC numbers: ModEnzA, EFICAz and EnzML. Data for columns 1-4 was taken from Supplementary Table 3 in [Desa 11], where the original caption reads: "Genome-wide enzyme identification for three bacterial genomes (E. Coli , B. Aphidicola and M. Pneumoniae) and one eukaryotic genome (P. Falciparum) by ModEnzA and EFICAz using KEGG annotations as a benchmark. Numbers within parentheses indicate the total number of sequences or EC numbers identified by each method." The EnzML column contains data from leave-one-proteome-out experiments (jackknife). For example, for the E. Coli rows, all E. Coli proteins belonging to all strains were excluded from the* $Swiss-Prot \bowtie KEGG$ *training set and used as the test set. The metrics presented (as a percentage) are: (A) Example based recall (sensitivity by sequence), (B) Example based specificity (specificity by sequence), (C) Example based precision, (D) Subset accuracy, (E) Macro-averaged recall (sensitivity by EC class), (F) Macro-averaged specificity (specificity by EC class), (G) Macro-averaged precision. The highest value for each row is highlighted in bold.*

| ***Methods*** | ***EFICAz (from [Desa 11])*** | ***ModEnzA (Tier I) (from [Desa 11])*** | ***ModEnzA (Tier I+II) (from [Desa 11])*** | ***EnzML*** |
|---|---|---|---|---|
| ***Annotation Benchmark*** | KEGG (Nov 2010) | KEGG (Nov 2010) | KEGG (Nov 2010) | Swiss-Prot (Dec 2010) ⋈ KEGG (Jan 2011) |
| ***Training Data*** | Swiss-Prot, TrEMBL, KEGG (2004) | Swiss-Prot, ENZYME (Jan 2010) | Swiss-Prot, ENZYME (Jan 2010) | Swiss-Prot (Dec 2010) ⋈ KEGG (Jan 2011) |
| ***B. Aphidicola*** | | | | |
| ***Sequences*** | *258 (273)* | *262 (271)* | *263 (273)* | **1,514** |
| Sensitivity | 93.81 | 95.27 | 95.63 | **99.5 (A)** |
| Specificity | 94.5 | 96.67 | 96.33 | **99.9 (B)** |
| Precision | | | | 99.5 (C) |
| Accuracy | | | | 98.8 (D) |
| ***EC numbers*** | *227 (238)* | *220 (229)* | *220 (233)* | **368** |
| Sensitivity | 91.53 | 88.7 | 88.7 | **97.5 (E)** |
| Specificity | 95.37 | 96.06 | 94.42 | **99.9 (F)** |
| Precision | | | | 98.0 (G) |

Table 4.4: *ModEnzA, EFICAz and EnzML comparison continued. Same caption as Table 4.3. The highest value for each row is highlighted in bold.*

| ***Methods*** | ***EFICAz (from [Desa 11])*** | ***ModEnzA (Tier I) (from [Desa 11])*** | ***ModEnzA (Tier I+II) (from [Desa 11])*** | ***EnzML*** |
|---|---|---|---|---|
| ***Annotation Benchmark*** | KEGG (Nov 2010) | KEGG (Nov 2010) | KEGG (Nov 2010) | Swiss-Prot (Dec 2010) ⋈ KEGG (Jan 2011) |
| ***Training Data*** | Swiss-Prot, TrEMBL, KEGG (2004) | Swiss-Prot, ENZYME (Jan 2010) | Swiss-Prot, ENZYME (Jan 2010) | Swiss-Prot (Dec 2010) ⋈ KEGG (Jan 2011) |
| ***M. Pneumoniae*** | | | | |
| ***Sequences*** | *112 (149)* | *114 (139)* | *114 (139)* | **297** |
| Sensitivity | 84.84 | 86.36 | 86.36 | **97.7 (A)** |
| Specificity | 75.16 | 82.01 | 82.01 | **99.9 (B)** |
| Precision | | | | 94.3 (C) |
| Accuracy | | | | 95.2 (D) |
| ***EC numbers*** | *91 (122)* | *102 (122)* | *102 (122)* | **191** |
| Sensitivity | 79.82 | 89.47 | 89.47 | **95.6 (E)** |
| Specificity | 74.59 | 83.6 | 83.6 | **99.9 (F)** |
| Precision | | | | 92.8 (G) |
| ***P. Falciparum*** | | | | |
| ***Sequences*** | *296 (480)* | *321 (415)* | *327 (431)* | **1,975** |
| Sensitivity | 54.91 | 59.55 | 60.66 | **97.9 (A)** |
| Specificity | 61.66 | 77.34 | 75.87 | **99.9 (B)** |
| Precision | | | | 99.9 (C) |
| Accuracy | | | | 96.0 (D) |
| ***EC numbers*** | *186 (247)* | *207 (234)* | *210 (242)* | **368** |
| Sensitivity | 62.2 | 69.23 | 70.23 | **96.7 (E)** |
| Specificity | 75.3 | 88.46 | 86.77 | **99.9 (F)** |
| Precision | | | | 99.9 (G) |

Table 4.5: *ModEnzA, EFICAz and EnzML comparison continued. Same caption as Table 4.3. The highest value for each row is highlighted in bold.*

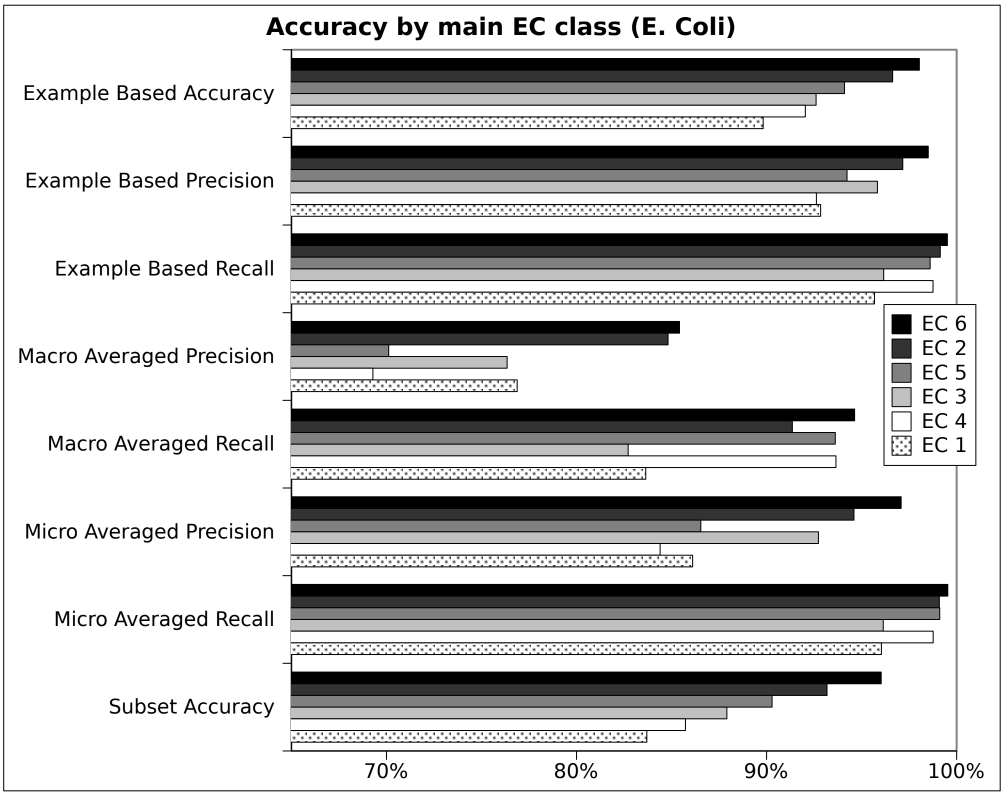


Figure 4.13: *Accuracy of EnzML predictions of* E. coli *by main EC class.*

The comparisons show that EnzML achieves greater overall sensitivity and specificity on a greater number of sequences, as this method uses more recent data. It also achieves very high accuracy and precision, important measures that, unfortunately, cannot be compared as they were not published in the papers cited [Clau 03, Desa 11, Tian 04].

## 4.3 Discussion

### 4.3.1 Effects of EC distribution

As is often the case with long-tail distributions, the shape of the distribution is conserved even when the data are further categorised, as visible in the similarity of distributions for single species and full databases (Figure 4.4 on page 57). This could be caused by evolutionary conservation of certain metabolic functions. Individual species, even compact bacterial genomes such as *E. coli*, have redundancy in certain enzymatic functions, and these functions seem to be common across species, leading to very frequent EC numbers such as cytochrome-c oxidase (EC 1.9.3.1, mitochondrial respiration pathway) alone representing 12% of all UniProt enzymes.

Rare EC numbers do not impact on most evaluation measures as they only affect a small number of proteins. However, in Figure 4.6 on page 61 we can note that the macro-averaged recall, a measure affected by the misprediction of rare classes is generally the lowest and most unpredictable metric for the EnzML method. This is also visible in the wider standard deviation in Figures 4.7 on page 62 and 4.11 on page 67. Also, the macro-averaged recall of *Swiss-Prot* $\bowtie$ *KEGG* cross evaluation is lower than expected at 83%, despite only 20% of its EC numbers being very rare (having less than 3 proteins) versus 63% in invertebrates and 22% in bacteria. However, the measure improves (from 83% to 88%) if 20-fold cross evaluation is used instead of 10-fold, hence raising the probability of having more examples of rare and very rare EC numbers in the training set (data not shown).

It is also intriguing to interpret the EC distribution in light of Bentley's *et al.* [Bent 09] work on long-tail distributions. Their model would suggest that different enzymes have repeatedly "entered" certain catalytic functionalities and, more rarely, "innovated" into new biochemical reactions. The "entering" of a new enzyme in the pool could be interpreted as mutations that make its sequence different enough to become a new protein (by mutation of an existing gene, gene duplication or gene fusion), while still preserving the EC number of its reaction. Only much more rarely will the mutations change the EC number altogether (that is, produce a distinct enzyme with different functionality). In addition, the effect could also be caused by human annotation bias towards more known enzymes.

### 4.3.2 Effects of protein sequence redundancy

In general, a machine learning test set should mirror the distribution of instances in nature. The results presented in this chapter have included datasets that span different levels of sequence redundancy: from full datasets (*Swiss-Prot* $\bowtie$ *KEGG* cross evaluation and *TrEMBL* $\bowtie$ *KEGG* testing), to UniRef sets, down to individual proteomes. Figure 4.14 represents in a schematic way the effects that data redundancy can have on the machine learning results, with predictive performance usually, but not necessarily, decreasing with redundancy (as visible in Figure 4.7 on page 62 for the UniRef50, UniRef90 and UniRef100 datasets). The use of a full, redundant dataset generates reasonable evaluation for scenarios such as entire knowledge base verification or metagenomic or microbiome analysis (soil, gut flora, sea floor samples). A less redundant dataset (such as the test on individual species) generates estimates more relevant to single proteome analysis.

## 4.4 Summary

The EnzML method can be applied to any sequenced protein, without the need for existing annotation or protein structures, and it can provide quick, accurate and complete results on extensive enzyme datasets. EnzML leverages the evolutionary similarity of metabolic function without losing performance when sequences redundancy is reduced.

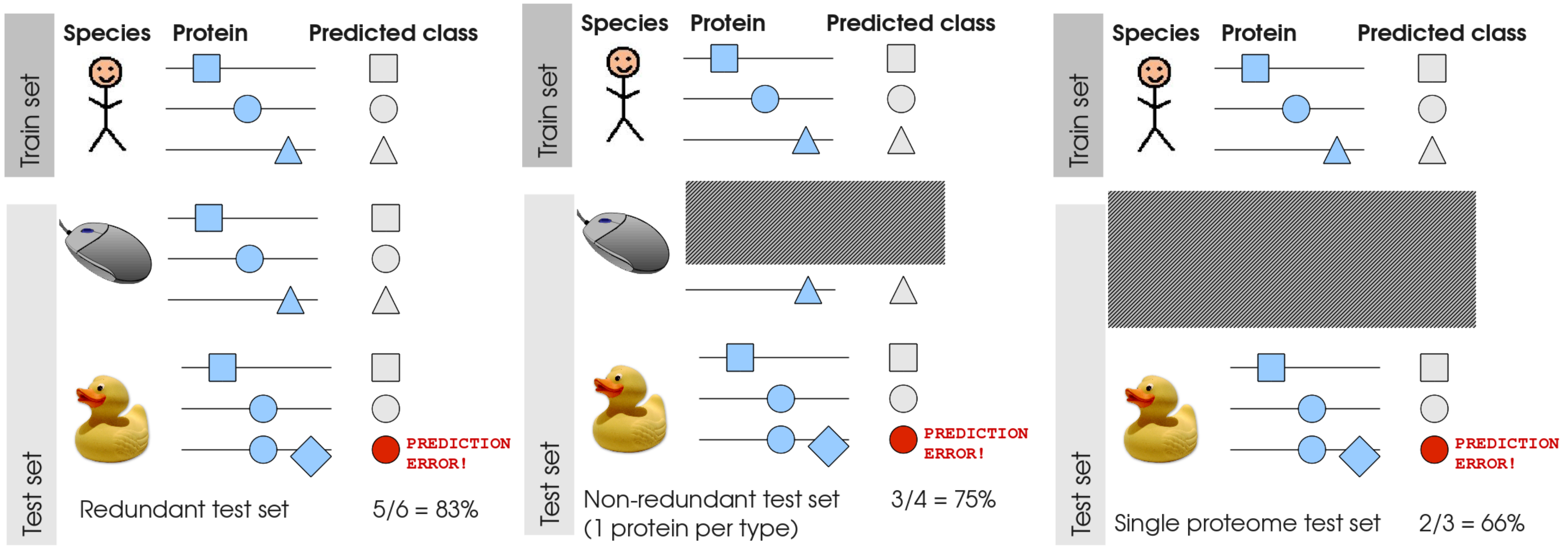


Figure 4.14: *Schematic example of the effect of sequence redundancy on accuracy. Duck differs from the other species in its third protein, which has a "diamond" domain, conferring the protein a floating functionality not present in stick man or mouse; hence the correct class should be "diamond", not circle. The machine learning method, having only seen square, circular and triangular domains in its training set, predicts it incorrectly. Accuracy = number of true positives divided by the total number of sequences in the test set. The relative impact of the error is inversely proportional to the redundancy in the dataset: 17% if the test set includes all proteins, 25% if only one protein per attributes set is included, up to 33% if only the duck genome is under test.*

Thanks to the MULAN BR-kNN implementation, this is possible in reasonable time even for million of sequences, showing clear potential for metagenomic analysis. The approach demonstrates the potential of InterPro signatures in predicting enzymatic function and easing the backlog of manual curation of enzymatic function. EnzML also provides better classification than simply using the InterPro2GO method [Camo 04] currently applied in UniProt and GO to EC transitive assignments.

The final aim is to couple EnzML with active learning to further reduce the number of annotated instances needed, saving precious annotators' time while further speeding up the predictions. The goal is to create a virtuous cycle between automatic and manual annotation, able to keep up with high-throughput sequencing, as described in the next chapter.

# Chapter 5

# Active curation

The overall aim of this work is to scale up the curation of enzymatic function to the task of annotating all existing enzymes. Chapter 3 explores two possible curation environments – open and closed – that could host this endeavour. Chapter 4 presents an accurate method to predict enzyme functionality, possibly the most accurate currently available in the literature. This chapter brings together manual and automatic curation by introducing active and guided learning. Active learning uses machine learning not only to predict, but also to prioritise the labelling of individual instances by curators. Guided learning also provides heuristics to prioritise the curation but it does so without the help of a machine learning algorithm. Guided learning provides indications about the characteristics of instances to be prioritised, but leaves the decision of which particular instance to label to the curator.

The process of curating with active or guided learning methods, which I refer to as *active curation*, could help as much in prioritising scarce manual curation as it could improve the speed and accuracy of the machine learning.

This chapter will simulate a number of active and guided learning methods on a curation workflow. These methods will be compared in terms of speed and prediction quality. Recalling the WikiSim model where users randomly pick records from their internal knowledge, here, in contrast, the users can coordinate their actions using an ordering specified by the system. The ordering is crucial to the effectiveness of the active learning. Perhaps proteins having uncertain predictions should be annotated first, but they may be a waste of curation if they are very easy to predict, so uncertainty, frequency of attributes, similarity to other unlabelled instances and dissimilarity from labelled instances could all have a role to play.

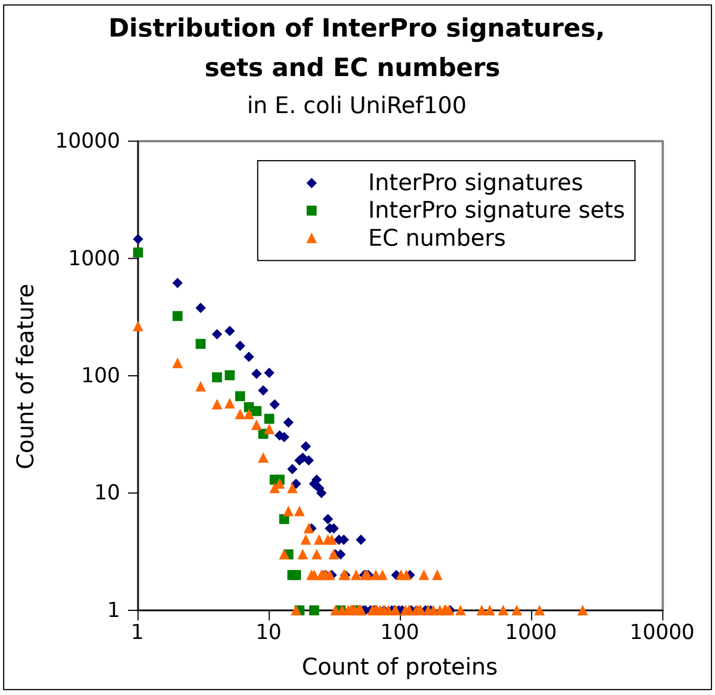


Figure 5.1: *Distribution of InterPro signatures, InterPro signature sets and EC numbers in the Ecoli_UniRef100 dataset. Both axes are logarithmic.*

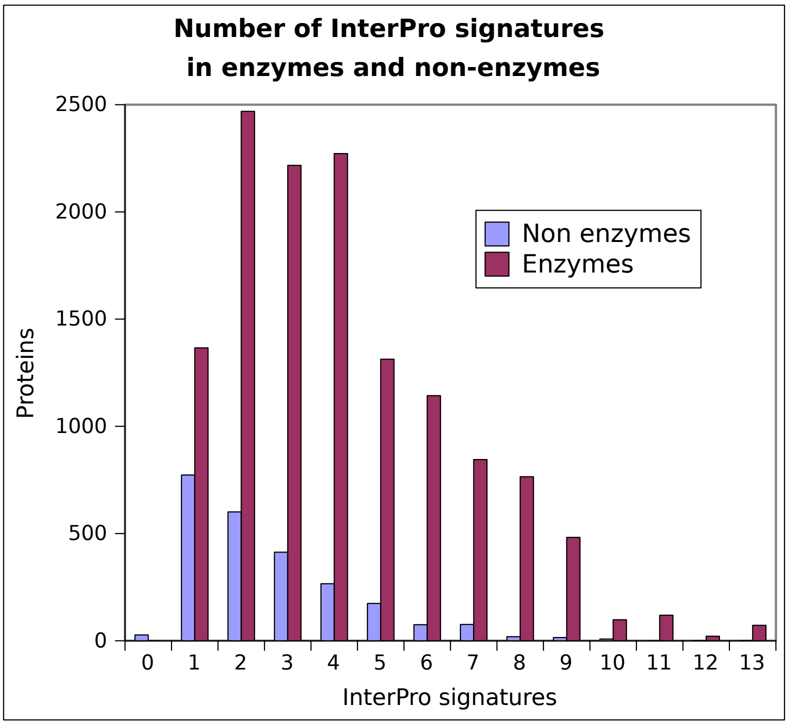


Figure 5.2: *The number of InterPro signatures for enzymes and non-enzymes in the Ecoli_UniRef100 dataset. Enzymes have, on average, larger InterPro signature sets than non-enzymes.*

## 5.1 Methods

### 5.1.1 Data

The dataset used in this chapter is the set of UniRef100 *E. coli* proteins (all strains) taken from the *Swiss-Prot* $\bowtie$ *KEGG* dataset of Chapter 4 (from now on referred to as *Ecoli_UniRef100*).

*Ecoli_UniRef100* provides a compact, but not too small set (5,750 proteins) taken from one of the most intensely investigated model organisms in molecular biology. To reduce the dataset redundancy, only one protein (the first alphabetically) for each UniRef100 cluster has been included. *Ecoli_UniRef100* has all the characteristics already seen in other protein datasets in Chapter 4: long tailed distribution of EC numbers (927 EC numbers), attributes (3,931 InterPro signatures) and attribute sets (2,122 distinct sets of InterPro signatures). Their distribution is shown in Figure 5.1. InterPro sets tend to be smaller in non-enzymes (with an average of 3.9 InterPro signatures) than in enzymes (5.6 signatures). Figure 5.2 shows the distribution of InterPro sets size of enzymes and non-enzymes.

EC numbers often appear together with sets of InterPro domains, as shown in Figure 5.3. For example, in *Ecoli_UniRef100*, 50% of the InterPro sets are only connected to non-enzymes and 37% are connected to a single four-digit EC number (745 InterPro sets, accounting for 2,543 proteins). This facilitates the task of the K-Nearest Neighbours algorithm.

### 5.1.2 Active learning

**Algorithm and distance** The base machine learning algorithm used in this chapter is BR-kNN with k=1, already described in the Methods section of the previous chapter. Due to the size of the domain, sophisticated distances (such as Mahalanobis) among instances are too computationally intensive to be used. Hence the methods will use the Euclidean distance (Figure 5.4 on page 81) used for BR-kNN throughout the previous chapter.

**Active learning cycle** The active or guided learning methodology generally works as described below. The core difference between methods is how the *utility* of each unlabelled instance is calculated. Utility here is a numerical value emitted by the method to represent the potential contribution of each unlabelled instance to improve prediction for the next learning step.

Input: Unlabelled instances $U$, Labelled instances $L$. $L$ can be empty initially.

1. For each $u \in U$, calculate utility

2. Select a query instance ($q \in U$) with best utility: $q \leftarrow arg\,max\,utility(u),\ \ u \in U$

3. Label query: $q \in L$, $q \notin U$

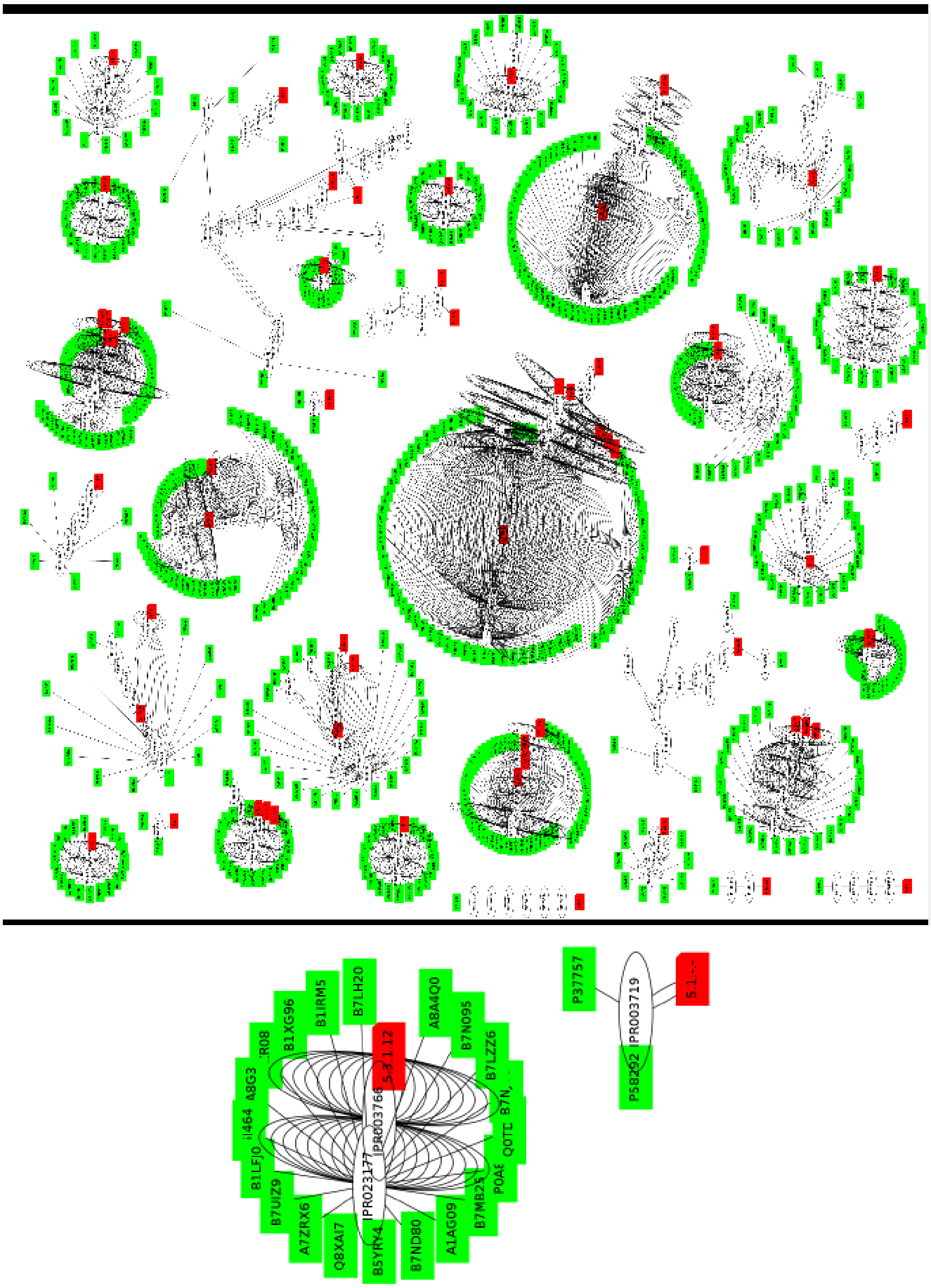


Figure 5.3: *Proteins having the same superset InterPro signatures, and hence often nearest neighbours for the purpose of machine learning, are also tightly clustered in terms of class. Proteins (in green) connected to the same InterPro signatures (in white) are usually connected to only one EC number (in red). Clusters having more then one red label often contain several distinct but partly overlapping InterPro sets. Dataset: E. coli all strains, proteins with EC 5.-.-.- base number.*

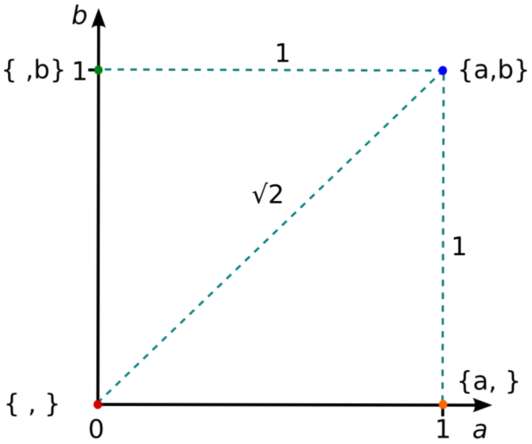


Figure 5.4: *Euclidean distance. An example from a domain with two attributes: $a$ and $b$. Each set of attributes in curly brackets represents an instance. Instance $\{a,b\}$ is maximally distant ($\sqrt{2}$) from the empty instance $\{,\}$. Instances that differ in only one attribute, such as $\{a,\}$ and $\{a,b\}$, have a Euclidean distance of 1.*

4. Train on $L$

5. Test on $U$ and emit evaluation metrics

6. If $U > 1$, GOTO 1

**Randomisation** The baseline method against which the other methods are compared is a uniformly random selection of protein instances. In fact, all methods described in this chapter break utility ties at random, that is, if the method emits the same (best) utility for more than one instance, the query instance to be labelled is chosen uniformly at random among those instances having the best utility.

Three active learning methods are presented: Global Distance, Confidence Based and Global Distance with Confidence.

**A1 Global Distance** The Global Distance method was inspired by Fujii *et al.* [Fuji 98] and Jain and Kapoor [Jain 09] (also discussed in Section 2.5.2 on page 27). In order to pick the best instance to be labelled next, the *Global Distance* method takes into account the sum of the distance (from which the "global" term) between the unlabelled instance and the remaining unlabelled and labelled instances. The query protein is the one having maximum distance to other labelled proteins and minimum distance to other unlabelled proteins (graphically represented in Figure 5.5). Thus, as next query instance (instance to be labelled) Global Distance prefers one:

1. Close to other unlabelled ($U$) instances and

2. Far away from already labelled instances ($L$)

The Global Distance utility for each instance is defined as:

$$GlobalDistance_{(u)} = \lambda \sum_{i \in L}^{L} distance_{(u,i)} - (1-\lambda) \sum_{j \in U}^{U} distance_{(u,j)} \tag{5.1}$$

where $u$ is the unlabelled instance, $\lambda$ is a balancing factor (discussed below), $L$ is the set of $i$ labelled instances, $U$ is the set of $j$ unlabelled instances and $distance$ is a function that returns the Euclidean distance $+1$ between the instance $u$ under consideration and another (labelled $l_i$ or unlabelled $u_j$) instance (1 is added to make sure that instances with distance 0 have an impact on the utility). $\lambda$ is a balancing factor between zero and one that gives more or less weight to the overall distance between the instance and all the labelled instances (if set $\lambda > 0.5$) or unlabelled instances (if set $\lambda < 0.5$). Fujii *et al.* [Fuji 98] obtained the best results where the two factors were equally balanced, consequently $\lambda = 0.5$ is used in the following experiments.

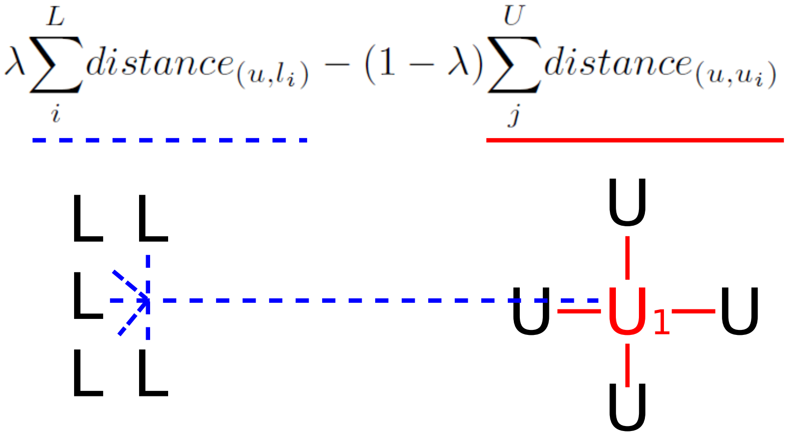

$\lambda \sum_{i}^{L} distance_{(u,l_i)} - (1-\lambda) \sum_{j}^{U} distance_{(u,u_i)}$
L
L
L
L
L
U
U
U1
U
U

Figure 5.5: *A schematic representation of the Global Distance formula. For unlabelled instance $U_1$, the dashed lines represents the distance to the labelled instances, the continuous lines the distance to the other unlabelled instances.*

**A2 Confidence Based** The Confidence Based method uses the measure of confidence that the BR-kNN algorithm emits with its predictions. The utility is highest (better) for instances having confidence close to 50%:

$$Utility(u_i) = 0.5 - \mid 0.5 - \frac{\sum_{l=1}^{n} c_{i,l}}{n} \mid \tag{5.2}$$

where the utility of unlabelled instance $i$ is the distance from 50% of the confidence for each of its labels' prediction. In BR-kNN, the confidence on the prediction of label $l$ for instance $i$ ($c_{i,l}$) is the neighbours vote on label $l$: $c_{i,l}$ is the count of nearest neighbours of $i$ having label $l$, divided by the total number of nearest neighbours for $i$.

Thus instances with predictions confidences close to 100% (certain of that label prediction) or 0% (certain of dismissing that label) are the least useful and the last to be labelled.

**A3 Global Distance with Confidence** The Global Distance with Confidence method combines the Global Distance utility with the Confidence Based utility by multiplying them.

### 5.1.3 Guided learning methods

Four guided learning methods are now defined. In contrast with active learning, guided learning does not require an input from a machine learning algorithm. The methods presented are based on attribute statistics of the unlabelled dataset. The methods simulate a curation scenario where a certain attribute (or set of attributes) are selected and then the curators are asked to provide *at least one* labelled protein having that attribute (or set of attributes), in the style of [Atte 10a]. The decision of which exact instance to annotate is hence delegated to the curators, who have to provide more intelligence, but are also more free to exercise their professional judgement regarding which protein could be more fruitful to annotate.

Guided learning can have remarkable advantages over active learning. Guided methods are computationally less intensive, more considerate in treating rare classes and can be parallelised over a number of human oracles [Atte 10b, Atte 10a, Atte 11]. Attenberg *et al.* [Atte 10a] suggests asking curators to provide a labelled instance for a defined *class*, whereas the methods presented here are based on pre-selecting *attributes* instead. The reason is that it is easier for biologists to find at least one protein having a certain sequence signature than finding a protein with a certain EC number. Curators can search proteins by InterPro attributes, but they cannot easily search them by EC class before they have been labelled. Selection by attribute is also more appropriate when multiple EC classifications are considered. Once the attribute is set, the protein under study can be labelled with several EC numbers if this is the opinion of the curator, while asking a curator to come up with an example of a certain set of two or three EC numbers might be difficult. The sheer number of combinations of two, three or four EC numbers would overwhelm the curation process. Thus four methods are presented, two based on InterPro attributes and two on InterPro attribute sets. In order to conduct simulation studies, the manual step of selecting an instance for labelling is here substituted by random selection.

**G1 Random selection of InterPro attributes** This method selects one InterPro attribute uniformly at random. Once the attribute is selected, a randomly selected instance having that attribute is labelled. Equal curation effort is thus dedicated to each attribute, regardless of its frequency among proteins.

**G2 Ordered selection of InterPro attributes by frequency** In this method the attributes are ordered by their frequency in the unlabelled dataset first. The attributes are then tackled one by one, starting with the most frequent. For each attribute, an instance, selected uniformly at random, is labelled. When the end of the ranked list of attributes is reached (with 3,931 InterPro signatures and only 5,750 proteins, this happens only once), the selection starts again from the most frequent attribute, until all unlabelled instances have been selected and labelled.

In real life, when faced with a certain attribute or attribute set, a curator might be able to find a certain example protein based on his expertise or literature, but not another protein. Selecting the attribute and then selecting the instance to be labelled at random tells us how much variability the picking of one example versus another introduces in the overall accuracy. This in turns affects the views of correct annotations available to the public during active curation.

**G3 Random selection of InterPro sets** This method works exactly like the random selection of InterPro attributes, only, the selection happens over InterPro attribute *sets*.

**G4 Ordered selection of InterPro sets by frequency** This method works exactly like the ordered selection of InterPro attributes by frequency, but the selection happens over ordered InterPro attribute *sets*. With 2,122 InterPro sets and 5,750 proteins, the method scrolls over the ranked list of attribute sets two and a half times before exhausting the unlabelled instances.

### 5.1.4 Hybrid active-guided methods

The two neighbourhood methods presented have characteristics of both active and guided learning. A neighbourhood is defined as all the instances being nearest neighbours of each other (it can also be defined as the set of all the nearest neighbours of an instance, plus the instance itself). Each instance only appears in one neighbourhood. The neighbourhood methods hence require a K-Nearest Neighbours machine learning algorithm to calculate the neighbourhoods, but then use simple heuristics to rank the neighbourhoods in order of utility and leave to the curators the choice of which neighbourhood instance should be labelled.

**H1 Random selection of neighbourhood** In the random selection of neighbourhood, a neighbourhood is selected at random and then one of its instances is selected at random for labelling. The process continues with another selected neighbourhood until all neighbourhoods have been labelled once. The process then continues until all neighbourhoods have two labelled instances and so on, until all the instances have been labelled.

**H2 Ordered selection of neighbourhood by frequency** The ordered selection of neighbourhood by frequency randomly selects one query instance from each neighbourhood, starting from the largest neighbourhood. When all the neighbourhoods have provided one instance, the method starts again from the biggest neighbourhood and continues until all instances have been labelled. In practice, considering that in enzyme datasets all the instances in a neighbourhood are likely to have the same InterPro set, this method is very close to the selection of InterPro sets by frequency.

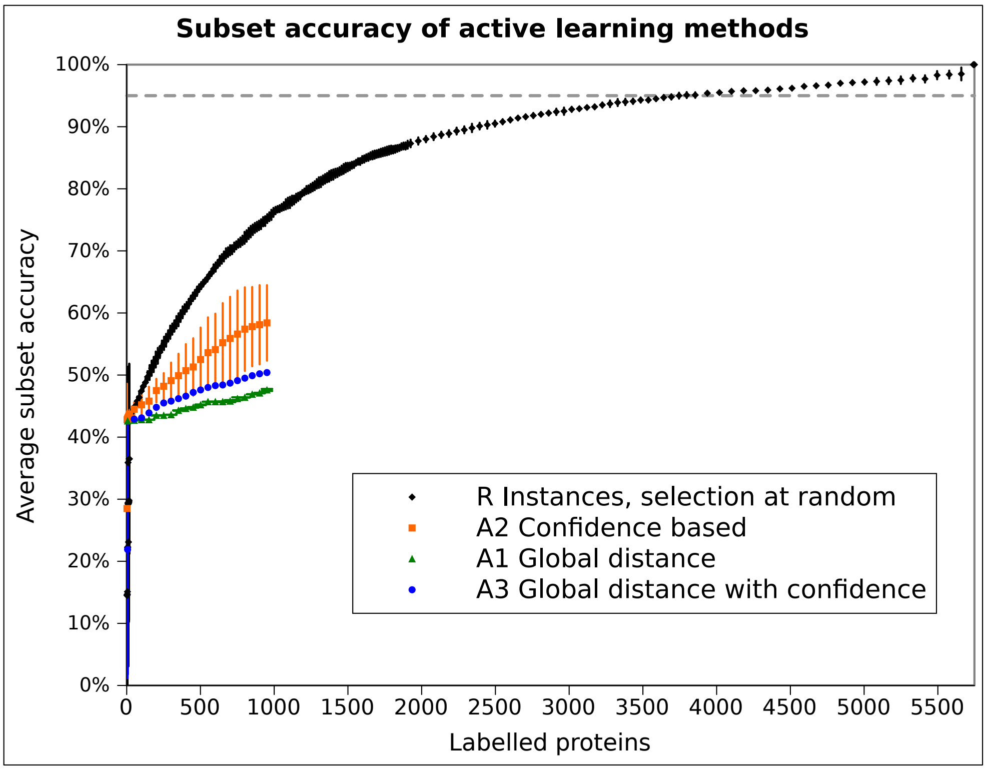


Figure 5.6: *Average subset accuracy of active learning methods. The dashed line parallel to the x axis marks the maximum subset accuracy reached in 20-fold cross-evaluation experiments (96%).*

### 5.1.5 Evaluation

An active learning experiment starts with all instances (proteins) in the testing (unlabelled) set. One by one the proteins are extracted in the order provided by one of the methods described above, labelled and added to the labelled set (training set). After each labelling, the utility of the remaining unlabelled instances is recalculated if the method requires it (Global Distance, Confidence Based and Global Distance with Confidence). The performance of the predictions on the remaining unlabelled instances (test set) is also re-calculated after each labelling.

The prediction metrics are then plotted and compared with the baseline of randomly se-

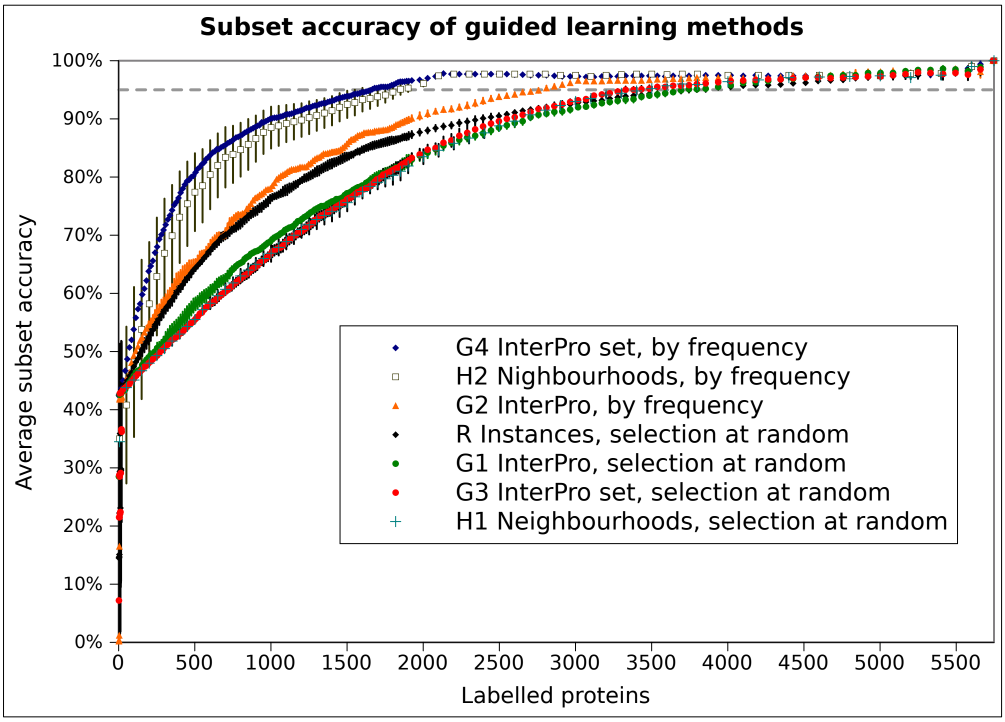


Figure 5.7: *Average subset accuracy of guided learning methods. The dashed line parallel to the x axis marks the maximum subset accuracy reached in 20-fold cross-evaluation experiments (96%). The methods in the legend are ordered from the best to the worst performing.*

lecting instances. Since all methods include some randomisation, all the experiments were repeated five times (three times and only for the first 1000 steps for Global Distance, Confidence Based and Global Distance with Confidence because of their computational intensity). Two metrics are shown for comparison: *subset accuracy*, the most strict multi-label metric because it only considers the exact prediction of the entire label set and *macro-averaged recall* is also included because it is regularly the lowest metric in absolute terms and it is the most sensitive to class coverage. The macro-averaged recall is also relevant because it is biologically important that rare classes are not overlooked. The metrics are averaged, plotted as a function of the number of labelled instances and presented together with their standard deviation. An algorithm or method is deemed superior to another if its curve dominates the other method's for most or all the points in the plot.

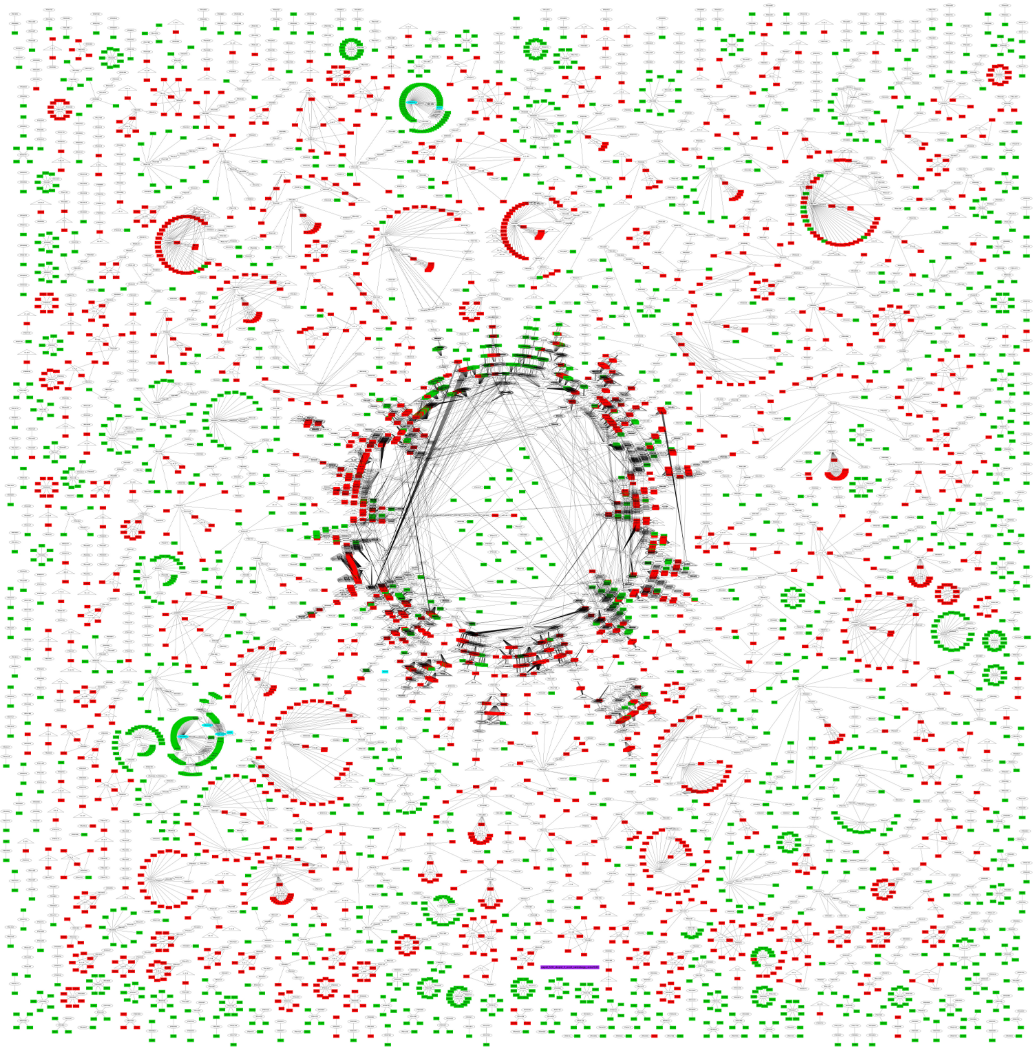


Figure 5.8: *Graphical view of correctly predicted proteins (green rectangles) and incorrectly predicted (red) at the second step of active learning (with only two labelled proteins). The green proteins are all non-enzymes, while the red proteins are enzymes. InterPro attributes are shown as white ovals and EC numbers as white triangles (partial EC numbers are not included to improve readability).*

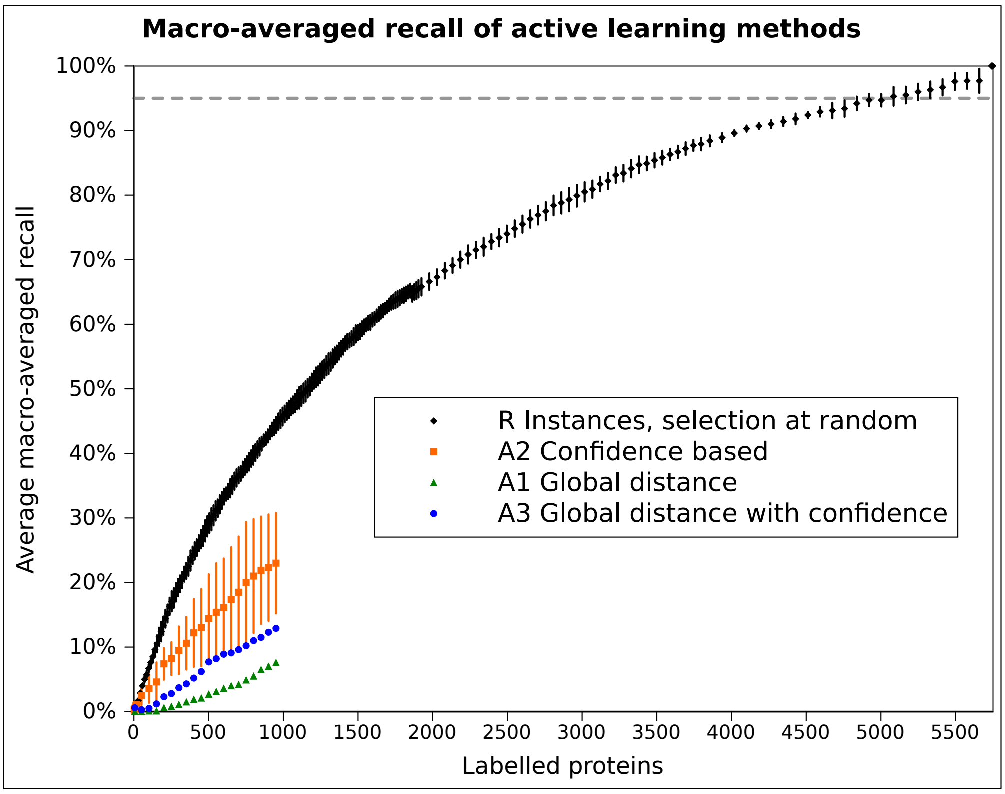


Figure 5.9: *Average macro-averaged recall of active learning methods. The dashed line parallel to the x axis marks the maximum macro averaged reached in 20-fold cross-evaluation experiments (95%).*

## 5.2 Results

### 5.2.1 Subset accuracy

Figures 5.6 and 5.7 show the subset accuracy of all active and guided learning methods presented, respectively. The accuracy of all methods starts at about 40% because the most common class in the data set is non-enzyme. If a method does not have enough confidence yet to assign any of the labels, then the non-enzyme instance is predicted "correctly" as not having any label. To express this graphically, Figure 5.8 shows the *Ecoli_UniRef100* protein network coloured according to the prediction status (green for correctly predicted, red for incorrectly predicted). The figure shows that all non-enzymes already appear marked in green at the third step of active learning (with only two labelled instances).

From these results in Figures 5.6 and 5.7, the random baseline (R - instances, selection at random) appears to be a rather strong guided learning method in itself, probably because it samples the underlying data well. In the following paragraphs, the other methods are discussed from the worst to the best performing. The Confidence Based, Global Distance and Global Distance with Confidence methods are much less effective than random. A more detailed

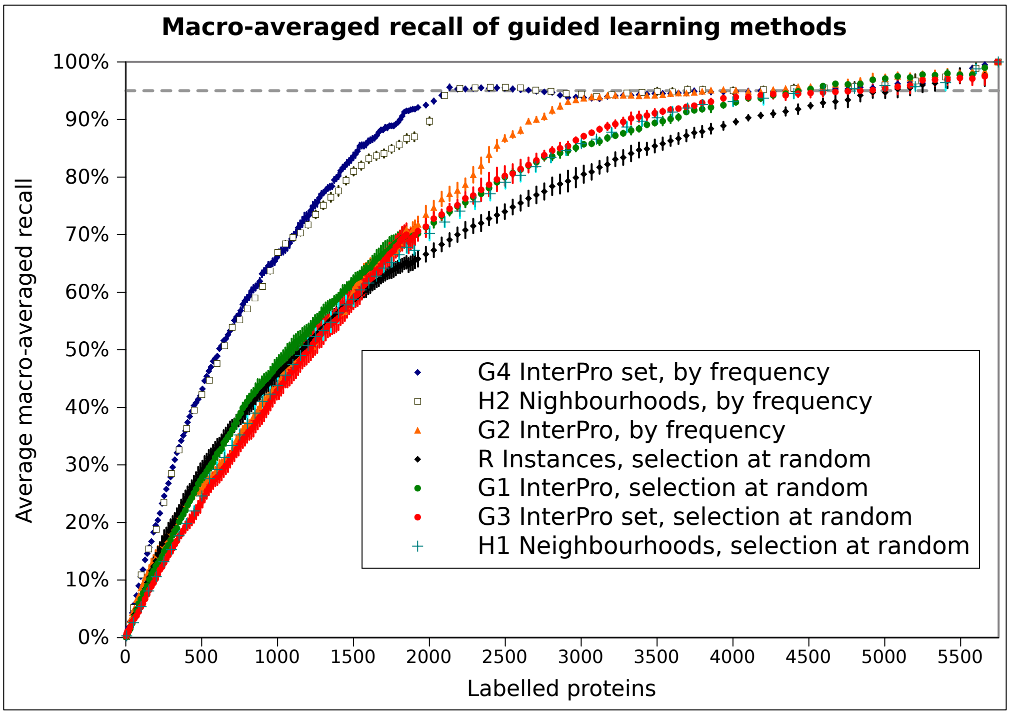


Figure 5.10: *Average macro-averaged recall of guided learning methods. The dashed line parallel to the x axis marks the maximum macro averaged reached in 20-fold cross-evaluation experiments (95%).*

analysis of their choice of query instances shows that the Confidence Based method tends to "oversample" each cluster of instances having the same InterPro set, often labelling 10 or 20 instances before moving on to a new cluster. The Global Distance method behaves in a similar way and it also prefers instances with very small attribute sets (only one InterPro domain), probably because they have smaller Euclidean distances overall.

The *random selection of Interpro attributes*, *InterPro sets* and *neighbourhood* perform slightly worse than a random choice of instance. The *ordered selection of Interpro by frequency* performs slightly better than random instance selection. The best two methods are *ordered selection of InterPro sets* and *neighbourhood by frequency* and they are clearly better strategies than random selection of instances. When the computational cost and the macro-averaged recall are also considered, the *ordered selection of InterPro sets by frequency* appears to be the best method. The *ordered selection of InterPro sets by frequency* reaches 95% subset accuracy when only 20% of the proteins have been labelled (97% subset accuracy is reached after 35% of the proteins have been labelled). That is, the method accuracy peaks when about four fifths of the most common InterPro sets have one labelled instance each. For comparison, picking instances at random only reaches 95% subset accuracy after 65% of the proteins have been labelled.

### 5.2.2 Macro averaged recall

The method of *selecting InterPro sets by frequency* also outperforms all the other methods in terms of macro-averaged recall (Figures 5.9 and 5.10), reaching 95% macro averaged recall when 37% of the proteins have been labelled. Picking instances at random only reaches 95% macro averaged recall after 86% of the proteins have been labelled.

In terms of computational and memory cost, the Global Distance with Confidence method is the most expensive, followed by the Confidence Based method and the Global Distance method. The guided learning methods have very limited computational cost, but also require more input from the oracle in terms of which instance to target.

## 5.3 Discussion

The results indicate that the best active curation strategy is to cover as many InterPro sets as possible, starting with the most frequent. However, there is some circularity in the evaluation of all these methods. The labelling of most frequent features boosts the accuracy measures by, in a sense, bagging "easy", similar instances. This is still good guidance for curators, in the sense of directing effort towards proteins that can cascade information to many other sequences. However, it is also important to examine carefully measures such as the macro-averaged recall, which are less biased by sequence redundancy.

In the extensive datasets examined, the existence of repeated InterPro sets is unsurprising. However, InterPro sets that are most frequent in knowledge bases might not be the most frequent in nature, but simply those more easily sequenced or frequently studied. Conserved signatures are recognised as such exactly by the fact of appearing in many similar sequences. Signatures are only connected to a function (and hence usable for prediction) if the sequences have been studied experimentally, which brings us back to the initial bias on which proteins are most studied.

In the end, EnzML and the guided curation strategies presented are simply very good at "reproducing" human curation. This is no mean feat, but EnzML, while giving a hint of how human curation might work (by recognition of known functional elements) also inevitably reproduces the same bias and mistakes present in the original data. If the method is applied on a more vast scale, the cascading of errors would be inevitable. On the other hand, with EnzML embedded into collaborative curation, errors would also be automatically corrected in the training set when new evidence appears, and hopefully this new knowledge would be extended to the unlabelled proteins. The emphasis of this thesis is on extrapolating existing knowledge over proteins not yet manually curated as accurately as possible. The human-computer interaction layer will also be of the utmost importance. Ideally, it would visually present the EnzML predictions in the curation environment as separate from manual annotations, and with as much justification as possible in terms of confidence and neighbours used for prediction.

An evaluation of how many EC numbers are assigned on the basis of experimental results and how many are assigned by sequence or structure similarity would be interesting to assess the validity of all enzyme prediction studies. EC numbers do not usually have justifications attached, but a similar study might be possible on Gene Ontology terms related to enzymatic function, because they better record justification.

Only three simple active learning methods have been applied, so this analysis does not exclude that more sophisticated active learning methods might outperform guided learning. However, the strategy of selecting InterPro attribute sets by frequency could immediately be applied to UniProt TrEMBL. TrEMBL contains 127,127 InterPro sets which do not exist at all in the manually curated Swiss-Prot UniProt. Of these, the most frequent, a combination of peptidase and reverse transcriptase signatures[1] alone accounts for 53,285 protein sequences. This set has many proteins, but all very similar in sequence, possibly because many human HIV virus variants have been completely sequenced. If the sequence redundancy is reduced, for example considering only UniRef50 reference sequences, the most frequent InterPro set becomes a combination of RNA polymerase sigma factors[2] which includes, in the same InterPro set, a diverse group of 2,093 sequences. Based on the evidence presented in this chapter, these InterPro sets could be a prime target for manual inclusion into Swiss-Prot (prime, but not necessarily easy ones) as they could cause a significant improvement in TrEMBL predictions.

## 5.4 Conclusion

This chapter demonstrates the potential of guided learning methods based on InterPro sets for curating enzymes. If all protein data behaved as well as the *Ecoli_UniRef100* set, the task of annotating all enzymes – at accuracy close to that of current manual curation – could be achieved with only *a third* of the manual curation necessary without these methods.

[1] Peptidase set prevalently found in viruses: IPR018061. Pept_A2A_retrovirus_sg. IPR001995. Peptidase_A2_cat. IPR021109. Peptidase_aspartic. IPR001969. Peptidase_aspartic_AS. IPR009007. Peptidase_aspartic_catalytic. IPR000477. RVT. IPR010661. RVT_thumb.

[2] Polymerase set: IPR014284. RNA_pol_sigma-70. IPR007627. RNA_pol_sigma70_r2. IPR013249. RNA_pol_sigma70_r4_t2. IPR013325. RNA_pol_sigma_r2. IPR013324. RNA_pol_sigma_r3_r4. IPR011991. WHTH_trsnscrt_rep_DNA-bd.

# Chapter 6

# Conclusion

This thesis contains three main contributions: a model of annotation, a machine learning method to annotate enzyme function, and an exploration of active and guided learning to improve the enzyme curation process. This concluding chapter will discuss future extensions of these methods to other domains and also the challenges of their evaluation.

## 6.1 Availability of authorship data

Wiki software is not necessarily the best solution to enter and search highly structured data. However, wikis offer very good support for collaborative editing and tracking changes. In particular, any edit in a wiki is usually logged with the author's name, the time stamp of the action and the text difference compared to the previous version. This fosters collaboration by easily identifying contributors (even if not necessarily by their real names), eases the reversion of vandalism and allows tracking of changes in favourite topics and pages.

In contrast, the history of edits for closed biological knowledge bases is often not provided at the same level of detail, especially regarding authorship. For this work, and in particular for the model evaluation (Section 3.3), I considered the closed systems OMIM [Hamo 02] and Reactome [Matt 09] and the open systems EcoliWiki [Ecol 11] and WikiPathways [Pico 08], which do provide authorship data. Unfortunately, many other highly curated knowledge bases and databases either do not track or cannot publicly provide complete authorship information, not even anonymised (communications via e-mail in 2009: from Anne Estreicher at UniProt/Swiss-Prot, from Rachel Kramer Green at PDB and from Harold J. Drabkin at MGI-GO).

This research highlights the lack of public availability of authorship information in many highly curated molecular biology knowledge bases. The current discussion on wikis versus closed systems is sometimes simplistic: wikis should not be trusted as scientific resources because the authors can use pseudonyms, while closed and highly curated databases can be trusted because the authors have been selected among experts. In reality, authors and edits can be more traceable in wikis, while no detailed information about individual edits or annotators is usually available for closed systems; OMIM [Ambe 09], Reactome [Crof 11] and Gene Ontology [Ashb 00] are notable exceptions.

The trust in closed systems seems to be based more on trust about the "source" in general. In 2009, Magnus [Magn 09] proposed an example of how, often, trust in authorship might in fact be trust in a knowledge provider. The example is a news published in the New York Times, an obituary of a famous writer. The reason we trust the factual evidence (that the writer really has died) has usually nothing to do with knowing the journalist who wrote the article and all to do with our general trust in the New York Times as a source of reliable information. In this sense, wikis, especially when they enforce user registration, could be paradoxically welcome in molecular biology as an example of transparent provenance in data curation. Wikis could also provide information on how knowledge and mistakes are distributed in a population of users.

Unfortunately, for now, the extension of the WikiSim model to simulate potential improvements to knowledge bases is limited by the lack of available logs in widely adopted closed systems and the lack of substantial volume of high quality molecular biology annotations in wikis, also due to their recent adoption.

## 6.2 Model applications

The WikiSim model (Chapter 3) allows complete control over simulations and could be extended to explore what parameters affect the completeness and correctness of the knowledge collected. WikiSim can cover conditions not easily measurable or reproducible in real life wikis. The model simulations have the potential to add a dynamic and measurable component to many insights and theories that have been discussed in wiki research in recent years, for example regarding data quality, neutrality of news or collective organisation of work in open knowledge and open source projects.

The WikiSim model offers a framework that could be extended to several research areas. Four areas could be of particular interest and are described in more detail below: modelling the users' internal knowledge, modelling software affordances, modelling structured semantic wiki features and comparing models of open and closed curation. The model could also be used to measure the impact of machine learning on the curation process, by simulating the impact of algorithms exhibiting varying accuracy, or to simulate the impact of oracles with varying quality for active learning in the style of [Donm 08].

**The user competence model** The current WikiSim model is based on an implicit premise: a user acts more on the knowledge items it is more interested in. If we go a step further and say that wiki users are reasonably good judges of their own knowledge and competence, we could introduce a second principle: the more the user is interested in a topic, the more likely it is to have some correct knowledge about that topic, *on average*. Obviously, this does not mean that the user knowledge will be perfect, it just introduces a correlation between the user awareness of its expertise, the quality of the user knowledge and the probability the user acts on it. In the model, this could be represented as a user making less mistakes on topics in its "favourite" knowledge; the interest in a topic correlates with a higher probability of knowing something correct about that topic, while also acting on it more often (the desire to contribute to a favourite topic outweights the effort of adding or editing).

This principle could have a powerful effect on wiki knowledge quality, however, it is challenging to prove; it would need a way of measuring, for a high number of cases how good a person's knowledge of a topic is, how good the person is at judging his own expertise on the topic and how that relates to the person's actions, such as the willingness to edit the corresponding Wikipedia article.

**Modelling wiki software affordances** The WikiSim model could be enriched to represent typical wiki software features, such as discussion spaces and watch lists, and measure their impact on user actions and knowledge quality. Discussion could be represented in the model by allowing the user to consider, before acting, both its own knowledge and a count of other users actions (true/false edits). Automatic notification of watched topics could be modelled by pushing items of interest – recently edited by other users – to the top of a user's action list. The simulations could then quantify how these social affordances, such as visibility of other authors choices, may help to build consensus, better allocate user effort or improve data quality. Also, different costs of addition, edit and search could be used to simulate barriers in editing and their effect. The challenge would then be to find datasets that have or have not been affected by these features to evaluate the model (for example, collections of wiki pages logged before or after a certain software affordance has been introduced to the community).

**Modelling semantic wikis** Structured semantic wikis (such as OntoWiki [Auer 06] or Semantic MediaWiki [Krot 06]) could be modelled by introducing relations between knowledge elements. Before changing a knowledge element in a correlated group, the user could be influenced by the value of the other connected elements. This would also create another measure of knowledge quality in the model: whether the element has been connected to an appropriate semantic group/class. The evaluation here is limited by the limited uptake of these platforms for annotation.

**Modelling open and closed curation** In biology, the wiki approach to data curation is relatively recent. The current curation scenarios (Figure 2.2) are either centralised – with edit reserved to official curators – or mixed, with users allowed to report mistakes but not to perform or discuss the edits. It would be interesting to further simulate and compare the three scenarios. Expertise could be modelled by giving expert curators more correct knowledge and more energy than basic users. Edits could either be exclusively initiated by expert curators or channelled through them. This could have an impact on the volume of knowledge that can be collected. Again, the limiting factor would be the availability of logs for the closed platforms.

## 6.3 Prediction of enzyme function

Regarding potential extensions of the machine learning schema presented in 4, InterPro is working on improving "EC tags" for signatures by using the PRIAM database [Clau 03]. Currently, "EC tags" are assigned statistically: if 80% of Swiss-Prot entries that have a signature also have a certain EC number, that EC number is assigned to the InterPro signature (kindly confirmed by David Lonsdale at EBI). InterPro "EC tags", once improved, could be used as attributes for learning, bypassing the direct use of InterPro signatures. They could provide an even more compact representation of about 4,000 attribute columns (one for each EC number) instead of the about 20,000 columns (one for each InterPro signature) in the method presented here.

The EnzML method could also be extended to learning all gene products annotations, for example in the form of Gene Ontology terms.

My analysis has highlighted dramatic discrepancies (on a third of the total data) between two of the most manually curated and reliable existing knowledge bases of enzymatic function, KEGG and UniProt Swiss-Prot. Curators have the right to disagree, but it would be useful if enzymatic annotation was justified in the same way Gene Ontology terms are, where a functional class is attached to a protein together with a standardised evidence code and link to the evidence, usually in the form of literature or data, as shown for Gene Ontology terms in Figure 2.4 on page 18.

The proposed EnzML method is applicable to any complete or partial protein sequence. Any genetic sequence can be scanned *in silico* for the presence of InterPro signatures using the InterProScan algorithm, also available as web service [Hunt 09, Muld 07], making EnzML a perfect complement to high throughput initiatives as diverse as personalised medicine or metagenomic sampling of ocean floors.

The overall success of EnzML is due to the fact that InterPro signatures provide a very compact representation of protein functionality. The 13.5 million proteins in UniProt are described by only 154,583 (unordered) sets of InterPro signatures (attributes). And many of these sets are very similar, only differing by one signature.

In relation to the method's application and evaluation, it must be noted that the distribution of annotation in metabolic databases tends, by definition, to be more enriched in enzymes than in non-enzymes. Even highly-populated databases such as UniProt are biased, with more accurate annotation (and Swiss-Prot status) going to widely studied biological functions. Using only annotations that agree in two manually curated databases (such as Swiss-Prot and KEGG in this work) increases trust, but decreases the number of EC numbers that can be predicted. Swiss-Prot contains 2,850 distinct EC numbers, and KEGG contains 2,636 EC numbers, but the set of annotations agreeing in both databases only contains 2,051 EC numbers. Rare EC numbers can easily be lost in case of disagreement among the data sources.

The accuracy of the predictions generally increases as the dataset size increases, which, when combined with the efficiency of the algorithm, is a good case for using a bigger training set whenever possible. Training a classifier on more data from non-manually curated databases, such as UniProt-TrEMBL, may reduce the bias and increase the number of predictable classes, but will also decrease trust. Alternative biocuration scenarios may call for a different balance between coverage and trust, to increase the probability of recognising rare Enzyme Commission classes in newly sequenced genomes.

The high accuracy of EnzML, combined with the measure of confidence that the method emits for each prediction, enables the curators to focus their work on correcting the weakest annotations. The majority of erroneous annotations have low confidence, so curators could tackle the more error prone annotations first. However, active learning research has shown that simply correcting low-confidence annotations is rarely the best strategy, as the representativeness and informative content of each instance also have an impact.

## 6.4 Active curation

**Empirical analysis** Active learning has proven useful on many problems. However, an active learner builds a training set that is intimately linked to the model over time; in the worst cases, this set may not represent the original distribution of instances. There are examples of active learning causing the algorithm to require *more* instances than passive learning [Sche 07].

Attenberg [Atte 11] also notes several difficulties in applying active learning to real life domains. One is the potential bias in examples selection that active learning carries, especially if used from the very beginning of the annotation process. A related problem is to obtain enough unbiased data in order to select the base learner algorithm. This is particularly problematic for questions such as "label all news articles related to sports", where no previous training sets may exist. Starting by labelling random examples to avoid initial bias is feasible, but this approach somehow defies the very purpose of active learning, that is to select instances in a meaningful way. Such an approach also reduces the budget available for the subsequent active learning. Luckily, the enzymes domain has seen decades of annotation guided not by active learning but by biological and medical interest. This also made it possible to experiment (in Chapter 4) with the best algorithm ( knn) before attempting active learning. Also, guided learning has shown promise in this analysis and could be used to enrich the training set with types of instances not yet widely annotated.

**Batch-mode active learning** Most active learning publications make assumptions that may impose limitations in real world learning. For example, the assumption that only one oracle exists, and that it is always right. Most active learning algorithms provide only one query instance at a time. The active learning methods explored in Chapter 5 are, unfortunately, no exception. If several annotators are available, they will have to wait until the previous annotator has classified a protein and the model has been refreshed. This generates a tight bottleneck in productivity, with all-annotators-bar-one inactive, if we assume full time employment of annotators. On the other hand, this setting might still suit a set of free-lance or volunteer part-time curators, especially if they do not follow this methodology full time, but instead alternate between serving labels to an active learning method and working on other tasks.

In general, a distributed, parallel curation environment would call for a batch-based approach. However, the problem of obtaining an optimal *group* of instances is hard. In general, the *worst* approach is to simply pick the *n* most uncertain instances, as they could be strictly related. A better approach would be to sample instances randomly, to forcibly introduce some diversity among the picked instances. Even better are approaches that try to build a set of diverse (distant) instances, which was implemented using Support Vector Machines by [Brin 03, Xu 07] and for Logistic Regression by [Hoi 06, Guo 08].

In this thesis, guided learning methods have been explored instead, as a simple alternative that allows more parallelisation. For example, ordering InterPro sets by frequency means that a high number of curators can be deployed on the most protein rich sets at the same time. A measure of how parallel curation is in closed systems would be of interest to extrapolate how many curators could be needed to annotate all enzymes. If wikis start to be more widely used for curation though, they could introduce different levels of parallelisation to the task.

**Noisy oracles** Another strong assumption is that the oracle is always right in selecting labels. In a distributed environment there could exist different oracles with different expertise and error levels. This may entail an extension of the curation scenario, where the active learning emits some requests to de-noise already annotated instances that look like outliers, using a ”second pass” annotation by another oracle.

De-noising would fit a wiki-style scenario of annotation well. Unfortunately, data about individual annotators is not available for the main metabolic databases, which makes it difficult to evaluate and model individual error rates. Were the data available, this scenario could be modelled with WikiSim in the style described in Chapter 3.

**Cost of labelling** In our scenario, the cost of labelling an instance is not equal, with some proteins being less studied or having fewer published results due to funding bias, less scientific or medical interest, or experimental difficulties (for example limited expression in a manageable host or crystallisation difficulties). It would be interesting to account for the cost of annotation by incorporating an estimate of the annotation cost for each enzyme. In practice, this is likely to be very difficult for unknown proteins, and for labelled proteins not even the data providers may have data about *how long* it took to annotate an individual protein (as curators might not accurately track the exact time spent on each protein entry). If the data were available, it would be interesting to investigate simple measures (for example, number of existing proteins in the same family, or published articles) that correlate with the time needed to annotate an enzyme. Also, time might not be a good proxy for economic cost, as some expert annotators may need less time to annotate an instance, but they may also be paid more.

**Stopping criteria** Also, when should active curation stop? Various criteria can be devised, for example, based on accuracy. [Sett 09] however suggests that the stopping criterion in real life is usually economic – the cost of hiring annotators – or due to external factors. In our case, evaluation has been performed on already annotated instances from Swiss-Prot, but in real life, if the instances were to be drawn from TrEMBL or new sequencing projects, there could be lack of literature or knowledge about a certain protein. Thus, it might not be possible to meet the active learner request every time. Again, a problem that might be lessened by using guided curation methods that allow a curator to move on to a different protein as long as it has the same attribute set.

## 6.5 Conclusion

I would like to conclude with a summary of the applicability of the methods developed in this thesis to other domains. The curation model is potentially relevant to any structured curation process in any domain. The EnzML prediction schema is specific to enzyme function annotation, but the general indications on grouping biological data by taxonomic domain or the consequences of reducing sequence redundancy could be useful for other biological classifiers. The active and guided learning scenarios could also be of interest to other domains having scarce manual curation, pools of readily available unlabelled instances and where K-Nearest Neighbours is the algorithm of choice.

# Bibliography

[Adle 08] B. T. Adler, D. L. Alfaro, I. Pye, and V. Raman. “Measuring Author Contributions to the Wikipedia”. Tech. Rep., School of Engineering, University of California, May 2008.

[Ambe 09] J. Amberger, C. A. Bocchini, A. F. Scott, and A. Hamosh. “McKusick’s Online Mendelian Inheritance in Man (OMIM)”. *Nucleic Acids Res*, Vol. 37, No. Database issue, pp. D793–D796, Jan 2009.

[Ambe 11] J. Amberger, C. Bocchini, and A. Hamosh. “A new face and new challenges for Online Mendelian Inheritance in Man (OMIM$^{®}$)”. *Human Mutation*, Vol. 32, No. 5, pp. 564–567, 2011.

[Ande 06] C. Anderson. *The Long Tail*. Hyperion, 2006.

[Arak 09] A. K. Arakaki, Y. Huang, and J. Skolnick. “EFICAz2: enzyme function inference by a combined approach enhanced by machine learning”. *BMC Bioinformatics*, Vol. 10, p. 107, 2009.

[Arsh 07] B. I. Arshinoff, G. Suen, E. M. Just, S. M. Merchant, W. A. Kibbe, R. L. Chisholm, and R. D. Welch. “Xanthusbase: adapting wikipedia principles to a model organism database”. *Nucleic Acids Res*, Vol. 35, No. Database issue, pp. D422–D426, Jan 2007.

[Ashb 00] 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, M. A. Harris, D. P. Hill, L. Issel-Tarver, A. Kasarskis, S. Lewis, J. C. Matese, J. E. Richardson, M. Ringwald, G. M. Rubin, and G. Sherlock. “Gene Ontology: tool for the unification of biology”. *Nat Genet*, Vol. 25, No. 1, pp. 25–29, May 2000.

[Asti 08] K. Astikainen, L. Holm, E. Pitḱonen, S. Szedmak, and J. Rousu. “Towards structured output prediction of enzyme function”. *BMC Proc*, Vol. 2 Suppl 4, p. S2, 2008.

[Atte 10a] J. Attenberg, P. Melville, and F. Provost. “A Unified Approach to Active Dual Supervision for Labeling Features and Examples”. In: J. Balcázar, F. Bonchi, A. Gionis, and M. Sebag, Eds., *Machine Learning and Knowledge Discovery in Databases*, pp. 40–55, Springer Berlin Heidelberg, 2010.

[Atte 10b] J. Attenberg and F. Provost. “Why label when you can search?: alternatives to active learning for applying human resources to build classification models under extreme class imbalance”. In: *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 423–432, ACM, New York, NY, USA, 2010.

[Atte 11] J. Attenberg and F. Provost. "Inactive learning?: difficulties employing active learning in practice". *SIGKDD Explor. Newsl.*, Vol. 12, pp. 36–41, March 2011.

[Auer 06] S. Auer, S. Dietzold, and T. Riechert. "OntoWiki A Tool for Social, Semantic Collaboration". In: *The Semantic Web - ISWC 2006*, pp. 736–749, Springer Berlin / Heidelberg, 2006.

[Baru 06] Z. Barutcuoglu, R. E. Schapire, and O. G. Troyanskaya. "Hierarchical multi-label prediction of gene function". *Bioinformatics*, Vol. 22, No. 7, pp. 830–836, Apr 2006.

[Baum 07] W. A. Baumgartner, K. B. Cohen, L. M. Fox, G. Acquaah-Mensah, and L. Hunter. "Manual curation is not sufficient for annotation of genomic databases". *Bioinformatics*, Vol. 23, No. 13, pp. i41–i48, Jul 2007.

[Bent 09] R. A. Bentley, P. Ormerod, and M. Batty. "An evolutionary model of long tailed distributions in the social sciences". *ArXiv e-prints*, March 2009.

[Bioc 99] I.-I. C. on Biochemical Nomenclature. "IUPAC-IUBMB Joint Commission on Biochemical Nomenclature (JCBN) and Nomenclature Committee of IUBMB (NC-IUBMB), newsletter 1999". *Eur J Biochem*, Vol. 264, No. 2, pp. 607–609, Sep 1999.

[Bisb 04] F. A. Bisby, R. Froese, M. A. Ruggiero, and K. L. Wilson. "Species 2000 and ITIS catalogue of life, annual checklist 2004: indexing the world's known species". CD-ROM, 2004.

[Boul 06] M. Boulos, I. Maramba, and S. Wheeler. "Wikis, blogs and podcasts: a new generation of Web-based tools for virtual collaborative clinical practice and education". *BMC Medical Education*, Vol. 6, No. 1, p. 41, 2006.

[Brin 03] K. Brinker. "Incorporating diversity in active learning with support vector machines". In: *Proceedings of the International Conference on Machine Learning (ICML)*, p. 5966, AAAI Press, 2003.

[Brit 06] E. Britannica. "Encyclopaedia Britannica. Fatally flawed: refuting the recent study on encyclopedic accuracy by the journal Nature". 2006.

[Brya 05] S. L. Bryant, A. Forte, and A. Bruckman. "Becoming Wikipedian: transformation of participation in a collaborative online encyclopedia". In: *Proceedings of the 2005 international ACM SIGGROUP conference on Supporting group work*, pp. 1–10, ACM, New York, NY, USA, 2005.

[Burg 12] S. Burge, E. Kelly, D. Lonsdale, P. Mutowo-Muellenet, C. McAnulla, A. Mitchell, A. Sangrador-Vegas, S.-Y. Yong, N. Mulder, and S. Hunter. "Manual GO annotation of predictive protein signatures: the InterPro approach to GO curation". *Database (Oxford)*, Vol. 2012, p. bar068, 2012.

[Cai 04] C. Cai, L. Han, Z. Ji, and Y. Chen. "Enzyme family classification by support vector machines". *Proteins: Structure, Function, and Bioinformatics*, Vol. 55, pp. 66–76, 2004.

[Camo 04] E. Camon, M. Magrane, D. Barrell, V. Lee, E. Dimmer, J. Maslen, D. Binns, N. Harte, R. Lopez, and R. Apweiler. "The Gene Ontology Annotation (GOA)

Database: sharing knowledge in Uniprot with Gene Ontology". *Nucleic Acids Research*, Vol. 32, No. suppl 1, pp. D262–D266, 2004.

[Capo 06] A. Capocci, V. D. P. Servedio, F. Colaiori, L. S. Buriol, D. Donato, S. Leonardi, and G. Caldarelli. "Preferential attachment in the growth of social networks: the internet encyclopedia Wikipedia". *Phys Rev E Stat Nonlin Soft Matter Phys*, Vol. 74, No. 3 Pt 2, p. 036116, Sep 2006.

[Clar 02] A. Clare and R. D. King. "Machine learning of functional class from phenotype data". *Bioinformatics*, Vol. 18, No. 1, pp. 160–166, Jan 2002.

[Clau 03] C. Claudel-Renard, C. Chevalet, T. Faraut, and D. Kahn. "Enzyme-specific profiles for genome annotation: PRIAM". *Nucleic Acids Res*, Vol. 31, No. 22, pp. 6633–6639, Nov 2003.

[Clau 08] K. A. Clauson, H. H. Polen, M. N. K. Boulos, and J. H. Dzenowagis. "Scope, Completeness, and Accuracy of Drug Information in Wikipedia". *The Annals of Pharmacotherapy*, Vol. 42, No. 12, pp. 1814–1821, 2008.

[Clau 09] A. Clauset, C. R. Shalizi, and M. E. J. Newman. "Power-law distributions in empirical data. arXiv:0706.1062v2 [physics.data-an]". arXiv:0706.1062v2 [physics.data-an], Feb 2009.

[Coch 09] G. Cochrane, R. Akhtar, J. Bonfield, L. Bower, F. Demiralp, N. Faruque, R. Gibson, G. Hoad, T. Hubbard, C. Hunter, M. Jang, S. Juhos, R. Leinonen, S. Leonard, Q. Lin, R. Lopez, D. Lorenc, H. McWilliam, G. Mukherjee, S. Plaister, R. Radhakrishnan, S. Robinson, S. Sobhany, P. T. Hoopen, R. Vaughan, V. Zalunin, and E. Birney. "Petabyte-scale innovations at the European Nucleotide Archive". *Nucleic Acids Res*, Vol. 37, No. Database issue, pp. D19–D25, Jan 2009.

[Cons 10] U. Consortium. "The Universal Protein Resource (UniProt) in 2010". *Nucleic Acids Res*, Vol. 38, No. Database issue, pp. D142–D148, Jan 2010.

[Cons 11] T. U. Consortium. "Ongoing and future developments at the Universal Protein Resource". *Nucleic Acids Research*, Vol. 39, No. suppl 1, pp. D214–D219, 2011.

[Cran 08] D. Crandall, D. Cosley, D. Huttenlocher, J. Kleinberg, and S. Suri. "Feedback effects between similarity and social influence in online communities". In: *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 160–168, ACM, New York, NY, USA, 2008.

[Crof 11] D. Croft, G. O'Kelly, G. Wu, R. Haw, M. Gillespie, L. Matthews, M. Caudy, P. Garapati, G. Gopinath, B. Jassal, S. Jupe, I. Kalatskaya, S. Mahajan, B. May, N. Ndegwa, E. Schmidt, V. Shamovsky, C. Yung, E. Birney, H. Hermjakob, P. D'Eustachio, and L. Stein. "Reactome: a database of reactions, pathways and biological processes". *Nucleic Acids Res*, Vol. 39, No. Database issue, pp. D691–D697, Jan 2011.

[Dahl 02] K. D. Dahlquist, N. Salomonis, K. Vranizan, S. C. Lawlor, and B. R. Conklin. "GenMAPP, a new tool for viewing and analyzing microarray data on biological pathways". *Nat Genet*, Vol. 31, No. 1, pp. 19–20, May 2002.

[De F 08] L. De Ferrari, S. Aitken, J. van Hemert, and I. Goryanin. “WikiSim: simulating knowledge collection and curation in structured wikis”. In: *WikiSym ’08: Proceedings of the 4th International Symposium on Wikis*, pp. 1–2, ACM, New York, NY, USA, 2008.

[De F 09] L. De Ferrari, S. Aitken, J. van Hemert, and I. Goryanin. “A model of social collaboration in Molecular Biology knowledge bases”. In: B. Edmonds and N. Gilbert, Eds., *Proceedings of the 6th Conference of the European Social Simulation Association (ESSA’09)*, p. 47, European Social Simulation Association, European Social Simulation Association, 2009.

[De F 10] L. De Ferrari, S. Aitken, J. van Hemert, and I. Goryanin. “Multi-label prediction of enzyme classes using InterPro signatures”. In: *Machine Learning for Systems Biology Workshop (International Conference on Systems Biology)*, 2010.

[De F 12a] L. De Ferrari, S. Aitken, and J. Mitchell. “Active and guided learning for enzyme function prediction”. In: *11th European Conference on Computational Biology*, 2012.

[De F 12b] L. De Ferrari, S. Aitken, J. van Hemert, and I. Goryanin. “EnzML: Multi-label prediction of enzyme classes using InterPro signatures”. *BMC Bioinformatics*, Vol. 13, p. 61, 2012.

[Desa 11] D. K. Desai, S. Nandi, P. K. Srivastava, and A. M. Lynn. “ModEnzA: Accurate Identification of Metabolic Enzymes Using Function Specific Profile HMMs with Optimised Discrimination Threshold and Modified Emission Probabilities”. *Adv Bioinformatics*, Vol. 2011, p. 743782, 2011.

[DEus 11] P. DÉustachio. “Reactome Knowledgebase of Human Biological Pathways and Processes”. In: C. H. Wu and C. Chen, Eds., *Bioinformatics for Comparative Proteomics*, pp. 49–61, Humana Press, 2011. 10.1007/978-1-60761-977-2_4.

[Donm 08] P. Donmez and J. G. Carbonell. “Proactive learning: cost-sensitive active learning with multiple imperfect oracles”. In: *Proceedings of the 17th ACM conference on Information and knowledge management*, pp. 619–628, ACM, New York, NY, USA, 2008.

[Ecol 11] EcoliWiki. “EcoliWiki, EcoliHub’s subsystem for community annotation of E. coli K-12 resources. Last accessed Sep 2011.”. http://ecoliwiki.net/, 2011.

[Egel 10] V. Egelhofer, I. Schomburg, and D. Schomburg. “Automatic assignment of EC numbers”. *PLoS Comput Biol*, Vol. 6, No. 1, p. e1000661, 2010.

[Erwi 91] T. L. Erwin. “How Many Species Are There?: Revisited”. *Conservation Biology*, Vol. 5, No. 3, pp. 330–333, 1991.

[Fuji 98] A. Fujii, T. Tokunaga, K. Inui, and H. Tanaka. “Selective sampling for example-based word sense disambiguation”. *Comput. Linguist.*, Vol. 24, pp. 573–597, December 1998.

[Gast 03] E. Gasteiger, A. Gattiker, C. Hoogland, I. Ivanyi, R. D. Appel, and A. Bairoch. “ExPASy: The proteomics server for in-depth protein knowledge and analysis”. *Nucleic Acids Res*, Vol. 31, No. 13, pp. 3784–3788, Jul 2003.

[Gile 05] J. Giles. “Internet encyclopaedias go head to head”. *Nature*, Vol. 438, No. 7070, pp. 900–901, Dec. 2005.

[Guo 08] Y. Guo and D. Schuurmans. “Discriminative batch mode active learning”. In: *Advances in Neural Information Processing Systems (NIPS)*, p. 593600, MIT Press, Cambridge, MA, 2008.

[Hamo 02] A. Hamosh, A. F. Scott, J. Amberger, C. Bocchini, D. Valle, and V. A. McKusick. “Online Mendelian Inheritance in Man (OMIM), a knowledgebase of human genes and genetic disorders”. *Nucleic Acids Res*, Vol. 30, No. 1, pp. 52–55, Jan 2002.

[Haus 89] D. Haussler. “Learning Conjunctive Concepts in Structural Domains”. *Mach. Learn.*, Vol. 4, pp. 7–40, October 1989.

[Hing 08] P. Hingamp, C. Brochier, E. Talla, D. Gautheret, D. Thieffry, and C. Herrmann. “Metagenome Annotation Using a Distributed Grid of Undergraduate Students”. *PLoS Biol*, Vol. 6, No. 11, p. e296, 11 2008.

[Hoi 06] R. Hoi, SCH ad Jin, J. Zhu, and M. Lyu. “Batch mode active learning and its application to medical image classification”. In: *Proceedings of the International Conference on Machine Learning (ICML)*, p. 417424, ACM Press, 2006.

[Howe 08] D. Howe, M. Costanzo, P. Fey, T. Gojobori, L. Hannick, W. Hide, D. P. Hill, R. Kania, M. Schaeffer, S. S. Pierre, S. Twigger, O. White, and S. Y. Rhee. “Big data: The future of biocuration”. *Nature*, Vol. 455, No. 7209, pp. 47–50, Sep 2008.

[Hunt 09] S. Hunter, R. Apweiler, T. K. Attwood, A. Bairoch, A. Bateman, D. Binns, P. Bork, U. Das, L. Daugherty, L. Duquenne, R. D. Finn, J. Gough, D. Haft, N. Hulo, D. Kahn, E. Kelly, A. Laugraud, I. Letunic, D. Lonsdale, R. Lopez, M. Madera, J. Maslen, C. McAnulla, J. McDowall, J. Mistry, A. Mitchell, N. Mulder, D. Natale, C. Orengo, A. F. Quinn, J. D. Selengut, C. J. A. Sigrist, M. Thimma, P. D. Thomas, F. Valentin, D. Wilson, C. H. Wu, and C. Yeats. “InterPro: the integrative protein signature database”. *Nucleic Acids Res*, Vol. 37, No. Database issue, pp. D211–D215, Jan 2009.

[Huss 08] J. W. Huss, C. Orozco, J. Goodale, C. Wu, S. Batalov, T. J. Vickers, F. Valafar, and A. I. Su. “A gene wiki for community annotation of gene function”. *PLoS Biol*, Vol. 6, No. 7, p. e175, Jul 2008.

[Huss 10] J. W. Huss, P. Lindenbaum, M. Martone, D. Roberts, A. Pizarro, F. Valafar, J. B. Hogenesch, and A. I. Su. “The Gene Wiki: community intelligence applied to human gene annotation”. *Nucleic Acids Res*, Vol. 38, No. Database issue, pp. D633–D639, Jan 2010.

[Jack 09] C. Jackson, R. F. Murphy, and J. Kovacevi. “Intelligent acquisition and learning of fluorescence microscope data models”. *IEEE Trans Image Process*, Vol. 18, No. 9, pp. 2071–2084, Sep 2009.

[Jain 09] P. Jain and A. Kapoor. “Active learning for large multi-class problems”. *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, Vol. 0, pp. 762–769, 2009.

[Kane 10] M. Kanehisa, S. Goto, M. Furumichi, M. Tanabe, and M. Hirakawa. “KEGG for representation and analysis of molecular networks involving diseases and drugs”. *Nucleic Acids Res*, Vol. 38, No. Database issue, pp. D355–D360, Jan 2010.

[Kese 11] I. M. Keseler, J. Collado-Vides, A. Santos-Zavaleta, M. Peralta-Gil, S. Gama-Castro, L. Muiz-Rascado, C. Bonavides-Martinez, S. Paley, M. Krummenacker, T. Altman, P. Kaipa, A. Spaulding, J. Pacheco, M. Latendresse, C. Fulcher, M. Sarker, A. G. Shearer, A. Mackie, I. Paulsen, R. P. Gunsalus, and P. D. Karp. "EcoCyc: a comprehensive database of Escherichia coli biology.". *Nucleic Acids Res*, Vol. 39, No. Database issue, pp. D583–D590, Jan 2011.

[Kitt 07] A. Kittur, E. Chi, B. A. P. nad B. Suh, and T. Mytkowicz. "Power of the few vs. wisdom of the crowd: Wikipedia and the rise of the bourgeoisie.". In: *25th Annual ACM Conference on Human Factors in Computing Systems (CHI 2007)*, 2007.

[Koeh 06] J. Koehler, J. Baumbach, J. Taubert, M. Specht, A. Skusa, A. Rüegg, C. Rawlings, P. Verrier, and S. Philippi. "Graph-based analysis and visualization of experimental results with ONDEX.". *Bioinformatics*, Vol. 22, No. 11, pp. 1383–1390, Jun 2006.

[Kote 12] M. Kotera, M. Hirakawa, T. Tokimatsu, S. Goto, and M. Kanehisa. "The KEGG databases and tools facilitating omics analysis: latest developments involving human diseases and pharmaceuticals.". *Methods Mol Biol*, Vol. 802, pp. 19–39, 2012.

[Krie 09] M. Krieger, E. M. Stark, and S. R. Klemmer. "Coordinating tasks on the commons: designing for personal goals, expertise and serendipity". In: *Proceedings of the 27th international conference on Human factors in computing systems*, pp. 1485–1494, ACM, New York, NY, USA, 2009.

[Krot 06] M. Krötzsch, D. Vrandecic, and M. Völkel. "Semantic MediaWiki". In: I. Cruz, S. Decker, D. Allemang, C. Preist, D. Schwabe, P. Mika, M. Uschold, and L. Aroyo, Eds., *The Semantic Web - ISWC 2006*, pp. 935–942, Springer Berlin / Heidelberg, 2006. 10.1007/11926078_68.

[Lanc 04] G. R. G. Lanckriet, M. Deng, N. Cristianini, M. I. Jordan, and W. S. Noble. "Kernel-based data fusion and its application to protein function prediction in yeast.". *Pac Symp Biocomput*, Vol. –, pp. 300–311, 2004.

[Late 12] M. Latendresse, S. Paley, and P. D. Karp. "Browsing metabolic and regulatory networks with BioCyc.". *Methods Mol Biol*, Vol. 804, pp. 197–216, 2012.

[Lewi 94] D. D. Lewis and W. A. Gale. "A sequential algorithm for training text classifiers". In: *Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 3–12, Springer-Verlag New York, Inc., New York, NY, USA, 1994.

[Lind 04] M. Lindenbaum, S. Markovitch, and D. Rusakov. "Selective Sampling for Nearest Neighbor Classifiers". *Machine Learning*, Vol. 54, pp. 125–152, 2004. 10.1023/B:MACH.0000011805.60520.fe.

[Lyse 09] A. Lysenko, M. M. Hindle, J. Taubert, M. Saqi, and C. J. Rawlings. "Data integration for plant genomics–exemplars from the integration of Arabidopsis thaliana databases.". *Brief Bioinform*, Vol. 10, No. 6, pp. 676–693, Nov 2009.

[Magn 09] P. D. Magnus. "On Trusting Wikipedia". *Episteme*, Vol. 6, No. 1, pp. 74–90, 2009.

[Matt 09] L. Matthews and al. "Reactome knowledgebase of human biological pathways and processes.". *Nucleic Acids Res*, Vol. 37, No. Database issue, pp. D619–D622, Jan 2009.

[May 88] R. M. May. “How Many Species are There on Earth?”. *Science*, Vol. 241, No. 4872, pp. pp. 1441–1449, 1988.

[May 92] R. M. May. “How Many Species Inhabit the Earth?”. *Scientific American*, 1992.

[McIn 12] B. K. McIntosh, D. P. Renfro, G. S. Knapp, C. R. Lairikyengbam, N. M. Liles, L. Niu, A. M. Supak, A. Venkatraman, A. E. Zweifel, D. A. Siegele, and J. C. Hu. “EcoliWiki: a wiki-based community resource for Escherichia coli.”. *Nucleic Acids Res*, Vol. 40, No. Database issue, pp. D1270–D1277, Jan 2012.

[Mitz 04] M. Mitzenmacher. “A Brief History of Generative Models for Power Law and Lognormal Distributions”. *Internet Mathematics*, Vol. 1, No. 2, pp. 226–251, Jan. 2004.

[Moha 10] T. Mohamed, J. Carbonell, and M. Ganapathiraju. “Active learning for human protein-protein interaction prediction”. *BMC Bioinformatics*, Vol. 11, No. Suppl 1, p. S57, 2010.

[Mons 08] B. Mons and al. “Calling on a million minds for community annotation in WikiProteins.”. *Genome Biol*, Vol. 9, No. 5, p. R89, May 2008.

[Muld 07] N. Mulder and R. Apweiler. “InterPro and InterProScan: tools for protein sequence classification and comparison.”. *Methods Mol Biol*, Vol. 396, pp. 59–70, 2007.

[Niel 07] F. A. Nielsen. “Scientific citations in Wikipedia”. *CoRR*, Vol. abs/0705.2106, 2007.

[Orte 07a] F. Ortega and J. M. Gonzalez-Barahona. “Quantitative analysis of the wikipedia community of users”. In: *Proceedings of the 2007 international symposium on Wikis*, pp. 75–86, ACM, Montreal, Quebec, Canada, 2007.

[Orte 07b] F. Ortega, J. M. Gonzlez-Barahona, and G. Robles. “The top-ten wikipedias”. In: *Proceedings of the 2nd International Conference on Software and Data Technologies (ICSOFT-2007)*, 2007.

[Orte 08] F. Ortega, J. Gonzalez-Barahona, and G. Robles. “On the Inequality of Contributions to Wikipedia”. In: *Hawaii International Conference on System Sciences, Proceedings of the 41st Annual*, pp. 304–304, 2008.

[Osma 10] H. Osmanbeyoglu, J. Wehner, J. Carbonell, and M. Ganapathiraju. “Active machine learning for transmembrane helix prediction”. *BMC Bioinformatics*, Vol. 11, No. Suppl 1, p. S58, 2010.

[Pico 08] A. R. Pico, T. Kelder, M. P. van Iersel, K. Hanspers, B. R. Conklin, and C. Evelo. “WikiPathways: pathway editing for the people.”. *PLoS Biol*, Vol. 6, No. 7, p. e184, Jul 2008.

[Pitk 10] E. Pitkaenen, J. Rousu, and E. Ukkonen. “Computational methods for metabolic reconstruction.”. *Curr Opin Biotechnol*, Vol. 21, No. 1, pp. 70–77, Feb 2010.

[Prie 07] R. Priedhorsky, J. Chen, S. T. K. Lam, K. Panciera, L. Terveen, and J. Riedl. “Creating, destroying, and restoring value in wikipedia”. In: *Proceedings of the 2007 international ACM conference on Supporting group work*, pp. 259–268, ACM, New York, NY, USA, 2007.

[Rose 06] R. Rosenzweig. “Can History Be Open Source? Wikipedia and the Future of the Past”. *The Journal of American History*, Vol. 93, No. 1, pp. 117–146, 2006.

[Rose 11] P. W. Rose, B. Beran, C. Bi, W. F. Bluhm, D. Dimitropoulos, D. S. Goodsell, A. Prlic, M. Quesada, G. B. Quinn, J. D. Westbrook, J. Young, B. Yukich, C. Zardecki, H. M. Berman, and P. E. Bourne. “The RCSB Protein Data Bank: redesigned web site and web services”. *Nucleic Acids Research*, Vol. 39, No. suppl 1, pp. D392–D401, 2011.

[Roth 07] C. Roth. “Viable wikis: struggle for life in the wikisphere”. In: *Proceedings of the 2007 international symposium on Wikis*, pp. 119–124, ACM, New York, NY, USA, 2007.

[Roth 08] C. Roth, D. Taraborelli, and N. Gilbert. “Measuring wiki viability: an empirical assessment of the social dynamics of a large sample of wikis”. In: *Proceedings of the 4th International Symposium on Wikis*, pp. 27:1–27:5, ACM, New York, NY, USA, 2008.

[Salo 07] N. Salomonis, K. Hanspers, A. C. Zambon, K. Vranizan, S. C. Lawlor, K. D. Dahlquist, S. W. Doniger, J. Stuart, B. R. Conklin, and A. R. Pico. “GenMAPP 2: new features and resources for pathway analysis.”. *BMC Bioinformatics*, Vol. 8, p. 217, 2007.

[Salz 07] S. L. Salzberg. “Genome re-annotation: a wiki solution?”. *Genome Biol*, Vol. 8, No. 1, p. 102, 2007.

[Sche 07] A. Schein and L. Ungar. “Active learning for logistic regression: an evaluation”. *Machine Learning*, Vol. 68, pp. 235–265, 2007. 10.1007/s10994-007-5019-5.

[Schi 10] L. Schietgat, C. Vens, J. Struyf, H. Blockeel, D. Kocev, and S. Dzeroski. “Predicting gene function using hierarchical multi-label decision tree ensembles.”. *BMC Bioinformatics*, Vol. 11, p. 2, 2010.

[Sett 08a] B. Settles. *Curious Machines: Active Learning with Structured Instances*. PhD thesis, University of Wisconsin Madison, 2008.

[Sett 08b] B. Settles and M. Craven. “fAn analysis of active learning strategies for sequence labeling tasks”. In: *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 1070–1079, Association for Computational Linguistics, Stroudsburg, PA, USA, 2008.

[Sett 08c] B. Settles, M. Craven, and S. Ray. “Multiple-instance active learning”. In: *In Advances in Neural Information Processing Systems (NIPS*, pp. 1289–1296, MIT Press, 2008.

[Sett 09] B. Settles. “Active learning literature survey.”. Computer Sciences Technical Report 1648, University of Wisconsin Madison., 2009.

[Seun 92] H. S. Seung, M. Opper, and H. Sompolinsky. “Query by committee”. In: *Proceedings of the fifth annual workshop on Computational learning theory*, pp. 287–294, ACM, New York, NY, USA, 1992.

[Sing 02] P. Singh, T. Lin, E. Mueller, G. Lim, T. Perkins, and W. Li Zhu. “Open Mind Common Sense: Knowledge Acquisition from the General Public”. In:

R. Meersman and Z. Tari, Eds., *On the Move to Meaningful Internet Systems 2002: CoopIS, DOA, and ODBASE*, pp. 1223–1237, Springer Berlin / Heidelberg, 2002. 10.1007/3-540-36124-3_77.

[Soko 09] M. Sokolova and G. Lapalme. “A systematic analysis of performance measures for classification tasks”. *Information Processing &amp; Management*, Vol. 45, No. 4, pp. 427 – 437, 2009.

[Spyr 08] E. Spyromitros, G. Tsoumakas, and I. Vlahavas. “An Empirical Study of Lazy Multilabel Classification Algorithms”. *Artificial Intelligence: Theories, Models and Applications*, pp. 401–406, 2008.

[Steh 10] H. Stehr, J. M. Duarte, M. Lappe, J. Bhak, and D. M. Bolser. “PDBWiki: added value through community annotation of the Protein Data Bank”. *Database*, Vol. 2010, 2010.

[Stor 07] N. E. Stork. “Biodiversity: world of insects.”. *Nature*, Vol. 448, No. 7154, pp. 657–658, Aug 2007.

[Stor 93] N. E. Stork. “How many species are there?”. *Biodiversity and Conservation*, Vol. 2, pp. 215–232, 1993. 10.1007/BF00056669.

[Stuc 09] J. Stuckman and J. Purtilo. “Measuring the wikisphere”. In: *Proceedings of the 5th International Symposium on Wikis and Open Collaboration*, pp. 11:1–11:8, ACM, New York, NY, USA, 2009.

[Stvi 05] B. Stvilia, M. B. Twidale, L. C. Smith, and L. Gasser. “Assessing information quality of a community-based encyclopedia.”. In: M. M. Cambridge, Ed., *In: F. Naumann, M. Gertz, S. Mednick (Eds.), Proceedings of the International Conference on Information Quality - ICIQ*, pp. pp. 442–454, 2005.

[Stvi 08] B. Stvilia, M. B. Twidale, L. C. Smith, and L. Gasser. “Information quality work organization in wikipedia”. *Journal of the American Society for Information Science and Technology*, Vol. 59, No. 6, pp. 983–1001, 2008.

[Suze 07] B. E. Suzek, H. Huang, P. McGarvey, R. Mazumder, and C. H. Wu. “UniRef: comprehensive and non-redundant UniProt reference clusters”. *Bioinformatics*, Vol. 23, No. 10, pp. 1282–1288, 2007.

[Tetk 08] I. V. Tetko, I. V. Rodchenkov, M. C. Walter, T. Rattei, and H.-W. Mewes. “Beyond the ’best’ match: machine learning annotation of protein sequences by integration of different sources of information.”. *Bioinformatics*, Vol. 24, No. 5, pp. 621–628, Mar 2008.

[Tian 04] W. Tian, A. K. Arakaki, and J. Skolnick. “EFICAz: a comprehensive approach for accurate genome-scale enzyme function inference.”. *Nucleic Acids Res*, Vol. 32, No. 21, pp. 6226–6239, 2004.

[Todd 02] A. E. Todd, C. A. Orengo, and J. M. Thornton. “Sequence and Structural Differences between Enzyme and Nonenzyme Homologs”. *Structure*, Vol. 10, No. 10, pp. 1435 – 1451, 2002.

[Torr 10] A. Torralba, B. Russell, and J. Yuen. “LabelMe: Online Image Annotation and Applications”. *Proceedings of the IEEE*, Vol. 98, No. 8, pp. 1467 –1484, aug. 2010.

[Troi 08] K. G. Troitzsch. “Simulating collaborative writing: software agents produce a wikipedia.”. In: F. Squazzoni, Ed., *The Fifth Conference of the European Social Simulation Association, Brescia*, September 2008.

[Tsou 07] G. Tsoumakas and I. Vlahavas. “Random k -Labelsets: An Ensemble Method for Multilabel Classification”. *Machine Learning: ECML 2007*, pp. 406–417, 2007.

[Tsou 10] G. Tsoumakas, I. Katakis, and I. Vlahavas. *Mining Multi-label Data. In: Data Mining and Knowledge Discovery Handbook.* Springer, 2010.

[Tsur 08] Y. Tsuruoka, J. Tsujii, and S. Ananiadou. “Accelerating the annotation of sparse named entities by dynamic sentence selection.”. *BMC Bioinformatics*, Vol. 9 Suppl 11, p. S8, 2008.

[Vale 08] G. Valentini and N. Cesa-Bianchi. “HCGene: a software tool to support the hierarchical classification of genes.”. *Bioinformatics*, Vol. 24, No. 5, pp. 729–731, Mar 2008.

[Vieg 04] F. B. Viegas, M. Wattenberg, and K. Dave. “Studying cooperation and conflict between authors with history flow visualizations”. In: *Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 575–582, ACM, Vienna, Austria, 2004.

[Voss 05] J. Voss. “Measuring Wikipedia”. In: *Proceedings of the ISSI*, 2005.

[Wall 10] B. C. Wallace, T. A. Trikalinos, J. Lau, C. Brodley, and C. H. Schmid. “Semi-automated screening of biomedical citations for systematic reviews.”. *BMC Bioinformatics*, Vol. 11, p. 55, 2010.

[Wang 06] K. Wang. “Gene-function wiki would let biologists pool worldwide resources.”. *Nature*, Vol. 439, No. 7076, p. 534, Feb 2006.

[Wate 07] N. L. Waters. “Why you can’t cite Wikipedia in my class”. *Commun. ACM*, Vol. 50, pp. 15–17, September 2007.

[Wilk 07] D. M. Wilkinson and B. A. Huberman. “Assessing the Value of Coooperation in Wikipedia”. *ArXiv Computer Science e-prints*, Feb. 2007.

[Wist 87] G. J. Wistow, J. W. M. Mulders, and W. W. de Jong. “The enzyme lactate dehydrogenase as a structural protein in avian and crocodilian lenses”. *Nature*, Vol. 326, No. 6113, pp. 622–624, Apr. 1987.

[Xu 07] Z. Xu, K. Akella, and Y. Zhang. “Incorporating diversity and density in active learning for relevance feedback”. In: *Proceedings of the European Conference on IR Research (ECIR)*, p. 246257, Springer-Verlag, 2007.

[Xu 08] J. Xu, L. Yilmaz, and J. Zhang. “Agent simulation of collaborative knowledge processing in Wikipedia”. In: *Proceedings of the 2008 Spring simulation multiconference*, pp. 19–25, Society for Computer Simulation International, San Diego, CA, USA, 2008.