

## Detecting inconsistency in biological molecular databases using ontologies

Qingfeng Chen · Yi-Ping Phoebe Chen ·  
Chengqi Zhang

Received: 31 July 2006 / Accepted: 19 March 2007 / Published online: 11 July 2007  
Springer Science+Business Media, LLC 2007

**Abstract** The rapid growth of life science databases demands the fusion of knowledge from heterogeneous databases to answer complex biological questions. The discrepancies in nomenclature, various schemas and incompatible formats of biological databases, however, result in a significant lack of interoperability among databases. Therefore, data preparation is a key prerequisite for biological database mining. Integrating diverse biological molecular databases is an essential action to cope with the heterogeneity of biological databases and guarantee efficient data mining. However, the inconsistency in biological databases is a key issue for data integration. This paper proposes a framework to detect the inconsistency in biological databases using ontologies. A numeric estimate is provided to measure the inconsistency and identify those biological databases that are appropriate for further mining applications. This aids in enhancing the quality of databases and guaranteeing accurate and efficient mining of biological databases.

---

Responsible editors: Shichao Zhang and M. J. Zaki.

---

Q. Chen (✉) · Y.-P. P. Chen  
School of Information Technology, Deakin University, Melbourne, VIC 3125, Australia  
e-mail: qingfeng.chen@deakin.edu.au

Y.-P. P. Chen  
ARC Centre in Bioinformatics, Australia  
e-mail: phoebe@deakin.edu.au

C. Zhang  
Faculty of Information Technology, University of Technology, P.O. Box 123, Broadway, Sydney,  
NSW 2007, Australia  
e-mail: chengqi@it.uts.edu.au

**Keywords** Data preparation · Inconsistency · Ontology · Measure · Biological molecular databases · Integration

## 1 Introduction

Recent development in laboratory technology has resulted in the explosive growth of biological data. Initially, biological data were published and collected using HTML (Hypertext Markup Language) format. However, it cannot describe complex structured documents. This has a negative impact on the presentation of biological information and the integration of biological databases. In addition, the varied organizations, storages and publications of biological data lead to different information types. For example, the representative database NCBI (National Center for Biotechnology Information) (NCBI 2005) adopts mostly the binary ASN.1 format, whereas flat-files are used in EMBL (European Molecular Biology Laboratory) (EMBL 2005), GenBank (Benson et al. 2004) and DDBJ (DNA DataBank of Japan) (Miyazaki et al. 2003).

Biological databases (nucleotide sequences and proteins) have been widely used by biologists for data analysis and querying. Due to the growth in the number of databases and their contents, it is necessary to answer a complex biological question by consulting more than a single database. However, the heterogeneity among independently designed and maintained biological databases has greatly impeded accessibility to these databases (Steven et al. 2002). Therefore, data preparation is a key prerequisite to biological database mining. Integrating diverse biological molecular databases is an essential action to cope with the heterogeneity of biological databases and to also guarantee efficient data mining.

The integration of biological databases is usually confronted with technical and semantical problems. The technical problems can be overcome as most biological molecular databases are implemented on relational database management systems (RDBMS) that provide standard interfaces like JDBC and ODBC for data and metadata exchange (Kohler et al. 2003; Philippi et al. 2004). The remaining problems involve semantic issues as described in Williams (1997) and Karp (1995). The inconsistency that arises from semantic issues, such as attribute conflicts, thus challenges current methods to integrate biological databases.

When we want to integrate heterogenous biological databases, they are situations in which we expect some degree of inconsistency in the obtained data. Suppose  $D_i$  represents a biological database. The databases may contain some data that is semantically inconsistent, such as an English species name *mouse* and a systematic species name *Mus Musculus* in  $D_1$  and  $D_2$ , respectively. The mining on highly inconsistent biological databases is inefficient and may result in uninteresting and even inaccurate results. Inconsistency must be dealt with before mining biological databases. It is critical to achieve efficient and accurate mining of biological databases using integration to enhance the quality of biological databases for users or data providers. This urges us to find an intuitive way to measure the inconsistency and to identify the databases that are acceptable for further mining applications.

There have been considerable efforts to address the inconsistency issues in knowledge bases. Hunter presented a method to measure inconsistency in knowledge bases and analyzed inconsistent knowledge by considering the conflicts arising in the minimal quasi-classical (QC) models for that knowledge (Hunter 2002). Hunter also proposed a framework for characterizing inconsistency that can be used in processes for conflict resolution (Hunter 2003). Jinxin presented a knowledge merging operator based on Dalal distance that is capable of measuring the inconsistency between a given world and the knowledge bases in an intuitive way (Jinxin 1996). Although Hunter and Jinxin have shown an ability to deal with the inconsistency in knowledge bases, their models are inappropriate when addressing the semantic heterogeneity of biological databases due to the databases' inherent complexity and diverse terminologies.

To measure inconsistency in biological databases, we often need to access semantically conflicting data. Although each biological database uses its own terminology to share languages for communication, the biological knowledge from a database is not 'machine understandable' and prevents us from accurately measuring the inconsistency in databases. Ontologies such as GO (Ashburner et al. 2000; Go 2006), consisting of an agreed upon vocabulary of concepts (terms) and specification of the relationships among these concepts, are an ideal option to handle the semantic heterogeneity of databases using precise description of the data's semantics and promote the reliable and reusable biological knowledge.

Nevertheless, biological database integration by ontologies is not as good as we might expect due to (1) ontologies with independent terminologies and structures are often incompatible, which causes difficulties in knowledge acquisition from biological databases; (2) heterogeneity, such as synonyms results in a significant lack of interoperability between biological databases; (3) biological databases with high inconsistency are subject to inefficient integration, and may lead to uninteresting knowledge. Although considerable efforts have been devoted to create OBO (Open Biomedical Ontologies), its ability to handle inconsistency is still far from perfect due to diverse heterogenous biological databases. It is desirable that the practical mining is only operated on consistent biological databases.

In this study, we present a framework to measure the inconsistency in biological molecular databases by using ontologies. This not only helps us to determine those biological databases that are appropriate for further mining applications, but also ensures efficient mining of biological databases. The presented experiments demonstrate that our framework is useful and promising in detecting inconsistency in biological molecular databases and enhancing the quality of databases for data mining.

The remainder of this paper is organized as follows. Section 2 presents related work. In Sect. 3, the basic concepts and preliminaries are defined. The framework to measure inconsistency in biological databases is shown in Sect. 4. In Sect. 5, experiments are presented. Section 6 briefly discusses our methodology and future directions. Section 7 concludes this paper.

## 2 Related work

In the past decade, over several hundred biological molecular databases such as GenBank and NCBI have been publicly available on the Internet. Not only do they provide a convenient and efficient way to access biological data but also pose a big challenge to extract interesting knowledge from these databases (Hunter 1993; Chen 2005). The heterogeneity of biological databases greatly impedes knowledge sharing and further database integration (Chen and Colomb 2003). How to derive high quality and reliable data from diverse biological databases has been a key issue in data mining. To achieve this goal, we should put extra emphasis on the stage of data preparation (Zhang et al. 2003, 2004).

Database integration, which mainly includes data warehousing methods and federated database methods, has been an important action to cope with heterogeneities of multiple databases. Data warehousing methods, like SRS (Etzold et al. 1996) and DBGET/LinkDB (Fujibuchi et al. 1998), directly provide integrated access by using indexed flat-files. Federated database methods, such as DiscoveryLink (Hass et al. 2001), integrate multiple autonomous database systems into a single federated database by using a meta-database management system (DBMS). Although the above approaches are effective in the integration of heterogeneous databases, the semantic issues as described in Karp (1995) have been greatly ignored.

A number of efforts have been devoted to achieve automated, intelligent and reliable integration of biological databases. One option is to combine all possible application programs into web service, and use them to make connections to other services. Oinn proposed a tool for the composition and enactment of bioinformatics workflows (Oinn et al. 2003). Another way is to use ontology technology which specifies a set of concepts (conceptualization) with precise semantics. SEMEDA (Semantic meta database) (Kohler et al. 2003) supports querying databases via a powerful interface and which enables users to query databases without requiring any details about the data sources. The interoperation of information requires a consistent and shared understanding of the meaning of that information, however the metadata, such as flat-files in biological databases, is often implicit. The terminologies provide common vocabularies of a domain, but cannot ensure that everyone has a consistent understanding amongst each other. Therefore, a comprehensive reusable reference ontology of biological concepts is a prerequisite for the integration of biological databases.

There have been several ontologies that were used as repositories of potentially reusable biological knowledge. RiboWeb (Chen et al. 1997) aims at facilitating the construction of three-dimensional models of ribosomal components such as bonded molecules, biological macromolecules and regions of molecules. EcoCyc ontology (Karp et al. 2000) covers *E. coli* gene regulation, metabolism and signal transduction. RiboWeb and EcoCyc both use frames as the type of knowledge representation. Gene ontology (GO) (Ashburner et al. 2000; Go 2006) project intends to produce a controlled vocabulary for all organisms, even as knowledge of gene and protein roles in cells is accumulating and changing.

TAMBIS ontology (Baker et al. 1999) enables biologists to ask questions about multiple external databases using a query interface. It uses DLS (description logics) as a knowledge representation language instead of frames. They include an individual vocabulary of terms and specification.

Recently, a number of ontology-based applications have also been developed (Steven et al. 2002; Kohler et al. 2003). Philippi (2004) proposed a method for the ontology-based semantic integration of life science databases using XML technology. Karp (2000) presented an ontology for biological function according to molecular interactions. Yeh (2003) proposed methods for knowledge acquisition, consistency checking and concurrency control for Gene Ontology. In addition, XML has also become commonly used as a means for data exchange in different areas due to the fact that XML can facilitate the carrying out of sophisticated retrievals and provide information about their structure. DNA data bank of Japan (DDBJ) (Miyazaki et al. 2003) itemizes the pieces of information in an entry (genome sequence) by XML, and proposes the DDBJ–XML format for presenting the contents of an entry. Fujibuchi (1998) proposed a general architecture for ontology driven data integration based on XML technology and described a prototypical implementation of this architecture based on a native XML database and an expert system shell.

The previous studies have been useful in prompting the interoperability of life science databases. However, the method to measure the inconsistency in biological databases is underdeveloped. Without a qualification of inconsistency, it is difficult to identify databases that are appropriate for further mining applications.

### 3 Semantics description

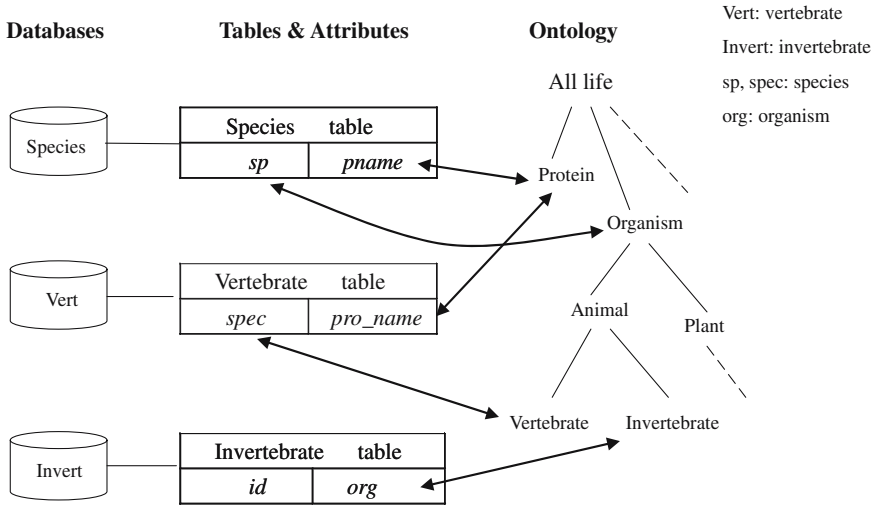
#### 3.1 Symbols and formal semantics

Suppose  $\mathcal{L}$  denotes a set of proposition formulae formed in the usual way from a set of atom symbols  $\mathcal{A}$ . We use variables  $c \in \mathcal{A}$  for concepts such as *Gene* and *Protein*;  $\phi$ ,  $X$  and  $Y$  for formulas; and  $\alpha$ ,  $\gamma$ ,  $\beta \in \mathcal{A}$  for database attributes. Let  $\equiv$  be logical equivalence, let  $\varpi$  be the weight of biological databases in a taxonomy and let  $CV_E$  and  $CV_S$  be *controlled vocabulary English species name* and *controlled vocabulary systematic species name*, respectively. A model of a formula  $\phi$  is a possible set of atoms where  $\phi$  is true in the usual sense.

Database *metadata* usually presents database schema information, data about the **DBMS** and relevant data such as mark-up in flat-files that are required to access a data source. The schema of databases consists of datatypes (*domains*) and tables (*relations*). Tables consist of attributes (*fields*) and corresponding datatypes, and may contain data within the limits of these domains.

*Controlled vocabularies* are a set of named concepts that may have an identifier. The concepts or their identifiers are often used as database entries.

**Definition 3.1** Suppose  $t$ ,  $def$ ,  $id$  and  $sn$  denote term, definition, identifier and synonyms, respectively. Let  $C$  be the set of concepts of databases.



**Fig. 1** Biological database attributes are linked to ontology concepts. Attributes *pname* and *pro\_name* from databases *Species* and *Vert* have different attribute names, but they can be connected by a common concept *protein* of the ontology

$$\text{Controlled Vocabulary } CV := \{c | c = (t, def, id, sn) \in C\}$$

For example, in *Gene Ontology*, each concept (*biological process*) includes a term (*recommended name*), an identifier (*id: GO: number*), definition (*explanation and references*) and synonyms (*other names*). Apart from concepts, an ontology also includes relationships, including ‘*is-a*’ (*Specification relationships*) and ‘*part-of*’ (*Partitive relationships*). Thus, concepts can correlate with each other, such as ‘*Enzyme is a Protein*’ and ‘*Membrane is part of Cells*’. Although ‘*part-of*’ relationship can be defined, only the transitive ‘*is-a*’ hierarchy is required for querying databases. Ontology can be viewed as a tree, where the nodes and directed edges denote concepts and relationships, respectively.

**Definition 3.2** Let *O* be an ontology, and *r* be relationships that link concepts. An ontology can be defined as a set of tuples.

$$\text{Ontology } O := \{(c_1, c_2, r) | c_1, c_2 \in CV, \text{ and } r: c_1 \rightarrow c_2\}$$

where  $c_1 \rightarrow c_2$  denotes the relation *r* from  $c_1$  to  $c_2$ , such as ‘ $c_1$  is-a  $c_2$ ’.

**Example 3.1** In Fig. 1, Vertebrate, Animal, Plant and Organism are connected by ‘*is-a*’ relationships. (*Animal, Organism, Animal* → *Organism*), (*Plant, Organism, Plant* → *Organism*), (*Vertebrate, Animal, Vertebrate* → *Animal*) and (*Invertebrate, Animal, Invertebrate* → *Animal*) represent the ontology.

To measure inconsistency in biological databases, the databases have to be semantically defined by ontologies. One of the key processes is to link the attributes of databases to a specified ontology. Then, users can execute queries to

obtain data from them. To search in attributes of heterogenous databases, a number of complex query operations need to be implemented. If the attributes cannot be found in a current database, it is often the case that the queried attributes have to be mapped to corresponding concepts of ontology, to enable the search for semantically equivalent attributes in the other databases. Occasionally, additional operations are needed, such as translation and cross-reference.

The above facts can be stated in the form of expressions called *sentences*. We define an *atomic sentence* by the following  $n$ -ary relation operator  $\pi$ . Let  $a_i, 1 \leq i \leq n$ , be atom symbol.

$$\pi(a_1, a_2, \dots, a_n)$$

where the atom symbols can be ontologies, controlled vocabularies such as  $CV_E$ , database attributes such as  $pname$  and  $pro\_name$ , concepts such as *animal* and *organism*.

There are entailment relationships between the above operations. Let  $Att_1$  and  $Att_2$  be attributes of database  $DB_1$  and  $DB_2$ , respectively.

**Mapping:**

$$maps(Att_1, c) \wedge maps(Att_2, c) \rightarrow Att_1 \equiv Att_2 \tag{1}$$

states that if  $Att_1$  and  $Att_2$  can be mapped to a common concept  $c$  of ontology  $O$ ,  $Att_1$  and  $Att_2$  are viewed as having the same semantic definition regarding  $c$ , such as  $pname$  and  $pro\_name$  in Fig. 1.

**Translation:**

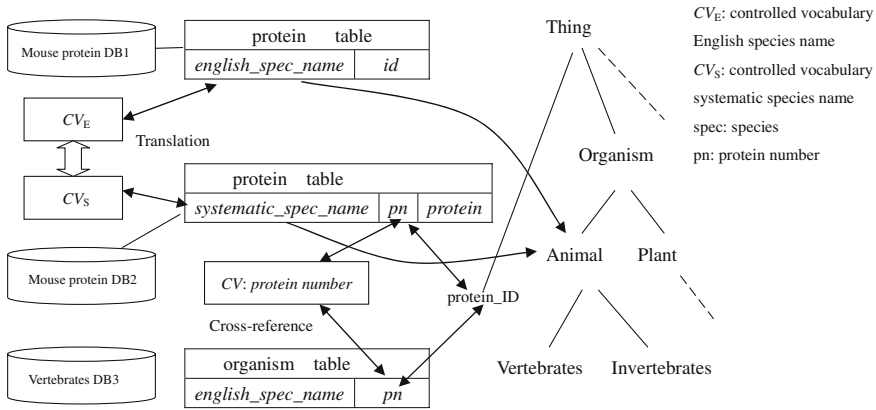
$$translates((CV_1, CV_2), < Att_1, Att_2 >, c) \rightarrow Att_1 \equiv Att_2 \tag{2}$$

states that if  $Att_1$  and  $Att_2$  can be translated into a common concept  $c$  by synonymous concepts,  $Att_1$  and  $Att_2$  are regarded as having the same semantic definition in relation to  $c$ , such as, in Fig. 2, *mouse* in the attribute ‘*systematic\_spec\_name*’ and *Mus Musculus* in the attribute ‘*english\_spec\_name*’ by  $CV_E$  and  $CV_S$ .

**Cross-reference:**

$$cross - reference(CV, Att_1, Att_2, c) \rightarrow Att_1 \equiv Att_2 \tag{3}$$

states that if  $Att_1$  and  $Att_2$  can be linked to a common concept  $c$  by *cross-reference*, they are semantically equivalent by  $c$ , such as, in Figure 2,  $pn$  in  $DB_2$  and  $DB_3$  by the concept  $protein\_ID$ .



**Fig. 2** Translation by mapping synonymous concepts of controlled vocabularies is used to link databases with synonyms. Database attributes corresponding to the same concept and sharing the same controlled vocabulary can be viewed as cross-references of attributes

**Taxonomy:**

$$\forall c_1, c_2, c_3 \in O, is - a(c_1, c_2) \wedge is - a(c_2, c_3) \rightarrow is - a(c_1, c_3) \tag{4}$$

states that if  $c_1$  'is-a'  $c_2$ , and  $c_2$  'is-a'  $c_3$ , we can deduce  $c_1$  'is-a'  $c_3$ . It actually indicates the 'is-a' relationship holds transitivity.

The above axioms describe the possible processes in response to user's queries about database attributes. Ontology plays a central role in mapping the attributes to common concepts or translating attributes between different controlled vocabularies. The query can be classified into two categories according to entered attributes.

- if a queried attribute is found in databases, it will be mapped to a concept of ontology, by which to connect to other database attributes;
- if no existing database attribute is matched, a corresponding concept of ontology is selected. Its *sub-concepts* and *super-concepts* will be searched for the attribute. The details can be seen below.

In the first category, users usually search in the attribute *Att* in conjunction with a specified term *T*. *T* is actually a complementary description that locates the relevant databases. For example, in Fig. 1, if a user wants to search in attribute *pname* for *mouse*, the term *mouse* will be combined with *pname* to locate databases *Species* and *Vert*. As to the latter, the queries first attempt to find the concept of ontology that is related to the queried attributes. The *sub-concepts* and *super-concepts* will be searched for the specified *term*. They are defined as:

$$sub(C) = \{c | \forall c, is - a(c, C)\} \tag{5}$$

$$sup(C) = \{c | \forall c, is - a(C, c)\} \tag{6}$$

where the ontology  $O$  can be regarded as a tree,  $sub(C)$  consists of all child nodes of the parent node  $C$  and  $sup(C)$  includes all parent nodes of the child node  $C$ .

However, some unwanted concepts can be returned, which will result in unnecessary queries and results. For example, for the query  $animal : mouse$ , we can get  $sub(Animal) = \{Vertebrate, Invertebrate\}$ . It is obvious that *mice* is exactly in the *Vertebrate*, rather than *Invertebrate*, concept by the taxonomic tree in Hunter (1993). Thus, the database attributes should be semantically defined as specifically as possible to facilitate the interoperability between biological databases. For example,  $(vertebrate, mouse)$  instead of  $(Animal, mouse)$  can avoid the access to ‘*invertebrate*’. Also, we can reduce the redundancy by including a term  $T$  along with the query. The concept constraints are thus defined to avoid redundancy concepts.

$$sub(C, T) = \{c | \forall c, is - a(c, C), c \sqsupseteq T\} \tag{7}$$

where  $\sqsupseteq$  denotes an inclusion relationship on account of semantics. In the similar manner,  $sup(C, T)$  is defined as

$$sup(C, T) = \{c | \forall c, is - a(C, c), c \sqsupseteq T\} \tag{8}$$

**Example 3.2** Suppose the queried database attribute is ‘*Animal: parrot*’. In Fig. 1, we have  $sub(Animal, parrot) = \{Vertebrate\}$ ,  $sup(Animal, parrot) = \{Organism\}$ .

It is observed that  $Vertebrate \in sub(Animal, parrot)$  is irrelevant to the term *parrot*. Therefore, ontologies should also be defined as specifically as possible. For example, if a *Mammals* concept is included to be a subconcept of *Vertebrate*, the search for the unwanted concepts such as *Fish* and *Reptiles* can be avoided. The obtained concept  $c \in sub(C, T) \cup sup(C, T)$  can then be mapped to the desired attributes of other databases.

**Definition 3.3** Let  $ATT_{DB} = \{a_1, a_2, \dots, a_n\}$  be the set of attributes of biological database  $DB$ .  $ATT_R$  and  $ATT_C$  denote the set of attributes from reference database and compared databases, respectively.

The reference database is the database that includes the queried attribute or the attribute that can be mapped to the concepts in  $sub(C, T) \cup sup(C, T)$ . It aids in deciding whether the attributes in compared databases are consistent with the queried attribute.

**Example 3.3** In Fig. 1,  $ATT_{Species} = \{sp, pname\}$ ,  $ATT_{Vert} = \{pro\_name, spec\}$  and  $ATT_{Invert} = \{id, org\}$ . Given a query  $pname$ , then  $ATT_R = \{sp, pname\}$  and  $ATT_C = \{pro\_name, spec, id, org\}$ .

**Definition 3.4** Let  $\models$  be a support relationship. For a set of database attributes  $ATT_{DB}$ ,  $ATT_{DB} \models$  is defined as follows.

(1) if the queried database attribute  $\alpha$  is found in current databases, we have

$$\begin{cases} ATT_R \models \alpha & \text{iff “} ATT_R \text{ contains } \alpha\text{”} \\ ATT_C \models \neg\alpha & \text{iff “} ATT_C \text{ contains } \beta\text{”} \end{cases}$$

Here  $\beta$  is a database attribute of compared databases, which is semantically equivalent to  $\alpha$  by a medium concept of ontology.

(2) if the queried database attribute  $\alpha$  is not in the databases but can be mapped to a concept of ontology, we have

$$\begin{cases} ATT_R \models \alpha_1 & \text{iff “} ATT_R \text{ contains } \alpha_1\text{”} \\ ATT_C \models \neg\alpha_1 & \text{iff “} ATT_C \text{ contains } \alpha_2\text{”} \end{cases}$$

where  $\alpha_1$  represents the mapped attribute in reference database by  $c \in sub(C, T) \cup sup(C, T)$ , and  $\alpha_2$  is the corresponding database attribute to  $c$  in compared databases.

In reality,  $(\alpha, \beta)$  and  $(\alpha_1, \alpha_2)$  are called synonyms because they use different names to mean the same thing.

#### 4 Detecting inconsistency of biological databases

##### 4.1 Minimal models of queried biological database attributes

This section defines an operator by the models of database attributes to measure the inconsistency in biological databases.

**Definition 4.1** Suppose  $ATT \in \wp(\mathcal{L}), X \in \wp(\mathcal{A})$ , where  $\wp$  is the power set function. Let  $ATT_{DB}$  be attributes from  $DB \in \{R, C\}$ . Let  $X \models ATT$  denote that  $X \models \alpha$  holds for every  $\alpha$  in  $ATT$ .

$$model(ATT) = \{X \in \wp(\mathcal{A}) \mid X \models ATT\}$$

where  $ATT$  denotes a set of database attributes. The model of  $ATT$  actually represents a set of atoms that support  $ATT$ .

For each atom  $\alpha \in ATT_R$  or  $ATT_C, X \models \alpha$  means that  $X$  contains  $\alpha$ . On the contrary,  $X \models \neg\alpha$  means that a semantically equivalent attribute of  $\alpha$  supported by  $X$  exists, which can be linked to  $\alpha$  via ontology. In particular, if no equivalent attribute is found in  $ATT_{DB}$ , we also have  $X \models \neg\alpha$ . The set of database attributes  $ATT$  is the union of  $ATT_R$  and  $ATT_C$ . Thus, we have

$$ATT = ATT_R \sqcup ATT_C$$

where  $\sqcup$  denotes a multiset union operator but the repeated items are reserved. For example, let  $ATT_R = \{\alpha\}$  and  $ATT_C = \{\alpha, \beta\}$ . Hence  $ATT_R \sqcup ATT_C = \{\alpha, \alpha, \beta\}$ . If  $ATT_R \sqcup ATT_C = \emptyset$ , it represents that the biological databases do

not support the queried attribute. In other words, the accessible databases are irrelevant to the queried database attribute.

**Example 4.1** (Continue Example 3.3)  $ATT = ATT_R \sqcup ATT_C = \{sp, pname, pro\_name, spec, id, org\}$ . Thus,  $\{sp, pname\} \models ATT_R, \{pro\_name, spec, id, org\} \models ATT_C$  and  $\{sp, pname, pro\_name, spec, id, org\} \models ATT$  denote a model of  $ATT_R, ATT_C$  and  $ATT$ , respectively.

The models of queried attributes might include not only database attributes but also the pathways of ontology, such as  $maps(pname, Protein) \wedge cross-reference(pname, pro\_name, Protein) \rightarrow pro\_name$  with  $ATT_R$ , and  $maps(pro\_name, Protein) \wedge cross-reference(pname, pro\_name, Protein) \rightarrow pname$  with  $ATT_C$ . However, the pathways and medium concepts, such as *Protein*, have no influence in measuring inconsistency. Thus, they should be ignored in the model.

To measure inconsistency in biological databases, we use minimal models  $Mmod$ .

**Definition 4.2** Let  $ATT \in \wp(\mathcal{L})$ . The set of minimal model for  $ATT$  is defined as

$$Mmod(ATT) = \{X \in model(ATT) \mid \text{if } Y \subset X, \text{ then } Y \notin model(ATT)\}$$

**Example 4.2** (Continue Example 3.2) Let  $ATT_1 = \{spec\}$  and  $ATT_2 = \{sp\}$  be the set of attributes with respect to  $sub(Animal, parrot)$  and  $sup(Animal, parrot)$ , respectively. Thus,  $ATT_C = ATT_1 \cup ATT_2$  and  $ATT_R = \{Animal\}$ . Let  $ATT = ATT_C \cup ATT_R$ . Then, we have  $Mmod(ATT) = \{\{Animal, spec, sp\}\}$ .

#### 4.2 Detecting inconsistency of minimal models

We now consider a measure of inconsistency called compatibility of biological databases. The *consistentset* of a minimal model  $Y$  includes the database attributes that have identical names with the reference attributes. The *conflictset* of  $Y$  consists of (1) the database attributes that are semantically equivalent to the queried attribute; and (2) the *null* attribute that denotes no attribute is semantically equivalent to the reference attribute.

**Definition 4.3** Let  $Y \in \wp(\mathcal{A})$  be a minimal model of the queried attribute. Let  $\alpha$  be a selected reference attribute from reference database  $R$ . The *consistentset* and *conflictset* are defined as

$$Consistentset(\alpha) = \{\beta \mid \beta \in Y, \beta \equiv \alpha\} \tag{9}$$

$$Conflictset(\alpha) = \{\beta \mid \beta \in Y, \beta \equiv \neg\alpha \text{ or } \beta \equiv null\} \tag{10}$$

If  $Consistentset(\alpha) = \emptyset$ ,  $Y$  is totally inconsistent with  $\alpha$ , and vice versa; if  $Conflictset(\alpha) = \emptyset$ ,  $Y$  is totally consistent with  $\alpha$ , and vice versa. For simplicity, we

use  $\neg\alpha$  to represent both *null* attributes and *semantically equivalent* attributes of  $\alpha$ .

By *consistentset* and *conflictset*, the compatibility function from  $\mathcal{A}$  into  $[0, 1]$ , is defined below when  $\alpha$  is not empty, and  $Compatibility(\emptyset) = 0$ .

$$Compatibility(\alpha) = \frac{|Consistentset(\alpha)|}{|Consistentset(\alpha)| + |Conflictset(\alpha)|} \times 100\% \quad (11)$$

where  $|Consistentset(\alpha)|$  and  $|Conflictset(\alpha)|$  denote the cardinality of *Consistentset*( $\alpha$ ) and *Conflictset*( $\alpha$ ), respectively. If  $Compatibility(\alpha) = 0$ , the minimal model  $Y$  has no conflict upon  $\alpha$  and vice versa; if  $Compatibility(\alpha) = 1$ , there is no negative attribute  $\neg\alpha$  in  $Y$ .

**Example 4.3** (Continue Example 4.2) Due to  $Mmod(ATT) = \{\{spec, sp, Animal\}\}$ . Let  $\alpha = 'Animal'$ . Thus, we have  $spec = \neg\alpha$  and  $sp = \neg\alpha$ . Thus, we have  $Conflictset(Animal) = \{spec, sp\}$  and  $Consistentset(Animal) = \{Animal\}$ , and thus  $Compatibility(Animal) = 1/3 = 33\%$ .

The above definition ideally assumes that the databases have an equal degree of importance. In reality, they are assigned different weights. It is reasonable that the databases with high authority, such as NCBI, GenBank and EMBL should have higher weight than other databases. This paper assumes that each biological database is associated with a *weight* which represents the relative degree of importance of the databases. If  $\varpi_{DB_i} > \varpi_{DB_j}, i \neq j$ , it indicates  $DB_i$  is more important than  $DB_j$ , and more of its opinion will be reflected in measuring the inconsistency in databases.

**Definition 4.4** Let  $\varpi_R$  and  $\varpi_{C_1}, \dots, \varpi_{C_k}$  be the weight of the reference database, and compared databases respectively. The weighted compatibility of the set of database  $\{R, C_1, \dots, C_k\}$  regarding database attribute  $\alpha$  is defined as follows, and  $Compatibility(\emptyset, \varpi) = 0$ .

$$Compatibility(\alpha, \varpi) = \frac{\sum_{i \in \{R, C_1, \dots, C_k\}} |Consistentset_i(\alpha)| * \varpi_i}{\sum_{i \in \{R, C_1, \dots, C_k\}} (|Consistentset_i(\alpha)| + |Conflictset_i(\alpha)|) * \varpi_i} \quad (12)$$

The number of occurrence of  $\alpha$  and  $\neg\alpha$  in databases are equal in case of  $Consistentset(\alpha) = Conflictset(\alpha)$ . In this scenario, the databases are definitely inconsistent with respect to  $\alpha$ . Hence, it seems reasonable to define a number that is greater than 0.5 as the threshold of *minimal compatibility*, namely  $mincomp > 0.5$ . Increasing the *mincomp* will lead to higher expectation of the consistency in biological databases. The beliefs in the compatibility are defined as

$$\begin{cases} consistent & \text{if } Compatibility(\alpha, \varpi) \geq mincomp \\ inconsistent & \text{if } Compatibility(\alpha, \varpi) < mincomp \end{cases}$$

The compatibility function provides an intuitive way to measure the inconsistency in biological databases. If the databases are highly inconsistent, they may contain too many incompatible terminologies or most of current databases do not contain the queried database attributes at all. From the results, we are able to identify those databases that are appropriate for further mining applications and enhance the quality of biological databases.

### 4.3 Algorithm design

The presented algorithm identifies database attributes that are semantically equivalent to the queried attribute.

**begin**

**Input:** *D*: biological database; *att*: *T*: queried database attribute with constraint *T*;

(1) **let** *Consistentset*  $\leftarrow \emptyset$ ; *Conflictset*  $\leftarrow \emptyset$ ; *DB*  $\leftarrow \emptyset$ ; *ATT<sub>R</sub>*  $\leftarrow \emptyset$ ;  
*ATT<sub>C</sub>*  $\leftarrow \emptyset$ ;

(2) **forall** *d*  $\in D$  **do**

**if** *d* satisfies *T* **then** *DB*  $\leftarrow DB \cup d$ ;

**end**

(3) **if**  $\exists DB_i \in DB$  contains *att* **then** {

(3.1) *ATT<sub>R</sub>*  $\leftarrow ATT_R \cup ATT_{DB_i}$ ;

(3.2) *ATT<sub>C</sub>*  $\leftarrow ATT_{DB} - ATT_R$ ;

(3.3) *Consistentset*  $\leftarrow Consistentset \cup att$ ;

(3.4) **forall** *att<sub>c</sub>*  $\in ATT_C$  **do**

**if** *att<sub>c</sub>* is identical with *att* **then**

*Consistentset*  $\leftarrow Consistentset \cup att_c$ ;

**else**

**if** *att<sub>c</sub>* is synonymous with *att* via ontology **then**

*Conflictset*  $\leftarrow Conflictset \cup att_c$ ;

**end**

}

(4) **else** {

**if**  $\exists$  concept *c* in ontology that can be mapped to *att* **then**  
the set of relevant concepts *con*  $\leftarrow sub \cup sup$ ;

(4.1) *ATT<sub>con</sub>*  $\leftarrow$  a set of database attributes that can be mapped to concepts of *con*;

(4.2) Find out *Consistentset* and *Conflictset* with respect to specified *att*  $\in ATT_{con}$ ;

}

**end**

In step 1, an empty set is assigned to the variable of *Consistentset*, *Conflictset*, *DB*, *ATT<sub>R</sub>* and *ATT<sub>C</sub>*, respectively. In the following processes, some new elements can be added if the condition is satisfied. We aim to obtain *Consistentset* and *Conflictset* by virtue of *DB*, *ATT<sub>R</sub>* and *ATT<sub>C</sub>*. Thus, we have to first find out *DB*, *ATT<sub>R</sub>* and *ATT<sub>C</sub>*.

Step 2 represents a cycle operation to decide whether a database satisfies the constraint  $T$ . If the database does not satisfy  $T$ , it will be ignored before constructing  $ATT_R$  and  $ATT_C$ . For example, the database irrelevant to the *mammal* will not be considered in the query regarding *Mus musculus*.

In Step 3, if a database  $DB_i$  contains the queried attribute  $att$ , then we can add its attributes  $ATT_{DB_i}$  to the attribute of reference database  $ATT_R$  (Step 3.1). The remaining attributes of database  $DB$  are then stored into the  $ATT_C$  (Step 3.2). Step 3.3 includes the  $att$  in *Consistentset* since  $DB_i$  contains it. According to the obtained  $ATT_C$ , step 3.4 checks whether  $att_c$  is identical with  $att$ . If so, it is saved to *Consistentset*, otherwise, it is saved to *Conflictset*.

However, if a database  $DB_i$  does not contain the queried attribute  $att$ , we will go to Step 4. This attempts to find out the medium concept in the ontology that can be mapped to  $att$  in terms of *sub-concepts* and *super-concepts*. The attribute that can be mapped to the concept of  $con$  will be stored into  $ATT_{con}$  (4.1). The obtained  $ATT_{con}$  is then used to generate *Consistentset* and *Conflictset* (Step 4.2). As a result, we can measure the inconsistency in databases.

## 5 Experiments

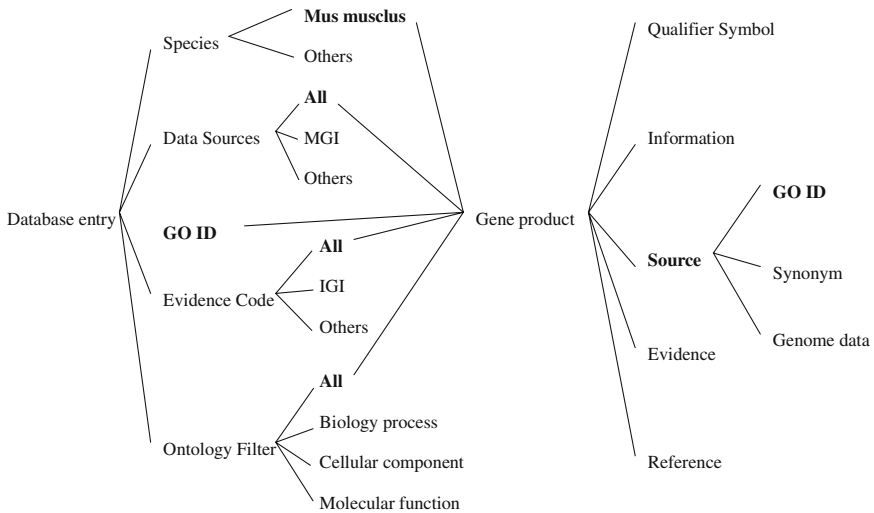
To demonstrate the potential of the principles of semantic inconsistency measures as described above, a prototype system was developed. It focuses on illustrating the measure of inconsistency, whereas the well-established features in other systems, like integration of bioinformatic analysis tools/applications and knowledge acquisition (Kohler et al. 2003) are not considered.

### 5.1 Dataset

We evaluate our approach on real-world gene association data of mouse from GOA (GO Annotation@EBI) (AMIGO 2005) which is responsible for the integration and release of GO (GO 2006) annotations to the multi-species proteomes. Using GOA, a query can be linked to various database sources, e.g. mouse genome information in MGI (Mouse genome informatics), DoTs (Database of transcribed sequences, 43164 human and 78054 mouse DoTS Transcripts (DTs)), UniGene (1313562 UniGene entries), NIA Mouse Gene Index (28219 protein-coding genes (81629 transcripts) 10334 additional gene candidates) and Entrez Gene (2585453 genes). As of September 2005, there are 70861 records in the dataset and of which a gene product can have one or more molecular functions. The gene product may be used in one or more biological processes and may be associated with one or more cellular components. The details of the file format is described in the Annotation Guide (Go 2006).

Three sets of records GO0005201, G00007160 and GO0005578 are selected to form the dataset. GO0005201 is the extracellular matrix structural constituent data of molecular function, containing 20 records retrieved from AmiGO using keyword "0005201". G00007160 is the cell-matrix adhesion data of biological process, containing 42 records retrieved from AmiGo using keyword

CHEN ET AL.



**Fig. 3** Relations between concepts and database entries

“0007160”. GO0005578 is the extracellular matrix (sensu Metazoa) data of cellular component, containing 132 records retrieved from AmiGo using keyword “0005578”. All three GO terms were retrieved only for *Mus Musculus*. After removing duplicated records in “GO0005578”, there are 123 unique gene products. Figure 3 illustrates the relations between concepts and database entries. Each database entry consists of species, data sources, GO ID, evidence code and ontology filter. The entries are highlighted on the left of gene product in bold. Of course, a background translation between English species name (mouse) and systematic species name (*Mus musculus*) is needed at the beginning. On the right of gene product, it shows how to link to other database sources and obtain the genome data.

5.2 Implementation

For simplicity, we assume all databases are assigned equal weight 1 in this study. Three queries are executed below by varying the *GO ID* from 0005201, 0007160 to 0005578, which represents *extracellular matrix structural constituent*, *cell-matrix adhesion* and *extracellular matrix(sensu Metazoa)*, respectively.

1. ‘0005201 : *Mus musculus*’ — search in the term ‘0005201’ for ‘*Mus musculus*’,
2. ‘0007160 : *Mus musculus*’ — search in the term ‘0007160’ for ‘*Musculus*’,  
and
3. ‘0005578 : *Mus musculus*’ — search in the term ‘0005578’ for ‘*Mus musculus*’.

**Table 1** Query results of gene product

GO ID	Qualifier symbol	Source	Evidence	Reference
0005201	<i>Col4a3</i>	MGI	RCA	PMID:12466851
0007160	4733401112Rik	MGI	RCA	PMID:12466851
0005578	<i>Adamts10</i>	MGI	RCA	PMID:12466851

**Table 2** Conversion of synonym of gene product without GO

Source	Synonym	GO ID	Conversion without GO
MGI	<i>Col4a3</i>	0005201	1
DoTs	DT.55281263	0005201	-1
DoTs	DT.91365816	null	-1
TIGR	TC1422845	null	-1
TIGR	TC1556182	null	-1
NIA Mouse Gene Index	<i>Col4a3</i>	0005201	1
Entrez Gene	<i>Col4a3</i>	0005201	1

Three datasets of gene product are obtained. Table 1 shows three records in which **PMID** and **RCA** denote ‘PubMed ID’ and ‘inferred from Reviewed Computational Analysis’, respectively. For brevity, the *information* attribute is not included in the table. The *source* attribute can be used to link to other biological databases. Table 2 shows that the first record in Table 1 can be connected to the database DoTs, TIGR, NIA Mouse Gene Index and Entrez Gene via database MGI.

### 5.3 Results and interpretation

Although most of the databases include the uniform GO ID, they use either the same name of the gene product or a semantically equivalent name. Table 2 presents the synonym of gene product in different databases, in which *null* denotes no GO ID or no predicted GO function was provided in the databases. It is observed that inconsistency exists within the terminology of gene product between database sources, such as *Col4a3* in databases MGI, NIA Mouse Gene Index and Entrez Gene, and *TC1422845* and *TC1556182* in database TIGR. Obviously, it is impractical for a normal user to know that they actually mean the same thing, and that the user could be expected to extract interesting knowledge from such raw data.

In the same manner, each query described above can bring out an association file of gene product, which contains the synonyms in different linked database sources. The details of association files can be reached by <http://www.deakin.edu.au/~qifengch/association.zip>.

To measure the inconsistency, we convert the *synonym of gene product* to integers. The positive integers and negative integers represent coherent name, and synonym or null value, respectively. For example, the last column in Table 2,

**Table 3** Conversion of synonym of gene product with GO

Source	Synonym	GO ID	Conversion with GO
<i>MGI</i>	<i>Col4a3</i>	<i>0005201</i>	1
<i>DoTs</i>	<i>DT.55281263</i>	<i>0005201</i>	1
<i>DoTs</i>	<i>DT.91365816</i>	<i>null</i>	-1
<i>TIGR</i>	<i>TC1422845</i>	<i>null</i>	-1

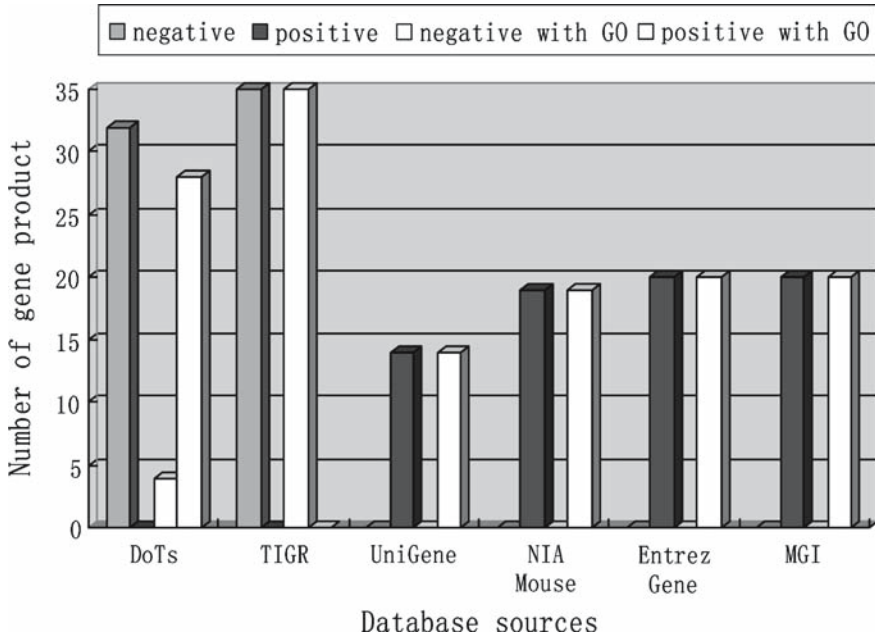
illustrates the converted synonyms of gene product without GO. In the same way, we can perform the conversion for other records. To distinguish different records, the integer is varied by increasing 1 each time until the end of the association file. As a result, the intervals of retrieved records using 0005201, 0007160 and 0005578 will be  $[-20, 20]$ ,  $[-42, 42]$  and  $[-123, 123]$ , respectively. The numbers 20, 42 and 123 represent the result sizes and denote the above three ranges, respectively.

In the association files, the name of gene product of MGI is selected as the reference attribute. A comparison between “without GO” and “with GO” was conducted. In the case of “without GO”, if the name of the gene product in other databases is inconsistent with MGI, the conversion will be assigned a positive integer, otherwise a negative integer. In the case of “with GO”, the conversion will be assigned a negative integer only if the name of gene product in the database is inconsistent with MGI and the database includes no GO ID. Table 3 presents the first four records of Table 2 using the GO conversion. Using the converted data, we are able to generate the *conflictset* and *consistentset*, by which to evaluate the inconsistency in biological databases.

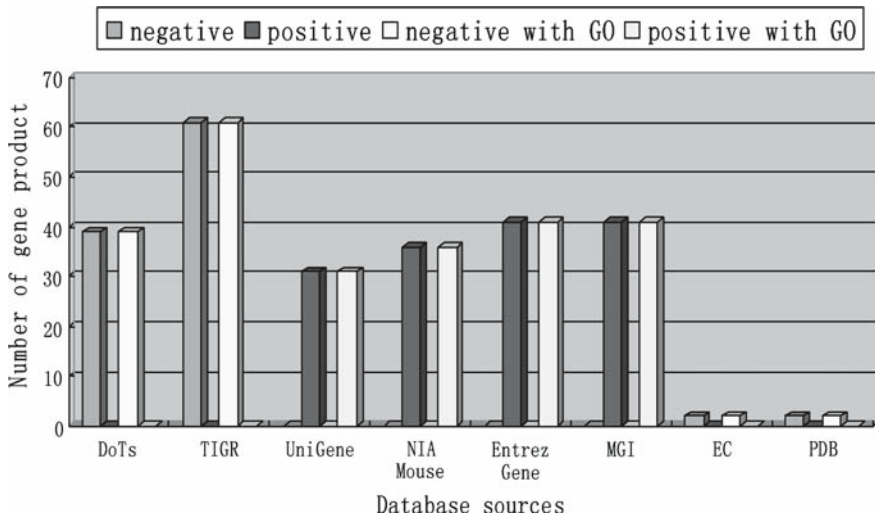
The statistics of synonym gene products in biological databases with consistent name is compared to those with inconsistent name. Figures 4–6 present the comparison between “without GO” and “with GO” with respect to GO:0005201, GO:0007160 and GO:0005578, respectively.

In Fig. 4, the sources UniGene, NIA mouse gene index, Entrez gene and MGI have high consistency with the name of the gene product. However, the sources DoTs and TIGR have high inconsistency with the name of the gene product. In particular, some gene products in DoTs include the same GO ID as the query, therefore the query can directly link to this database using ontology. That is why we can see many positives if GO is applied. Using Eqs. (9–11), we have  $Compatibility(0005201) = 52\%$  without GO and  $Compatibility(0005201) = 72\%$  with GO. This implies that we are able to decrease the inconsistency in biological databases using ontology.

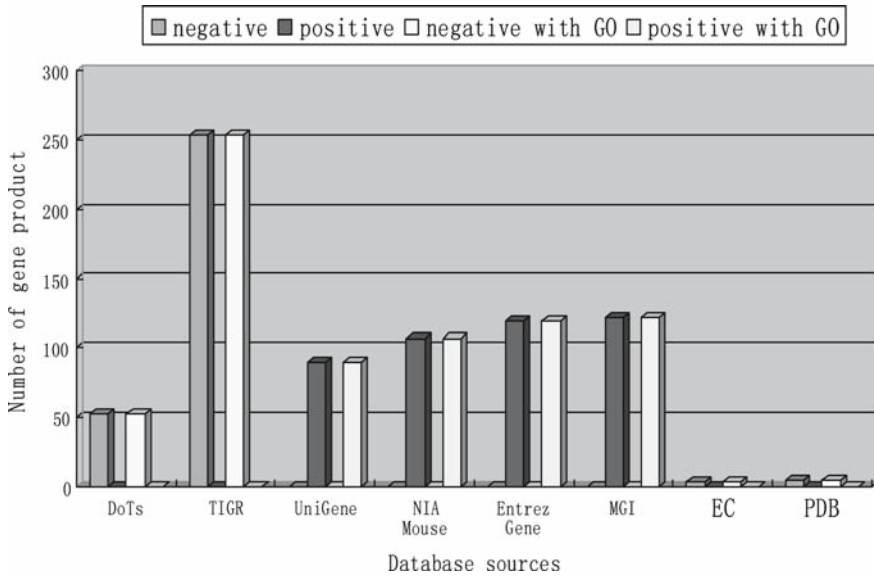
In Figs. 5 and 6, UniGene, NIA mouse gene index, Entrez gene and MGI still show their high coherence with the name of the gene product. In comparison with Figs. 4 and 5, Fig. 6 includes more negative numbers of gene product from TIGR. This indicates that TIGR is the primary source that causes the inconsistency. In addition, two new sources, EC and PDB, are included. It is not surprising that very few gene products are found from these since they are not very relevant to our queries. In the same way, we can compute the compatibility



**Fig. 4** The statistics of synonym of gene products regarding GO: 0005201 in biological databases



**Fig. 5** The statistics of synonym of gene products regarding GO: 0007160 in biological databases



**Fig. 6** The statistics of synonym of gene products regarding GO: 0005578 in biological databases

for 0007160 and 0005578, respectively. Thus, we have  $Compatibility(0007160) = 59\%$  without GO and  $Compatibility(0007160) = 59\%$  with GO;  $Compatibility(0005578) = 58\%$  without GO and  $Compatibility(0005578) = 58\%$  with GO.

It is observed that the compatibility without GO is equal to the compatibility with GO in both queries, 0007160 and 0005578. This is because no matching GO ID is available in database sources and the names of gene products are inconsistent with the reference database, such as MGI used in this study.

Table 4 presents the percentage of occurrence of positive attributes and negative attributes in database sources with respect to the queries, in which “+” and “-” denote positive and negative, respectively. In particular,  $0005201_G^+$ ,  $0007160_G^+$  and  $0005578_G^+$  denote the percentage of occurrence of attributes with GO. Except for 0005201, 0007160 and 0005578 have no difference in the occurrence of attributes between “without GO” and “with GO”. On the whole, the database sources UniGene, MGI, NIA Mouse Gene Index and Entrez Gene are consistent with the name of gene product because, without exception, they contain either consistent name of gene product or matching GO ID with the query. In contrast, DoTs, TIGR, EC and PDB show very low consistency with the name of gene product and contain even no GO ID. This may result in low quality data and also prevent us from extracting interesting knowledge.

As a consequence, our approach is not only able to measure the inconsistency in biological databases, but also report to the users the sources that cause the inconsistency. This benefit improves the interoperability between databases, and enhances the data quality for data mining.

**Table 4** Percentage of occurrence of attributes

Query	DoTs	TIGR	UniGene	NIA	Entrez	MGI	EC	PDB
0005201 <sup>+</sup>	0	0	10%	13.6%	14.3%	14.3%	0	0
0005201 <sup>-</sup>	22.9%	25%	0	0	0	0	0	0
0005201 <sub>G</sub> <sup>+</sup>	20%	0	10%	13.6%	14.3%	14.3%	0	0
0005201 <sub>G</sub> <sup>-</sup>	2.9%	25%	0	0	0	0	0	0
0007160 <sup>+</sup>	0	0	12.4%	14.5%	16.5%	16.5%	0	0
0007160 <sup>-</sup>	15.7%	24.5%	0	0	0	0	0.8%	0.8%
0005578 <sup>+</sup>	0	0	12%	14.2%	15.8%	16.2%	0	0
0005578 <sup>-</sup>	7%	33.6%	0	0	0	0	0.4%	0.7%

## 6 Discussion

To achieve efficient and reliable data mining, one of the most important steps is to obtain high quality data that is complete, consistent and clean. In addition, we have to collect data from multiple database sources. Thus, data integration has been a key issue in data mining. There have been many efforts in developing tools for collecting data from biological databases. However, the inconsistent terminology, duplicated records, and even conflicting identifier in different databases can remarkably influence the interoperability between database sources. The obtained raw data are definitely unready for further mining applications and may lead to inefficient mining, missing of interesting patterns and even incorrect knowledge. To confront these significant challenges, this article provides an intuitive way to measure the inconsistency in biological databases using ontology by identifying the sources that are appropriate for further mining applications.

The initial dataset of GO:0005578 contains duplicate records that have the same name of gene product and PMID. This may arise from (1) the same gene product may be submitted by the biologist to more than one database; (2) the gene product is submitted repeatedly to the same database; or (3) fragment and partial entries of the gene product may be saved in different database records. Although the duplicate records are removed from the dataset, it warns us that they may give rise to incorrect evaluation of inconsistency. On the other hand, most of the predicted GO ID in DoTs is *unreviewed* or *none*. In that case, we ignore the records in DoTs.

Although the inconsistency in biological databases can be evidence for further mining applications, the database entries should be input as specifically as possible. For example, the query 0005201:Mus musculus will return more interesting gene products in comparison with 0005201 only. Occasionally, a query is too simple to search but the inconsistency in databases might be low. Our approach depends on the quality of databases. As described above, we can see some duplicate records in the association files and many unreviewed GO ID in the database DoTs. In addition, the accuracy of ontology is also critical to accurately evaluate the inconsistency in biological databases. For example, if

there is no translation between English species name and systematic species name, the query 0005201:Mus musculus might return nothing. Thus, the database attributes should be semantically defined to be as specific as possible. In other words, no database attributes are semantically defined as general top-level concepts such as 'thing'. Nevertheless, these should not impact the capability of our approach to detect inconsistency in biological databases.

Another issue that might influence the efficiency and accuracy of further mining applications is the extra or unpredictable consumption of time and computation due to the use of ontology. It is clear that an unperfect and complex ontology might result in low efficient mining and even inaccurate results. Regardless of the difficulties, it may be an optimal way to unify the terminology by formalizing the collected data from biological databases before commencing mining, or establish a complete mapping relation between them using ontology.

In this study, we measure the inconsistency in biological databases using a numeric estimate and compare their different effects on inconsistency. Based on the comparison, we not only discovered the databases that are appropriate for further mining application but also the databases that need to be further improved.

## 7 Conclusions

Data preparation is a key prerequisite for mining life science databases. However, the heterogeneity caused by varied data formats and data schemas of biological databases results in a significant lack of interoperability among databases and prevents users from extracting interesting patterns from multiple sources. Ontology-based approaches have been successfully used to exploit biological knowledge by reducing the semantic heterogeneity and thus promoting the flexible and reliable interoperability between biological databases. This paper proposes an ontology-based framework to detect the inconsistency in biological databases, to allow users to identify the sources that are appropriate for further mining application, and data providers can enhance the quality of databases.

Unlike the general data integration, our approach provides an intuitive compatibility function to measure the inconsistency. It enables us to not only discover the sources with duplicate or inconsistent records but also to minimize data loss due to discrepant terminology. The mining can be implemented either on the sources with high consistency or on the sources after enhancement. Thus, it has good potential for guaranteeing accurate and efficient mining of biological databases. The conducted experiments demonstrate that our approach is useful and promising.

**Acknowledgements** This research is partially supported by Discovery Grants from the Australian Research Council (DP0559251, DP0449535) and a China NSFC major research Program (60496321). Authors would like to thanks action editor, executive editor and anonymous reviewers for their time, input and useful feedback to increase the quality of this paper.

## References

- AmiGO browser, (2005) <http://www.godatabase.org/dev/>
- Ashburner M, Ball CA, Blake JA, Botstein D, Butler H, Cherry JM, Davis AP, Dolinski K, Dwight SS, Eppig JT, Harris MA, Hill DP, Issel-Tarver L, Kasarskis A, Lewis S, Matese JC, Richardson JE, Ringwald M, Rubin GM, Sherlock G (2000) The Gene Ontology Consortium, Gene Ontology: tool for the unification of biology. *Nat Genet* 25(1):25–29
- Baker PG, Goble CA, Bechhofer S, Paton NW, Stevens R, Brass A (1999) An ontology for bioinformatics applications. *Bioinformatics* 15(6):510–520
- Benson DA, Karsch-Mizrachi I, Lipman DJ, Ostell J, Wheeler DL (2004) GenBank update. *Nucleic Acids Res* 32(Database issue):23–26
- Chen Y-PP (ed) (2005) *Bioinformatics technologies*. Springer.
- Chen Y-PP, Colomb BM (2003) Database technologies for L-system simulations in virtual plant applications on bioinformatics. *Knowledge Inform Syst* 5(3):288–314, Springer-Verlag.
- Chen RO, Felciano R, Altman RB (1997) RiboWeb: Linking structural computations to a knowledge base of published experimental data. In: *Proceeding of the 5th international conference on intelligent systems for molecular biology*. AAAI Press, pp 84–87
- DNA data bank of Japan, <http://www.ddbj.nig.ac.jp/>
- EMBL-the European molecular biology laboratory (2005) <http://www.ebi.ac.uk/embl/>
- Etzold T, Ulyanov A, Argos P (1996) SRS: information retrieval system for molecular biology data banks. *Methods Enzymol* 226:114–128
- Fujibuchi W, Goto S, Migimatsu H, Uchiyama I, Ogiwara A, Akiyama Y, Kanehisa M (1998) DBGET/LinkDB: an integrated database retrieval system. In: *Proceeding of the pacific symposium on biocomputing*, pp 683–694, Hawaii
- Gene ontology (2006) <http://www.geneontology.org/>
- Gene ontology annotation database (2006) <http://www.ebi.ac.uk/GO>
- Haas LM, Schwarz PM, Kodali P, Kotlar E, Rice JE, Swope WC (2001) DiscoveryLink: a system for integrated access to life sciences data sources. *IBM Syst J* 40(2); DOI: 10.1147/sj.402.0489
- Hunter L (ed) (1993) *Artificial intelligence and molecular biology*. MIT Press
- Hunter A (2002) Measuring inconsistency in knowledge via quasi-classical models. In: *Proceedings of AAAI-02*, pp 68–73
- Hunter A (2003) Evaluating the Significance of Inconsistencies. In: *Proceedings of the International Joint Conference on AI (IJCAI'03)*, pp 468–473
- Karp PD (1995) A strategy for database interoperation. *J comput Biol* 2(4):59–61
- Karp PD (2000) An ontology for biological function based on molecular interactions. *Bioinformatics* 16(3):269–285
- Karp PD, Riley M, Saier M, Paulsen IT, Paley SM, Pellegrini-Toole A (2000) The EcoCyc and MetaCyc databases. *Nucleic Acids Res* 30(1):59–61
- Kohler J, Philippi S, Lange M (2003) SEMEDA: ontology based semantic integration of biological databases. *Bioinformatics* 19(18):2420–2427
- Lin JX (1996) Integration of weighted knowledge bases. *Artif Int* 83(2):363–378
- Miyazaki S, Sugawara H, Gojobori T, Tateno Y (2003) DNA Data Bank of Japan (DDBJ) in XML. *Nucleic Acids Res* 31(1):13–16
- Oinn TM (2003) Talisman—rapid application development for the grid. *Bioinformatics* 19(Suppl):212–214
- Philippi S, Kohler J (2004) Using XML technology for the ontology-based semantic integration of life science databases. *IEEE Trans Inf Technol Biomed* 8(2):154–160
- Stevens R, Goble C, Horrocks I, Bechhofer S (2002) OILing the way to machine understandable bioinformatics resources. *IEEE Trans Inf Technol Biomed* 6(2):129–134
- The national center for biotechnology information (NCBI) (2005). <http://www.ncbi.nlm.nih.gov/>
- Williams N (1997) Bioinformatics: how to get databases talking the same language. *Science* 275(5298):301–302
- Yeh I, Karp PD, Noy NF, Altman RB (2003) Knowledge acquisition, consistency checking and concurrency control for Gene Ontology (GO). *Bioinformatics* 19(2):241–248
- Zhang SC, Yang Q, Zhang CQ (2003) Data preparation for data mining. *Appl Artif Intel* 17:375–382
- Zhang SC, Zhang CQ, Yang Q (2004) Information enhancement for data mining. *IEEE Intelligent Syst* 9(2):12–13