Estimating the Quality of Ontology-Based Annotations by Considering Evolutionary Changes

Anika Gross[1], Michael Hartung[1], Toralf Kirsten[1,2], and Erhard Rahm[1,3]

[1] Interdisciplinary Centre for Bioinformatics, University of Leipzig
[2] Institute for Medical Informatics, Statistics and Epidemiology, University of Leipzig
[3] Department of Computer Science, University of Leipzig
{gross,hartung,tkirsten}@izbi.uni-leipzig.de, rahm@informatik.uni-leipzig.de

Abstract. Ontology-based annotations associate objects, such as genes and proteins, with well-defined ontology concepts to semantically and uniformly describe object properties. Such annotation mappings are utilized in different applications and analysis studies whose results strongly depend on the quality of the used annotations. To study the quality of annotations we propose a generic evaluation approach considering the annotation generation methods (provenance) as well as the evolution of ontologies, object sources, and annotations. Thus, it facilitates the identification of reliable annotations, e.g., for use in analysis applications. We evaluate our approach for functional protein annotations in Ensembl and Swiss-Prot using the Gene Ontology.

Keywords: annotation, evolution, quality.

1 Introduction

Ontologies and their application have become increasingly important especially in the life sciences. Typically, they are used to semantically describe or annotate properties of real world objects, such as genes and proteins. The associations between object descriptions and the elements (concepts) of an ontology form a so-called *annotation mapping*. For instance, the protein objects of Ensembl [11] and Swiss-Prot [3] are associated with concepts of the popular Gene Ontology [9] to describe the molecular functions and biological processes in which the proteins are involved. Annotation mappings are utilized in different analysis scenarios and applications. These include functional profiling of large datasets such as gene expression microarrays (e.g., [1,4]), network reconstruction and retrieval [7], or instance-based ontology matching [13].

Computed results of these applications significantly depend on which annotations are used and hence rely on a good quality of the annotations, e.g., with respect to their correctness and completeness. A particularly important quality aspect is the stability of annotations since major changes in the annotation mappings may substantially influence or even invalidate earlier findings. This is potentially a major issue since annotation mappings change frequently, e.g., due to changes (additions, deletions,

N.W. Paton, P. Missier, and C. Hedeler (Eds.): DILS 2009, LNBI 5647, pp. 71–87, 2009.

Instance ID	Concept ID	v_{48}	v_{49}	v_{50}	v_{51}	v_{52}
ENSP00000344151	GO:0015808 (L-alanine transport)	IDA	IDA	IDA	IDA	IDA
ENSP00000230480	GO:0005615 (extracellular space)	TAS	TAS	IDA	TAS	IEA
ENSP00000352999	GO:0006915 (apoptosis)	IDA	-	-	-	IDA

Fig. 1. Evolution of functional protein annotations in Ensembl versions (v_{48}-v_{52} = Dec.2007-Dec.2008)

modifications) in the underlying ontologies [10], objects and annotation associations. Furthermore, annotation quality is influenced by the method that has been used to create the annotation because it likely affects how biologically founded or reliable an annotation is. The relevance of the creation method is underlined by the increasing use of predefined *evidence codes* (EC) to classify functional annotations based on the Gene Ontology [8]. These evidence codes allow a distinction of whether annotations are experimentally founded, are based on author or curator statements or generated by automatic algorithms, e.g., data mining techniques or homology mappings. The evidence codes represent *provenance* information (sometimes also called lineage[1] [2,5]) that can be utilized by analysis applications to focus on specific annotation sets, e.g., manually curated or automatically generated annotations.

For illustration, Figure 1 shows the evolution of selected functional protein annotations in five succeeding Ensembl versions (v_{48}-v_{52}). The first annotation (ENSP00000344151, GO:0015808) was continuously available with unchanged evidence code (IDA, *inferred from direct assay*) indicating a stable annotation. Conversely, the evidence code of the second annotation for protein ENSP00000230480 has been changed from *traceable author statement* (TAS) over IDA to *inferred from electronic annotation* (IEA). Such a frequent revision of the provenance information indicates reduced reliability of the annotation. Furthermore, the last annotation (Figure 1, line 3) was temporarily absent also indicating a reduced stability.

So far, the quality of annotation mappings w.r.t. their stability and provenance information is largely unexplored despite their potential importance for many analysis applications. We therefore present and evaluate a general approach to analyze annotation mappings by taking their evolution and evidence information into account. To that end we first propose an evolution model for annotation mappings including change operators and quality measures (Section 2). The model captures ontology, instance and quality changes w.r.t. annotation changes. Based on the evolution model, we propose evolution-based quality measures to identify reliable annotations (Section 3). Finally, we evaluate our evolution model by comparatively analyzing the annotation evolution in two large life science annotation sources, namely Ensembl and Swiss-Prot (Section 4). In particular, we study typical annotation changes and classify current annotations by applying the proposed assessment method. Section 5 discusses related work before we conclude.

The analysis results and the proposed assessment method for annotations are expected to be valuable for users and applications of life science annotations. In particular, algorithms may utilize information of annotation history and annotation quality to derive more robust / reliable results.

[1] We further use the term provenance to determine the original source of data.

2 Annotation Models

The stability of annotation mappings is affected by the changes in the involved instance (object) sources, ontologies and object-ontology associations. In the following we first introduce our model of annotation mappings including models for instance sources, ontologies and annotation quality. We will assume that annotations (object-ontology associations) include several quality indicators whose values may be taken from predefined quality taxonomies. In Section 2.2 we will introduce our evolution model including change operators for instances, ontologies and annotations. Furthermore, measures are proposed in order to quantify the evolution of annotations.

2.1 Annotation Mapping and Quality Models

As usual in life sciences, we assume that ontologies and instance sources are versioned so that a specific version reflects a stable data snapshot from a specific point in time. The versioning scheme is assumed to be linear, i.e., a particular version v_i has exactly one successor version v_{i+1} and one predecessor version v_{i-1}. The latest (first) version form exceptions since no successor (predecessor) versions are available.

As illustrated in Figure 2, annotation mappings interrelate a specific version of an instance source with a specific version of an ontology. Furthermore, annotation mappings can refer to common quality taxonomies to specify the quality of individual annotation associations by different criteria, e.g., provenance or stability. Before we define the details of annotation mappings we briefly introduce our models for instance sources and ontologies which are based on [10].

An instance source of version v is denoted by $I_v = (I, t)$ consisting of a set of instances $I = \{i_1, \ldots, i_n\}$ and a release timestamp t. An instance item i of I is described by a set of attributes, e.g., name or current status. A special attribute called *accession number* identifies instance items unambiguously. Accession numbers are utilized to reference instance items within annotation mappings.

An ontology $ON_v = (C, R, t)$ of version number v and release timestamp t consists of concepts $C = \{c_1, \ldots, c_n\}$ and relationships $R = \{r_1, \ldots, r_m\}$. A concept $c \in C$

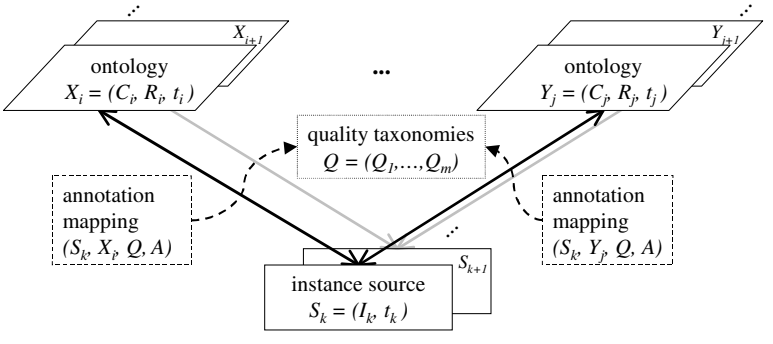


Fig. 2. Model of instance sources, ontologies and annotation mappings with versioning and quality

comprises attributes for its detailed description, e.g., synonyms or a definition. An *accession number* is utilized for unambiguous identification of concepts and the *obsolete* status signals whether a concept is active or not within the ontology. Furthermore, concepts can be interconnected by directed relationships $r = (c_1, c_2) \in R$, e.g., is-a or part-of relationships. Overall, concepts C and relationships R form the graph structure of an ontology which is usually a directed acyclic graph (DAG) with *root concepts* (concepts of C that have no relationships to a super concept).

An annotation mapping $AM = (I_u, ON_v, Q, A)$ associates an instance source version I_u with an ontology version ON_v by a set of correspondences A. A single association or annotation $a \in A$ is denoted by $a = (i, c, \{q\})$, i.e., an instance item $i \in I_v$ is annotated with an ontology concept $c \in ON_v$ and a set of quality indicators (ratings) $\{q\}$. The quality indicators $\{q\}$ of annotations may be numerical values or come from predefined *quality taxonomies* $Q_1,\ldots,Q_m \in Q$. Quality taxonomies represent predefined criteria for uniform quality characterization, e.g., the evidence codes for provenance information or stability indicators. Note that for each quality taxonomy at most one quality indicator can be utilized in an annotation. Typically, the quality ratings of an annotation are specified when an annotation is first generated. However annotation ratings may be modified, as seen in the examples of Figure 1, e.g., when changed information about the annotation becomes available.

A quality taxonomy representing a particular quality criterion consists of a set of predefined quality terms $\{q_1, \ldots, q_n\}$ which may be arranged in an is-a-like hierarchy. In the general case, a quality term $q = (q', type)$ of name q is defined by a *type* and an optional super term q'. Every quality term has exactly one parent term, if no parent term exists, the quality term is assumed to be the root of the quality taxonomy. Quality terms can be of two different types: *instantiable* and *abstract*. While instantiable quality terms are applicable for rating an annotation, abstract ones are not utilized in annotations, i.e., they only act as aggregation nodes within the taxonomy. For our study, we assume that quality taxonomies remain unchanged.

We will utilize three different types of quality indicators to specify (1) *provenance type*, (2) *stability* and (3) *age* of annotations. First, for provenance information we utilize and analyze the existing Evidence Codes (EC) [8] for GO annotations which specify their generation method. Figure 3 shows the current *EC quality taxonomy* including different groups, in particular 'Manually assigned' (*man*), 'Automatically assigned' (*auto*) and 'Obsolete' (*obs*). Manually determined annotations are further

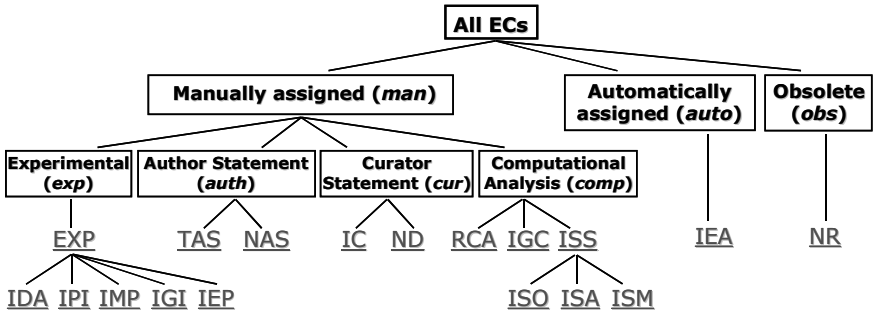


Fig. 3. Evidence Code Taxonomy

refined by the *exp*, *auth*, *cur* and *comp* groups. In contrast, *auto* annotations are unverified but have been generated by algorithms such as homology or keyword mappings. For stability and age, we do not directly use numerical values but map them into categorical terms of a quality taxonomy to simplify their use and evaluation. Our *stability quality taxonomy* consists of only two terms to differentiate *stable* and *unstable* annotations based on their evolution history. Our *age quality taxonomy* differentiates between *novel*, *middle* and *old* annotations. Hence, an automatically generated, stable and middle-aged annotation between instance item i and ontology concept c can be described by $a = (i,c,\{\text{IEA},stable,middle\})$. The introduced quality taxonomies will be used in our evaluation in Section 4. Note that the EC information is frequently available for GO annotations but has not yet been comparatively evaluated. Furthermore, to the best of our knowledge the stability and age of annotations has not yet been analyzed and utilized.

In life sciences, annotation mapping versioning usually follows the versioning scheme of the instance source, i.e., a new instance source version possibly includes changed annotations as well as referring to some (current or older) versions of the respective ontologies. On the other hand, a new ontology version is generally not released with a new version of annotation mappings. Furthermore, succeeding versions of an instance source may refer to the same ontology version.

2.2 Evolution Model

We extend the evolution model for ontologies and mappings of [10] which is limited to simple addition and deletion changes. In order to study evolution in annotations in more detail, we introduce new change types and consider quality changes in annotations as well as the influence of instance / ontology changes on annotations.

Figure 4 summarizes the possible change operations for instances, ontologies and annotations in a simple taxonomy. For instance sources (object ≙ instance item) and ontologies (object ≙ ontology concept), we distinguish between the following operations:

- *add:* addition of a new object
- *del:* deletion of an existing object
- *toObs:* marking an existing object as obsolete, i.e., the object becomes inactive
- *subs:* substitution of an existing object by a new object
- *merge:* merging of an object into an existing object

For annotations we differentiate between the following change operations based on the operations for instance sources and ontologies:

- *add:* addition of a new annotation
- del_{ann}: deletion of an existing annotation
- del_{ont}: deletion of an annotation caused by ontology concept change or delete
- del_{ins}: deletion of an annotation caused by instance item change or delete
- chg_{ont}: adaptation of an annotation caused by ontology concept change
- chg_{ins}: adaptation of an annotation caused by instance item change
- chg_{qual}: change of the quality indicator of an annotation

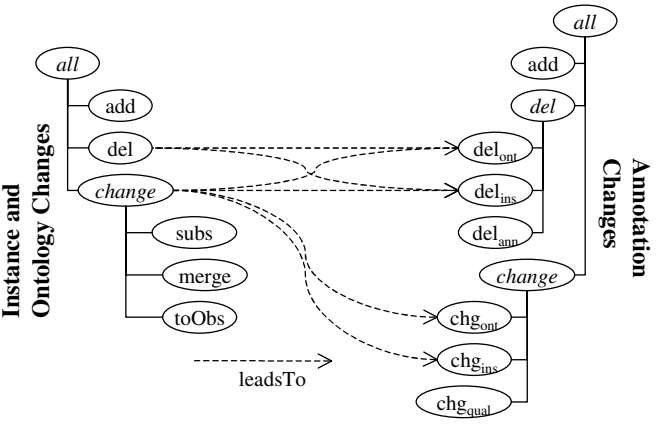
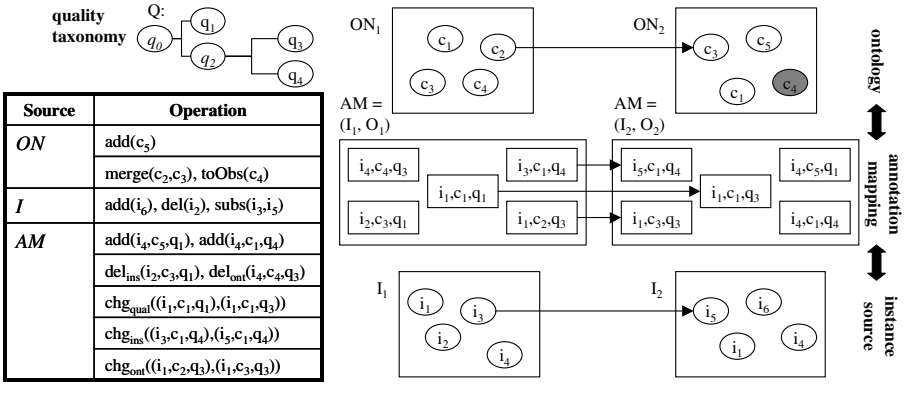


Fig. 4. Effects of instance and ontology changes on annotations

Several dependencies exist between instance/ontology changes and annotation changes (see *leadsTo* dependencies in Figure 4) leading to a corresponding propagation of changes when ontologies and instances evolve. Deletions of ontology concepts and instances always lead to the removal of dependent annotations (del_{ont}, del_{ins} changes). Furthermore, a change (*subs*, *merge*, *toObs*) of an instance item or ontology concept may cause the deletion or adaptation of dependent annotations as described with the del_{ins}, del_{ont}, chg_{ins} and chg_{ont} operations. Besides these changes quality changes (chg_{qual}), e.g., when an automatically generated annotation was later proved by an experiment, and conventional additions / deletions (*add*, del_{ann}) for annotations are distinguished.

Figure 5 illustrates the various change operators by a rather comprehensive example of annotation evolution. The example displays an evolution step between two versions for an instance source I ($I_1 \rightarrow I_2$), an ontology ON ($ON_1 \rightarrow ON_2$) and an annotation mapping AM ($(I_1,ON_1) \rightarrow (I_2,ON_2)$). The table on the left summarizes the change



Source	Operation
ON	add(c_5)
	merge(c_2,c_3), toObs(c_4)
I	add(i_6), del(i_2), subs(i_3,i_5)
AM	add(i_4,c_5,q_1), add(i_4,c_1,q_4)
	del_{ins}(i_2,c_3,q_1), del_{ont}(i_4,c_4,q_3)
	chg_{qual}((i_1,c_1,q_1),(i_1,c_1,q_3))
	chg_{ins}((i_3,c_1,q_4),(i_5,c_1,q_4))
	chg_{ont}((i_1,c_2,q_3),(i_1,c_3,q_3))

Fig. 5. Evolution example with possible change operations

operations resulting in the new versions for *I*, *ON* and *AM,* shown on the right of Figure 5. So the instance source as well as the ontology possess added (i_6, c_5) and deleted objects (i_2, c_4). For c_2 a merge into concept c_3 was performed and c_4 has become obsolete. Furthermore, i_3 was replaced by the new instance item i_5. As a result some annotations were adapted, e.g., (i_1,c_2) to (i_1,c_3) and (i_3,c_1) to (i_5,c_1), or deleted, e.g., (i_2,c_3) and (i_4,c_4). Moreover, (i_1,c_1) changed its quality from q_1 to q_3 in the new version. New annotations were also added: (i_4,c_5,q_1) and (i_4,c_1,q_4).

2.3 Measures to Quantify Annotation Evolution and Changes

For our evaluation, we will utilize several measures to quantitatively assess the evolution of life science annotations. In addition to some general cardinality and growth measures we want to specifically evaluate annotation changes such as the change propagations between instances/ontologies and annotations as well as changes in the quality of annotations.

By using quality-specific statistics we can quantify how annotations with different quality indicators evolve over time, e.g., to discover which quality groups (annotations with a particular quality q) changed heavily or remained almost stable in a period p under review. For these purposes, we use the following measures:

$\|A_{vi}\|$	number of annotations in version v_i of an annotation mapping
$\|A_{vi,q}\|$	number of annotations with quality q in version v_i
$\|A_{vi,q}\| / \|A_{vi}\|$	relative share of annotations with quality q to the overall number of annotations in version v_i
$Add_{vi,vj,q}$, $Del_{vi,vj,q}$, $Chg_{vi,vj,q}$	number of *added*, *deleted* or *changed* annotations with quality q between version v_i and v_j
$Add_{p,q}$, $Del_{p,q}$, $Chg_{p,q}$	number of *added*, *deleted* or *changed* annotations with quality q within an observation period p
$growth_{A,q,vi,vj} = \|A_{vj,q}\| / \|A_{vi,q}\|$	growth rate of annotations with quality q between version v_i and v_j

We further investigate the impact of instance/ontology changes on annotation changes. Since instance/ontology changes especially deletions, merges or substitutions affect changes in annotations we propose measures that assess these influences w.r.t. a version change ($v_i \rightarrow v_j$) or an observatio006E period (p):

Chg_{ont}, Chg_{ins}	number of annotations that have changed caused by a change of the referenced instance item or ontology concept
Chg_{qual}	number of annotations that changed their quality
Del_{ont}, Del_{ins}	number of annotations that have been deleted caused by a change or a deletion of the referenced instance item or ontology concept

3 Assessment of Annotation Stability

In this section we propose a method to assess the stability of annotations based on their evolution history and changes in quality indicators. To assess the evolution history without considering quality criteria, we define the history h of an annotation $a = (i,c)_n$ of version v_n :

$$h((i,c)_n) = ((i,c)_0, (i,c)_1, \ldots, (i,c)_n) \mid 0 \leq i < n: (i,c)_i \rightarrow (i,c)_{i+1}$$

So an annotation $(i,c)_{i+1}$ in v_{i+1} has evolved from $(i,c)_i$ in v_i, e.g., caused by an instance merge or substitution (see change taxonomy in Figure 5), or remained unchanged. The non-existence of an annotation in a version is denoted by a null value, e.g., after a deletion or before the first occurrence. The computation occurs with respect to all versions of a predefined observation period *p,* e.g., the last year. Given the history *h* for an annotation *a* we can determine different measures for its evolution within an observation period *p*.

First, the age of an annotation (in number of versions) is defined as

- $a_{age} = (n\text{-}fo)+1$

where *n* is the number of the current version (v_n) and *fo* denotes the number of the version (v_{fo}) in which the annotation occurs for the first time within *p*. In addition, we count the number of versions in *p* in which an annotation appeared ($a_{present}$). Note that the counts ignore all versions of the annotation mapping before the first occurrence of an annotation. Based on a_{age} and $a_{present}$ we define a simple *existence stability* measure that evaluates the relative existence of a single annotation *a*:

- $stab_{exis}(a) = a_{present} / a_{age}$

To evaluate quality changes of annotations within *p* we use an extended history h_Q of an annotation with respect to a quality indicator (e.g., provenance):

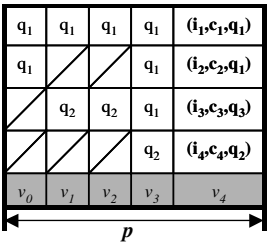
$$h_Q((i,c,q)_n) = ((i,c,q)_0, (i,c,q)_1, \ldots, (i,c,q)_n) \mid 0 \leq i < n: (i,c,q)_i \rightarrow (i,c,q)_{i+1}$$

The extended history h_Q incorporates the values of the considered quality indicator w.r.t. a particular quality taxonomy *Q*. Note that the consideration of quality changes in an annotation history may only be useful for some quality criteria. For instance, we will focus on provenance changes in our evaluation, e.g., when the evidence code of an annotation is modified due to new experimental findings. We count quality changes by determining the number of versions in the history of *a* where a quality change occurred ($a_{changed}$). Conversely, $a_{unchanged}$ specifies the number of versions without quality modification. Versions for which an annotation was temporarily missing are skipped in the change comparison of the quality indicator.

Utilizing the counts we define a stability measure for *quality stability* as well as a *combined stability* for a single annotation *a*:

- $stab_{qual}(a) = a_{unchanged} / (a_{unchanged}+a_{changed})$
- $stab_{comb}(a) = min (stab_{qual}(a), stab_{exis}(a))$

While $stab_{qual}$ assesses the frequency of quality changes of an annotation, the combined stability measure $stab_{comb}$ conservatively integrates $stab_{exis}$ and $stab_{qual}$ by calculating the minimum. Note that the proposed measures have a value range of [0,1]. Thereby, a low value signals instability. Perfect stability is achieved in case of 1, e.g., if an annotation is permanently present since its first occurrence (perfect existence stability) or possesses no quality changes (perfect quality stability). In our evaluation (Section 4) we will utilize these measures to classify annotations w.r.t. the two quality criteria *age* and *stability* discussed in Section 2.1. Particularly, we use a threshold criterion to map numerical stability values into corresponding terms of the stability taxonomy.



annotation a	a_{age}	$stab_{exis}$	$stab_{qual}$	$stab_{comb}$
(i_1,c_1,q_1)	*5*	5/5 = *1*	4/(4+0) = *1*	*1*
(i_2,c_2,q_1)	*5*	3/5 = *0.6*	2/(2+0) = *1*	*0.6*
(i_3,c_3,q_3)	*4*	4/4 = *1*	1/(1+2) = *0.33*	*0.33*
(i_4,c_4,q_2)	*2*	2/2 = *1*	1/(1+0) = *1*	*1*

Fig. 6. History and measure results of four example annotations

The example in Figure 6 illustrates the proposed measures for four annotations. An observation period with 5 versions of an annotation mapping (v_0-v_4) is considered. For each version the quality term of an annotation is displayed, an empty cell denotes the temporal non-existence of an annotation in the respective version. The four histories of (i_1,c_1,q_1), (i_2,c_2,q_1), (i_3,c_3,q_3) and (i_4,c_4,q_2) of version v_4 exhibit different evolution characteristics. Annotation (i_1,c_1,q_1) has been introduced in v_0 (i.e., $a_{age}=5$) and shows a perfect stability of 1 in $stab_{exis}$ as well as $stab_{qual}$ and thus also in $stab_{comb}$. By contrast, annotation (i_2,c_2,q_1) of the same age possesses periods of temporal non-existence (v_1,v_2) resulting in a low existence stability of 0.6. Furthermore, (i_3,c_3,q_3) is continuously present in 4 versions of p but received two quality changes ($q_2 \rightarrow q_1 \rightarrow q_3$). Hence, the quality and the combined stability are poor (0.33). The last annotation (i_4,c_4,q_2) shows a perfect combined stability, however it is quite novel ($a_{age}=2$) due to its first occurrence in version v_3.

4 Evaluation

In our evaluation experiments we comparatively analyze the evolution of annotations in the two large annotation sources Ensembl [11] and Swiss-Prot [3] which annotate their proteins with concepts of the Gene Ontology [9]. We first analyze how the annotations evolved for the different provenance types, i.e., different kinds of evidence codes, and how instance (protein) and ontology changes propagated to annotations. In Section 4.2, we additionally analyze the age and stability indicators of Section 3.

4.1 Provenance Analysis

For our study we use available Swiss-Prot and Ensembl versions between March 2004 and December 2008. During this observation period Swiss-Prot (Ensembl) released 14 (28) major versions, namely versions 43-56 (25-52). Both sources provide many functional protein annotations for various species. Whereas Swiss-Prot primarily contains manually curated entries, Ensembl focuses on the automatic generation and integration of data. We consider the functional annotations of human proteins with the concepts of the Gene Ontology (GO) [9] which consists of the three sub ontologies 'biological process', 'molecular function' and 'cellular component'. In the following we do not differentiate between these sub-ontologies and treat GO as one ontology.

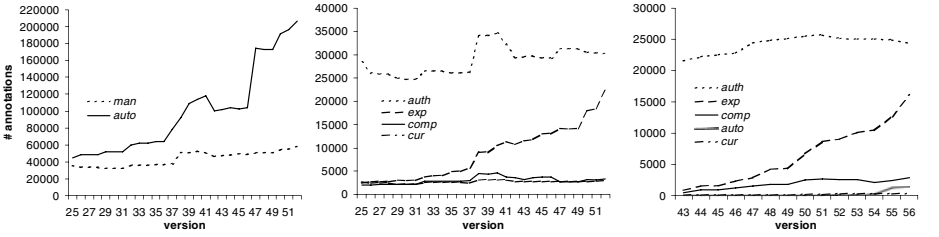


Fig. 7. Evolution of annotations in different EC groups
(a) Manually curated vs. automatically assigned (Ensembl)
(b) "Subclasses" of manually-curated (Ensembl)
(c) All annotations (Swiss-Prot)

Note that Swiss-Prot always attempts to incorporate the current GO release whereas Ensembl often relies on older GO releases in several versions.

Figure 7 shows how the number of GO annotations evolved for different evidence code groups of the EC taxonomy for Ensembl and Swiss-Prot, respectively. Figure 7(a) indicates that Ensembl is dominated by automatically assigned GO annotations (about 78% of the 265,000 annotations in the last version). Furthermore, the growth in the number of automatically determined annotations is very high (factor 4.6 within the last four years). In addition, there is a substantial number of deletions between v_{40} to v_{42}. By contrast, the manually curated annotations grew only modestly by a factor of 1.7. Figure 7(b) shows the development for the manually determined annotations in more detail. We observe a strong increase for experimentally validated annotations ($growth_{exp}$: 8.9) while author statement annotations increased only slightly ($growth_{auth}$: 1.1). The number of curator and computational assigned annotations remained on a very low level.

Figure 7(c) illustrates the evolution of annotations in Swiss-Prot which currently covers about 45,000 annotations, i.e., about six times less than Ensembl. In contrast to Ensembl, Swiss-Prot contains very few automatically generated annotations (1,440) which were recently introduced. The main part of Swiss-Prot annotations encompasses *auth* annotations (about 24,000 in v_{56}). Note that their number is slightly decreasing since v_{51}. The number of *exp* annotations has significantly increased ($growth_{exp}$: 18.5) to about 16,000 at present. Overall, Swiss-Prot provides predominantly manually curated annotations that exhibit a continuous, stable evolution without remarkable fluctuations.

The table in Figure 8 summarizes the number of evolution operations that have been carried out since March 2004 in Swiss-Prot and Ensembl. To determine the changes we compared objects of different versions based on their accession numbers to generate sets of added or deleted objects. More complex changes such as the substitution or merge of proteins that may cause annotation changes (Chg_{ins}) or deletions (Del_{ins}) were identified with the help of evolution information provided by the source distributors. Particularly, Swiss-Prot offers web services to keep track of the protein history, e.g., accession number changes, while Ensembl logs change events between released versions, e.g., what proteins were replaced by others in a new version. Whereas the

		Add	Chg			Del		
			Chg_{ins}	Chg_{ont}	Chg_{qual}	Del_{ann}	Del_{ins}	Del_{ont}
Sp	abs. (%)	32,613 (53%)	18,214 (30%)			10,502 (17%)		
	-	16,106	56	2,052	8,511	1,369	622	
E	abs. (%)	391,771 (60%)	47,805 (8%)			208,585 (32%)		
	-	4,310	171	43,324	145,209	60,788	2,588	

Fig. 8. Number (and percentage) of evolution operations aggregated over all versions in Swiss-Prot (Sp) and Ensembl (E)

	Swiss-Prot						Ensembl					
	Add		*Chg*		*Del*		*Add*		*Chg*		*Del*	
exp	15,751	48.2%	1,830	10.0%	1,784	17.0%	25,979	6.6%	5,826	12.2%	7,575	3.6%
auth	11,307	34.6%	15,177	83.3%	7,350	70.0%	34,046	8.7%	16,381	34.3%	29,148	14.0%
cur	339	1.0%	65	0.4%	73	0.7%	6,362	1.6%	300	0.6%	6,318	3.0%
comp	3,730	11.4%	1,107	6.1%	1,214	11.6%	6,734	1.7%	5,720	12.0%	4,362	2.1%
auto	1,541	4.7%	35	0.2%	81	0.8%	316,979	80.9%	18,344	38.4%	157,632	75.6%
obs	0	0.0%	0	0.0%	0	0.0%	1,826	0.5%	1,234	2.6%	3,550	1.7%
sum	32,668		18,214		10,502		391,926		47,805		208,585	

Fig. 9. Distribution of the operations add, change, delete in different EC groups in Ensembl and Swiss-Prot

majority of changes are additions (60% in Ensembl, 53% in Swiss-Prot) there is a surprising number of deletions and changes, apparently influenced by some major reorganization such as introduction of new accession numbers. For example, in Swiss-Prot about 30% of all evolution changes are annotation changes (*Chg*) which were primarily caused by instance changes keeping corresponding annotations alive instead of deleting them. By contrast, annotation changes in Ensembl are dominated by quality (here: EC code) changes. In both sources ontology changes only marginally influence changes on annotations. This is also influenced by the fact that annotations are administrated within the instance sources while ontologies are developed independently from the instances. Finally, the number of deletions is non-negligible in both sources especially in Ensembl where 32% of all changes are annotation deletions.

We now analyze the distribution of the evolution operations *add*, *change* and *delete* for the different EC groups, as summarized in Figure 9. In Swiss-Prot about one half of the additions are experimentally validated annotations and a third comprises *auth* annotations. By contrast, change (83%) and delete operations (70%) primarily occur for *auth* annotations indicating a rather high instability for this provenance type. On the other hand, Ensembl predominantly adds and deletes automatically generated annotations (81% and 75% of all adds/deletes, respectively). Annotation changes are distributed mainly over automatically assigned (38%) and author statement annotations (34%). In summary, the evolution of existing annotations occurs primarily for *auto* and *auth* annotations.

We further analyze provenance (EC) changes in more detail to see which new EC codes are chosen for improved annotation quality. The tables in Figure 10 aggregates EC changes in Swiss-Prot and Ensembl for versions since March 2004. Each cell

from / to	exp	auth	cur	comp	auto	Sum	
exp	147	24	0	42	1	214	10%
auth	1,121	270	34	165	0	1,590	72%
cur	7	9	0	3	0	19	1%
comp	160	197	7	0	0	364	16%
auto	16	4	0	1	0	21	1%
Sum	1,451	504	41	211	1	2,208	
	66%	23%	2%	10%	0%		

from / to	exp	auth	cur	comp	auto	obs	Sum	
exp	896	413	11	1,259	2,966	3	5,548	13%
auth	1,592	798	73	1,038	11,901	23	15,425	35%
cur	21	27	0	16	182	0	246	1%
comp	1,280	1,206	26	0	3,101	0	5,613	13%
auto	3,311	10,169	228	2,329	0	116	16,153	37%
obs	79	391	9	12	725	0	1,216	3%
Sum	7,179	13,004	347	4,654	18,875	142	44,201	
	16%	29%	1%	11%	43%	0%		

Fig. 10. Evidence codes changes in Swiss-Prot (left) and Ensembl (right)

outlines how many annotations changed *from* one evidence code (rows) *to* another (columns). Note, that we aggregate changes into the EC groups *exp*, *auth*, *cur*, *comp*, *auto* and *obs*, e.g., changes from ISS to TAS are summarized in "from *comp* to *auth*" while changes from IPI to IDA are mapped into "from *exp* to *exp*". We observe that, annotation changes in Swiss-Prot primarily (72%) occur for author statement (*auth*) annotations and that most new annotations (66%) are experimentally proved (*exp*). This shows the progress of annotation development in the recent years by increasingly using biologically proved annotations which are preferred over mere author statements. In Ensembl, the vast amount of automatically generated annotations leads to a somewhat different picture. Only for the shares of two EC groups, *auto* and *exp*, there is an increase for the new EC codes compared to the original ones. All other EC types reduced their shares due to EC changes, especially *auth* annotations. Most EC changes occurred – in both directions – between *auto* and *auth* annotations indicating a high instability of these provenance categories.

4.2 Age and Stability Analysis

In addition to the evidence code (provenance) information, we now analyze the age and stability measures introduced in Section 3. This analysis occurs for the currently available annotations in the latest versions of Ensembl and Swiss-Prot. We compare these annotations with all versions in the last three years (p), i.e., we use the versions 26-52 of Ensembl and versions 47-56 of Swiss-Prot.

We map the age and stability values into quality taxonomies mentioned in Section 2. We differentiate three *age* groups: annotations that exist since half a year (*novel*), those that were generated between half and one and a half years ago (*middle*) and annotations that are older than one and a half years (*old*). For the stability criteria $stab_{exis}$, $stab_{qual}$ and the combination $stab_{comb}$ we use a minimum threshold of 0.9 for *stable* annotations; lower values indicate *unstable* annotations. Hence, a stable annotation must be present in at least 90% of the versions since its first occurrence and at most 10% quality (EC) changes can occur in the history of an annotation. Note, that we leave out all annotations with evidence code NR (*not recorded*) and ND (*no biological data available*) since these annotations provide no valuable information.

Figure 11 displays the classification results of our method for both annotation sources. The 45,000 (263,000) annotations in Swiss-Prot (Ensembl) are classified using the three mentioned criteria: *provenance* (rows), *age* (columns) further separated by the three stability criteria. White (grey) rows denote the number of

Swiss-Prot

	old			*middle*			*novel*		
	$\|stab_{exis}\|$	$\|stab_{qual}\|$	$\|stab_{comb}\|$	$\|stab_{exis}\|$	$\|stab_{qual}\|$	$\|stab_{comb}\|$	$\|stab_{exis}\|$	$\|stab_{qual}\|$	$\|stab_{comb}\|$
exp	7,980	6,965	6,905	2,306	2,266	2,266	5,655	5,637	5,637
	84	1,099	1,159	0	40	40	0	18	18
auth	22,064	21,913	21,760	1,107	1,101	1,101	1,054	1,054	1,054
	169	320	473	0	6	6	0	0	0
cur	184	160	160	36	36	36	115	115	115
	0	24	24	0	0	0	0	0	0
comp	1,651	1,599	1,589	364	362	362	845	844	844
	16	68	78	0	2	2	0	1	1
auto	96	96	96	35	35	35	1,308	1,308	1,308
	1	1	1	0	0	0	0	0	0
sum	**31,975**	**30,733**	**30,510**	**3,848**	**3,800**	**3,800**	**8,977**	**8,958**	**8,958**
	270	**1,512**	**1,735**	**0**	**48**	**48**	**0**	**19**	**19**

Ensembl

	old			*middle*			*novel*		
	$\|stab_{exis}\|$	$\|stab_{qual}\|$	$\|stab_{comb}\|$	$\|stab_{exis}\|$	$\|stab_{qual}\|$	$\|stab_{comb}\|$	$\|stab_{exis}\|$	$\|stab_{qual}\|$	$\|stab_{comb}\|$
exp	9,473	8,774	8,415	3,378	3,062	3,057	8,808	8,650	8,650
	641	1,340	1,699	9	325	330	0	215	158
auth	22,421	20,488	19,700	4,244	3,949	3,942	2,492	2,425	2,425
	1,024	2,957	3,745	9	124	311	0	35	67
cur	238	190	184	67	60	60	157	149	149
	15	63	69	0	7	7	0	8	8
comp	1,715	1,170	1,079	470	354	353	942	885	885
	198	743	834	7	303	124	0	32	57
auto	71,082	89,115	68,440	62,136	63,245	61,442	49,909	49,608	49,608
	21,392	3,359	24,034	1,818	709	2,512	0	301	301
sum	**104,929**	**119,737**	**97,818**	**70,295**	**70,670**	**68,854**	**62,308**	**61,717**	**61,717**
	23,270	**8,462**	**30,381**	**1,843**	**1,468**	**3,284**	**0**	**591**	**591**

Fig. 11. Classification of annotations in Swiss-Prot and Ensembl by provenance, age and stability; stab>0.9 (white), stab <= 0.9 (grey)

annotations that lie above (beyond) the stability threshold. Swiss-Prot covers proportionately more older annotations (72%) than Ensembl (49%). By contrast, the use of automatic annotations allows Ensembl a relative high share (24%) of young/novel annotations. Despite the high share of older annotations, only 4% of the Swiss-Prot annotations are classified as *unstable* compared to 13% in Ensembl (using $stab_{comb}$). In other words, Swiss-Prot (Ensembl) covers 96% (87%) stable annotations.

Considering the three stability criteria one can recognize for both sources that novel and middle aged annotations are rarely classified as *unstable* due to their short history compared to old annotations. Hence, we examine *old* annotations more precisely w.r.t. their stability. In Swiss-Prot the majority of unstable annotations is due to EC changes ($stab_{qual}$, $stab_{comb}$) while relatively few annotations had an existence instability. Most of the existentially unstable annotations ($stab_{exis}$) are of type *auth* while the absolute majority of unstable Swiss-Prot annotations are of type *exp*. This is in accordance to our observations for EC changes (Figure 10) where many annotations changed to experimental proved annotations. Such instabilities for the current annotations may thus be seen as a provenance improvement. In Ensembl the number of

unstable annotations is primarily caused by existential instability ($stab_{exis}$) caused by temporal non-existence of annotations. The majority of unstable annotations occurs for *auto* (79%) and *auth* (12%) annotations confirming their high instability observed earlier.

Our assessment approach seems especially valuable for annotation sources such as Ensembl containing many unverified annotations that are automatically generated. The approach allows the identification of reliable and less reliable annotations w.r.t. three significant criteria: age, provenance and stability. The used measures $stab_{exis}$ and $stab_{qual}$ constitute orthogonal methods providing different classification results. Users can thus filter a set of annotations, e.g., using only those annotations that existed for a longer time, are experimentally proved or do not show existence or provenance instabilities. For example, one may consider annotations as reliable if they are stable with a middle or old age exhibiting a manual provenance. For these criteria, 34,179 (36,790) annotations of Swiss-Prot (Ensembl) would qualify, i.e., 76% (14%) of all available annotations. Naturally, the selection of quality criteria and the corresponding thresholds (e.g., for age or stability) are highly dependent on the application. So users could also be interested in novel or unstable annotations as these are under strong revision due to a high research interest.

The last aspect underlines that annotation instability is not necessarily a negative feature but may indicate interesting objects or significant new biological findings. Conversely, a high stability may be observed for objects of little interest. The proposed evaluation method allows the selection of either stable or unstable annotations and can thus meet the requirements of different applications and annotation use cases.

5 Related Work

Our work is related to the areas of ontology-based *data quality* and *change management* which have received only little attention so far. The current work on change management mainly focuses on ontologies instead of annotations. There are several approaches that investigate ontology versioning [14,15], define change operations describing differences between two ontologies [17], and formalize the evolution process [20,21]. Complementary, there are only few approaches analyzing the ontology evolution quantitatively [10,24]. In [10] we utilized a generic framework to study the evolution of existing ontologies and to quantify changes of annotation and ontology mappings. Our approach in this paper refines the proposed framework by capturing causes of mapping changes. Hence, we can quantify the changes that have been influenced by ontology and instance changes (additions and deletions) and those resulting from provenance changes, whereas [10] only quantifies added and deleted mapping correspondences (annotations). Furthermore, we introduce and analyze several quality indicators of annotation in this paper.

Data or information quality [19] has been primarily addressed in the context of data integration [16,18]. In life sciences, the quality of annotations especially Gene Ontology annotations including evidence codes has been studied in [6,12,22]. Particularly, the case study in [6] assesses annotation quality by using quality-scores for ECs thereby the scores are intuitively defined by the authors. They show descriptive and comparative statistics w.r.t. the quality-scores and annotations in model eukaryotes.

Furthermore, [12] developed a method to estimate the error rate of curated sequence annotations for a particular evidence code (ISS). The approach utilizes the GOSeqLite database to compare annotations that were generated by sequence similarity vs. those that were not. In [22] the authors recommend the utilization of ECs as an indicator for their reliability. In addition, they show simple distribution statistics of annotations for three self-defined classes (homology-based, literature-based and others) and different species but do not examine the annotation evolution. In contrast to previous work on annotation quality, we propose a generic evolution model allowing a multidimensional analysis of annotations w.r.t. different quality taxonomies (age, stability, provenance). The model makes heavily use of quantified evolutionary changes on instance and ontology level but also includes annotation (quality) modifications.

Like our work, [23] provides stability measures to rate correspondences of available mappings but is focused on ontology mappings interconnecting two ontologies. The idea behind this approach is to consider the correspondence stability in addition to the computed element similarity. Conversely, our approach in this paper focuses on annotation mappings and takes multiple quality taxonomies into account to specify and classify the quality of annotations.

6 Conclusion and Future Work

We propose a generic approach to estimate the quality of ontology-based annotations by taking their evolution history into account. The approach considers instance and ontology changes and their influence on annotation mappings. Our annotation model supports different quality measures, such as provenance, age, and stability and the use of quality taxonomies. For provenance information we utilize existent information on evidence codes. We propose different stability measures for annotations taking temporal non-existence and provenance changes into account. Our approach can be used in different scenarios, e.g., by various analysis applications to filter ingoing annotations and by annotation providers to improve their data quality, especially when they integrate annotations from other data sources.

We applied our model-based approach in a comparative evaluation to study functional protein annotations provided by two large life science annotation sources, namely Swiss-Prot and Ensembl. We observed that most annotation changes are additions of new annotations but there are also many changes and deletions of existing annotations. Most of the annotation changes are caused by instance changes or evidence code changes while ontology changes had a minor impact on existing annotations. We also observed that new experimental findings frequently cause the evidence code of existing annotations to be updated. The high instability was observed for automatically generated annotations (in Ensembl) and annotations based on author statements.

We see several directions for future work. First, our annotation model can be applied for additional annotation data sets, e.g., for different species. Second, the proposed approach can be utilized for enhancing instance-based matching techniques that heavily depend on the reliability of input annotations. Likewise, the quality of automatically generated annotations can probably be improved when they are based on existing high quality annotations, e.g., to avoid verified annotations to be overwritten by automatically determined ones or to mark them as new when they are generated for the first time.

Acknowledgements. This work was supported by BMBF grant 01AK803E "MediGRID – Networked Computing Resources For Biomedical Research" as well as the Interdisciplinary Centre for Bioinformatics founded by the German Research Foundation (DFG).

References

1. Berriz, G.F., King, O.D., Bryant, B., et al.: Characterizing gene sets with FuncAssociate. Bioinformatics 19(18), 2502–2504 (2003)
2. Bose, R., Frew, J.: Lineage retrieval for scientific data processing: A survey. ACM Computing Surveys 37(1), 1–28 (2005)
3. Boutet, E., Lieberherr, D., Tognolli, M.: UniProtKB/Swiss-Prot. Methods in Molecular Biology 406, 89–112 (2007)
4. Boyle, E.I., Weng, S., Gollub, J., et al.: GO:TermFinder - open source software for accessing Gene Ontology information and finding significantly enriched Gene Ontology terms associated with a list of genes. Bioinformatics 20(18), 3710–3715 (2004)
5. Buneman, P., Chapman, A., Cheney, J.: Provenance management in curated databases. In: Proc. of the 2006 ACM SIGMOD International Conference on Management of Data, pp. 539–550 (2006)
6. Buza, T.J., McCarty, F.M., Wang, N.: Gene Ontology annotation quality analysis in model eukaryotes. Nucleic Acids Research 36(2), e12 (2008)
7. Dahlquist, K.D., Salomonis, N., Vranizan, K., et al.: GenMAPP, a new tool for viewing and analyzing microarray data on biological pathways. Nature Genetics 31(1), 19–20 (2002)
8. Gene Ontology - Evidence Codes: `http://www.geneontology.org/GO.evidence`
9. The Gene Ontology Consortium: The Gene Ontology project in 2008. Nucleic Acids Research 36, D440–D441 (2008) (Database issue)
10. Hartung, M., Kirsten, T., Rahm, E.: Analyzing the Evolution of Life Science Ontologies and Mappings. In: Bairoch, A., Cohen-Boulakia, S., Froidevaux, C. (eds.) DILS 2008. LNCS (LNBI), vol. 5109, pp. 11–27. Springer, Heidelberg (2008)
11. Hubbard, T.J., Aken, B.L., Ayling, S., et al.: Ensembl 2009. Nucleic Acids Research 37, D690–D697 (2009) (Database issue)
12. Jones, C.E., Brown, A.L., Baumann, U.: Estimating the annotation error rate of curated GO database sequence annotations. BMC Bioinformatics 8(1), 170 (2007)
13. Kirsten, T., Thor, A., Rahm, E.: Instance-based matching of large life science ontologies. In: Cohen-Boulakia, S., Tannen, V. (eds.) DILS 2007. LNCS (LNBI), vol. 4544, pp. 172–187. Springer, Heidelberg (2007)
14. Klein, M.: Change Management for Distributed Ontologies. PhD thesis, Vrije Universiteit Amsterdam (2004)
15. Klein, M., Fensel, D.: Ontology versioning on the Semantic Web. In: Proceedings of the International Semantic Web Working Symposium (SWWS), pp. 75–91 (2001)
16. Naumann, F., Leser, U., Freytag, J.C.: Quality-driven Integration of Heterogeneous Information Systems. In: Proc. of the International Conference on Very Large Data Bases (VLDB), pp. 447–458 (1999)
17. Noy, N., Klein, M.: Ontology evolution: Not the same as schema evolution. Knowledge and Information Systems 6(4), 428–440 (2004)

18. Rahm, E., Do, H.H.: Data Cleaning: Problems and Current Approaches. IEEE Data Engineering Bulletin 23(4), 3–13 (2000)
19. Redman, T.C.: Data Quality for the Information Age. Artech House (1996)
20. Stojanovic, L., Maedche, A., Motik, B., Stojanovic, N.: User-driven ontology evolution management. In: Gómez-Pérez, A., Benjamins, V.R. (eds.) EKAW 2002. LNCS, vol. 2473, pp. 285–300. Springer, Heidelberg (2002)
21. Stojanovic, L., Motik, B.: Ontology evolution within ontology editors. In: Proceedings of the International Workshop on Evaluation of Ontology-based Tools, pp. 53–62 (2002)
22. Thomas, P.D., Mi, H., Lewis, S.: Ontology annotation: mapping genomic regions to biological function. Current Opinion in Chemical Biology 11(1), 4–11 (2007)
23. Thor, A., Hartung, M., Gross, A., Kirsten, T., Rahm, E.: An evolution-based approach for assessing ontology mappings - A case study in the life sciences. In: Proc. Conference of the Business, Technology and Web (BTW), pp. 277–286 (2009)
24. Yang, Z., Zhang, D., Ye, C.: Ontology Analysis on Complexity and Evolution Based on Conceptual Model. In: Leser, U., Naumann, F., Eckman, B. (eds.) DILS 2006. LNCS (LNBI), vol. 4075, pp. 216–223. Springer, Heidelberg (2006)