9

Prioritising Genes with an Artificial Neural Network Comprising Medical Documents to Accelerate Positional Cloning in Biological Research

Norio Kobayashi and Tetsuro Toyoda
Bioinformatics And Systems Engineering (BASE) division, RIKEN, Japan

1. Introduction

Linkage analysis is used to identify genes with a certain phenotype or genetic defect and determine chromosomal intervals containing several tens to hundreds of candidate genes for positional cloning. Before further experiments are performed, the candidate genes must be prioritised using as much biological knowledge as possible. For this purpose, it is an ambitious challenge to create an artificial superbrain that has learned a vast amount of knowledge stored as various omics data.

As conventionally such omics data have been published in different data structures and semantics, a data platform that integrates heterogeneous data into a single machine-readable data set is desired. The Semantic Web is a framework for knowledge description and discovery by inferences; it uses relationships given as semantic links between two entities denoted by Uniform Resource Identifiers (URIs) (Berners-Lee et al., 2001). A goal of the Semantic Web is the realisation of human-machine communication by adding metadata describing the semantic links of entities based on Resource Description Framework (RDF) (Manola & Miller, 2004). In biomedical fields, some datasets using common ontologies shared by people on the Semantic Web, such as the Gene Ontology (Ashburner et al., 2000, The Gene Ontology Consortium, 2006) and the uniprot RDF (http://dev.isb-sib.ch/projects/uniprot-rdf/) have been published in RDF. However, because the task of generating consistent RDF triples (subject, predicate and object) against a vast amount of biomedical content is too expensive, an information space that covers our entire exhaustive biomedical knowledge on the Semantic Web has not been realised.

For the practical use of published biomedical data in the Semantic Web, especially of the data that is difficult to utilise because of the lack of semantic links, it is beneficial to reinforce the acquisition of such data by supplying a hybrid methodology combining not only inferences over the knowledge described with RDF but also those supported by statistical significance over multiple raw documents. For instance, MEDLINE, a biomedical document repository, includes more than 18 million reports; the entire knowledge amount of represented by MEDLINE cannot be consistently reconstructed as well-formed ontology-based knowledge.

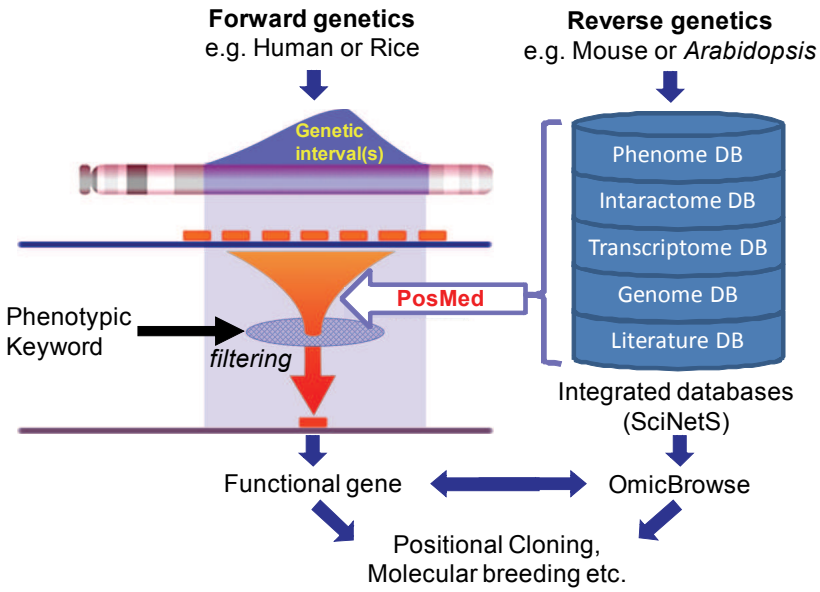


Fig. 1. PosMed accelerates forward genetics gene discoveries (left) by integrating the omics knowledge collected from reverse genetics as an integrated databases named Scientists' Network System (SciNetS; Masuya et al., 2010) (right). PosMed helps users narrow down the candidate responsible genes from those existing within chromosomal intervals. OmicBrowse (Matsushima et al., 2009; Toyoda et al., 2007) helps users look into every piece of detailed information for each candidate gene. The entire system is designed to coherently support positional cloning studies, plant molecular breeding research and plant-upgrading science.

However, the current framework of the Semantic Web cannot handle numerical criteria such as relationship strength or the result of statistical tests of relationships. To effectively use both well-formed RDF datasets and a vast number of biomedical documents, the extension of current query languages should support not only Boolean relationships but also statistically evaluated relationships.

In this chapter, we will discuss the development of a web-based tool named Positional Medline (PosMed) that can immediately suggest genes related to a certain phenotype by accessing a Semantic Web based databases over omics entities named the Scientists' Networking System (SciNetS; Masuya et al., 2010) and document databases (Fig. 1). We initially developed a semantic link database for each entity, which holds relationships between the entity and other entities, including documents such as orthologue relationships and document co-citation relationships. We then developed a search engine named General and Rapid Association Study Engine (GRASE) and an associated query language named General and Rapid Association Study Query Language (GRASQL) (Kobayashi & Toyoda, 2008). GRASQL is a powerful language for expressing the statistical analysis of data retrievable by the RDF query language SPARQL (Prud'Hommeaux & Seaborne, 2008) in a Semantic Web manner. The current implementation of GRASE is optimised to efficiently calculate the statistical prioritisation of candidate genes on the bases of more than 18 million

medical and biological documents and to facilitate quick return of the results within a few seconds of computational time.

Several software tools that have been developed for prioritising positional candidate genes are based on functional annotation, gene expression patterns, protein–protein interaction (PPI) and/or sequence-based features (Adie et al., 2006; Aerts et al., 2006; GeneSniffer; Köhler et al., 2008; Seelow et al., 2008; Van Driel et al., 2005). The evaluation of two of these tools and PosMed using their data sets has demonstrated the effectiveness of PosMed, which showed an accuracy of 88.7%, the highest among the three tools (Thornblad et al., 2007).

Currently, PosMed supports prioritisation of candidate genes for positional cloning in the human, mouse and rat, and prioritisation of other entities not having genomic positions such as metabolites, drugs, diseases and researchers (Yoshida et al., 2009). Further, a plant service version of PosMed named Positional MEDLINE for plant-upgrading science (PosMed-plus)[1] was implemented as the first cross-species integrated database that inferentially prioritises candidate genes for forward genetics approaches in plant science supporting Arabidopsis and rice (Makita et al., 2009).

PosMed and PosMed-plus are available at http://omicspace.riken.jp/.

2. Data model for gene prioritisation in PosMed

2.1 Neural Network representation of statistical algorithm for searching complex semantic web data

PosMed prioritises candidate genes for positional cloning by employing our original database search engine GRASE. As an example of this prioritisation against mouse genes, GRASE is used to execute an inferential process similar to that of an artificial neural network comprising documental neurons (or 'documentrons') that represent each document contained in databases such as MEDLINE (Fig. 2). Given a user-specified query, PosMed initially performs a full-text search of each documentron in the first-layer artificial neurons and then calculates the statistical significance of the connections between the hit documents and the second-layer artificial neurons representing each mouse gene. When a chromosomal interval(s) in mice is specified, PosMed explores the second- and third-layer artificial neurons representing genes within the chromosomal interval by evaluating the combined significance of the connections from the hit documentrons to the genes.

When a chromosomal interval(s) in human is specified, PosMed further explores the fourth-layer artificial neurons representing human genes within the chromosomal interval by using orthologous correspondences between mouse genes and human genes. For the output, PosMed displays the ranked genes with evidence documents in which the user's keyword is highlighted.

PosMed is, therefore, a powerful tool that immediately ranks the candidate genes by connecting them to user's keywords, with connections representing both gene–gene interactions and other biological interactions such as metabolite–gene, drug–gene, disease–gene, phenotype–gene, subcellular localisation–gene, co-expression, PPI, and orthologue and paralogue data. By using orthologous and paralogous connections, PosMed facilitates the ranking of genes based on evidence found in other species.

[1] In this article, PosMed-plus is included in PosMed unless otherwise stated.

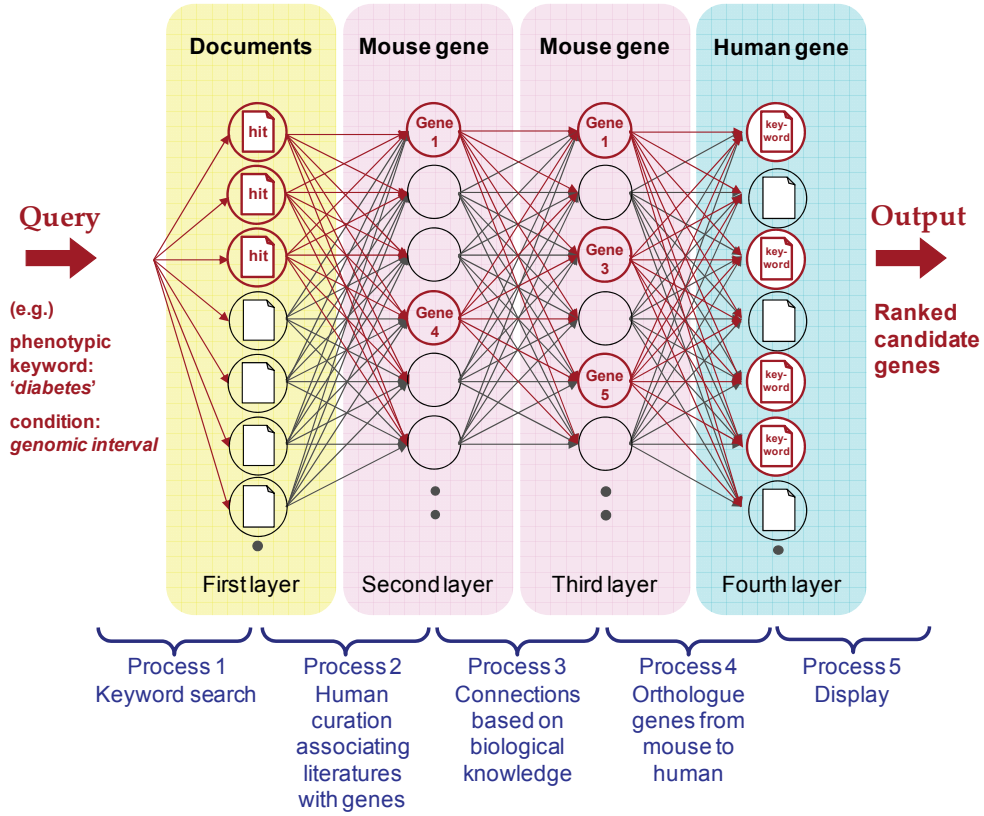


Fig. 2. Neural network model for the PosMed gene search algorithm. As an example, the user's keyword 'diabetes' can be found in several documents, including some in MEDLINE (Process 1). These documents are mapped to genes that are supported by manual curation (Process 2). Using biological knowledge (e.g. protein–protein interaction and co-citation of document sets), PosMed can also suggest genes that do not have the user's keyword 'diabetes' in their associated documents (Process 3). PosMed then returns the candidate genes that are located within the user's specified genomic interval using orthologous relationships (Process 4). Thereafter, the resultant genes are displayed with documents in which the user's keyword is highlighted (Process 5).

2.2 Manual curation work connecting genes to the literature

The accuracy of PosMed is strongly correlated with its ability to make correct associations between genes and documents. This is because GRASE uses these associations to execute direct searches and inference searches that are supported by co-citations. To increase the accuracy of PosMed, we employed manual curation to connect genes and papers by semantic links. Our original curation method is based on named entity recognition (NER; see Section 4.2. for details). Rather than connecting every literature reference to genes, specialised curators create search rules to retrieve all the correct references from document titles, abstracts and MeSH terms.

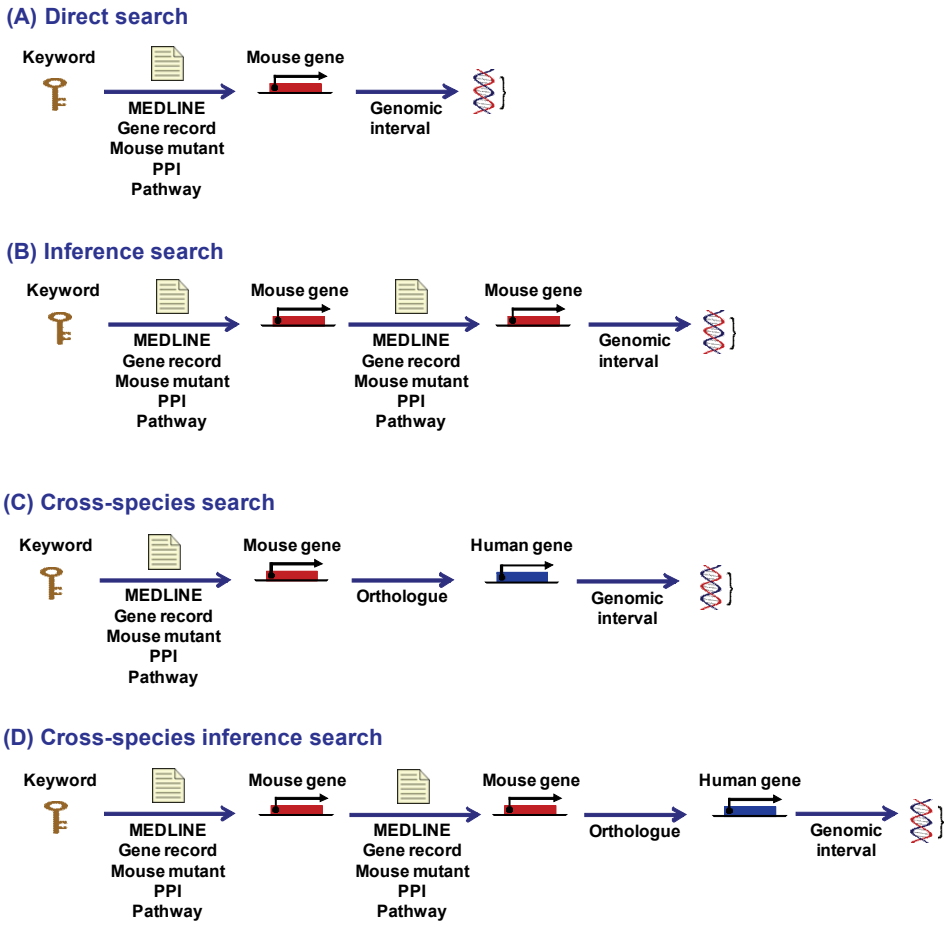


Fig. 3. Sequential data flow representation of PosMed search paths. (A and B) Data flow of PosMed search and comparison of direct search and inference search, respectively. (C and D) Data flow of PosMed cross-species searches.

2.3 Search paths of the PosMed Neural Network

Using the search functionalities of GRASE, PosMed supports the following four types of search:

i. Direct search: GRASE searches genes located in the user's chromosomal interval by performing a full-text search against the set of databases with the user's keyword; i.e. the following search path is realised: *keyword* → *document* → *gene* → *chromosomal interval* (Fig. 3(A)).

ii. Inference search: By applying gene–gene relationships over the genes extracted by a direct search outside the user's chromosomal interval, GRASE discovers further genes that are indirectly related to the keyword via gene–gene relationships; i.e. the following

search path is realised: $keyword \rightarrow document \rightarrow gene_1 \rightarrow gene_2 \rightarrow chromosomal\ \ interval$. The link between $gene_1$ and $gene_2$ is supported by omics data (Fig. 3(B)).

iii. Cross-species search: This is an extension of the direct search (i) to the human genome. The connections from mouse genes to human genes are supported by orthologue data (Fig. 3(C)).

iv. Cross-species inference search: This is an extension of the inference search (ii) to the human genome. As for (iii) above, orthologue data connect mouse genes to human genes (Fig. 3(D)).

In the final stage, these types of search result are integrated into a ranked gene list by species.

3. Statistical query language and its processor

As mentioned above, the core data processing software component of PosMed, which performs statistical inference searching, is GRASE. GRASE is implemented as a prototype of a language processor that interprets and executes a program written in a query language named GRASQL. GRASQL is our extension of SPARQL, which is highly rated because it seems intuitively understandable for typical biologists who are not familiar with programming languages, but does not adequately support statistical evaluation of semantic links.

GRASQL is designed as a language for ranking resultant entities such as genes to discover entities statistically associated with a user's keyword. It does this by considering statistical values computed for each entity on the basis of sets of RDF triples hit by RDF graph pattern matching. In the rest of this section, we present an overview of GRASQL and discuss the programs in PosMed that use it.

3.1 Overview of GRASQL

We start with researcher ranking problems as introductory programs in GRASQL to show how a statistical evaluation is integrated with the existing RDF search.

The first example is researcher ranking using an index called the *h* index to characterise a researcher's scientific output (Hirsch, 2005). The *h* index is introduced as follows: 'A scientist has index *h* if *h* of his or her N_p papers have at least *h* citations each, and the other (N_p - *h*) papers have at most *h* citations each', where N_p is the number of papers published over *n* years. Figure 4 shows a GRASQL query of this problem, which uses MEDLINE abstracts and citation relationships.

First, we obtain a set $\overline{d_r}$ of documents published by each researcher *r* in the MEDLINE abstracts published in 2005 or later, and then documents $\overline{c_d}$ that each cite document $d \in \overline{d_r}$. This search implements the RDF graph pattern matching specified in the WHERE clause in Fig. 4, as shown in Fig. 5. Then, we compute the *h* index for each researcher. This statistical step cannot be realised with RDF graph pattern matching unless an external procedure against the sequences of solutions obtained in the first step is used. In Fig. 4, the EVALUATE clause, which is newly introduced in GRASQL, specifies a method of computing the *h* index `?h` for each researcher `?researcher` using the external statistical function `ris:hIndex` given in Fig. 6. Since MEDLINE does not contain citation information, the `rip:hasCitation` links should be generated on the bases of other resources, such as Google Scholar (Noruzi, 2005).

```
@prefix rio: <http://omicspace.riken.jp/GRASQL/>
@prefix rip: <http://omicspace.riken.jp/GRASQL/predicate>
@prefix ris: <http://omicspace.riken.jp/GRASQL/statistics>
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
@let  %documentSet  rio:MEDLINE
@let  %researcher   rio:Researcher
SELECT ?researcher ?h
WHERE {
    ?researcher    rdf:type            rio:Researcher ;
                   rip:hasDocument     ?doc .
    ?docCite       rip:hasCitation     ?doc ;
                   rdf:type            %documentSet .
    ?doc           rip:publishYear ?year ;
                   rdf:type            %documentSet   .
    FILTER (?year >= 2005)
}
EVALUATE ?h FOR ?researcher {
    ?h = ris:hIndex([?doc,?docCite]);
}
ORDER BY DESC(?h)
```

Fig. 4. GRASQL query that ranks researchers by the *h* index using MEDLINE abstracts and citation relationships. The LET statement is written as `@let %constantName value`, where `% constantName` is the name of a constant that starts with `%`, and *value* is its constant value. The statistical function `ris:hIndex` in the EVALUATE clause is called for each `?researcher` value containing the sequences of solutions obtained by RDF graph pattern matching specified in the WHERE clause. `[?doc,?docCite]` is a sequence of pairs of `?doc` and `?docCite` included in the sub-sequences of solutions concerning the value of `?researcher` when `ris:hIndex` is called.

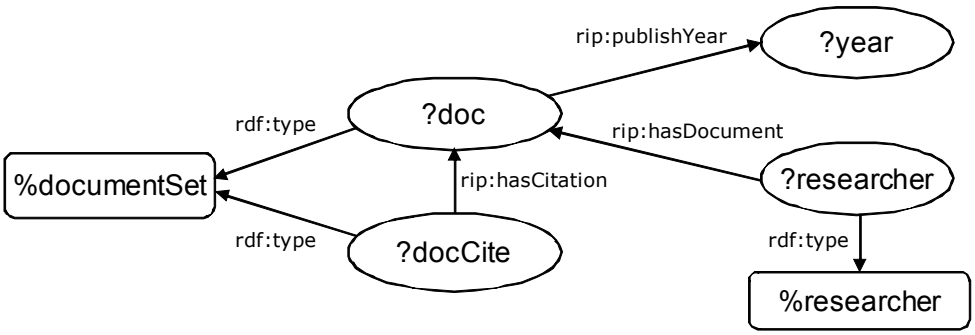


Fig. 5. RDF graph pattern specified in the WHERE clause in Fig. 4.

The second example is for ranking researchers in a topic specified with a keyword. That is, we would like to rank researchers by considering $N_{r,k}$ number of documents written by a researcher *r* including a keyword *k*. We call the number $N_{r,k}$ the *k* index of researcher *r*. Figure 7 shows a query for this example in GRASQL. The documents `?doc` written by the researcher `?r` including the keyword `%keyword` are obtained not only by RDF graph pattern matching but also by calling an external program specified in the WHERE clause

shown in Fig. 8. The predicate `rix:hasWord` of this example is used to call a full-text search engine for finding MEDLINE abstracts including the keyword, and the results are cached in

Statistic Function `ris:hIndex`(*D*): integer
{ Input *D* is a sequence of pair of document *doc* and document *docCite* which sites document *doc*.}
begin
Count number *Np* of documents including the first element *doc* of *D* without overlap ;
val *h*:=0 ;
repeat
begin
h:=*h*+1;
Count number *L* of documents cited at least *h* documents utilising *D* ;
Count number *M* of documents cited at most *h* documents utilising *D* ;
if ($L \geq h$ and $\left(N_p - \boldsymbol{h}\right) \leq M$) then break ;
end ;
return ***h*** ;
end ;

Fig. 6. Algorithm that implements the statistical function `ris:hIndex`, which computes the *h* index with a sequence of pairs of document *doc* and document *docCite* that cites document *doc*.

```
@prefix rio: <http://omicspace.riken.jp/GRASQL/>
@prefix rip: <http://omicspace.riken.jp/GRASQL/predicate>
@prefix rix: <http://omicspace.riken.jp/GRASQL/procedure>
@prefix ris: <http://omicspace.riken.jp/GRASQL/statistics>
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
@let  %documentSet  rio:MEDLINE
@let  %researcher   rio:Researcher
@let  %keyword      "arabidopsis"
SELECT ?researcher ?k
WHERE {
    ?researcher    rdf:type             %researcher ;
                   rip:hasDocument  ?doc .
    ?doc           EXT:rix:hasWord  %kayword ;
                   rip:publishYear ?year ;
                   rdf:type             %documentSet .
    FILTER (?year >= 2005)
}
EVALUATE ?k FOR ?researcher {
    ?h = ris:hIndex([?doc,?docCite]);
}
ORDER BY DESC(?k)
```

Fig. 7. GRASQL query that ranks researchers by *k* index using MEDLINE abstracts.

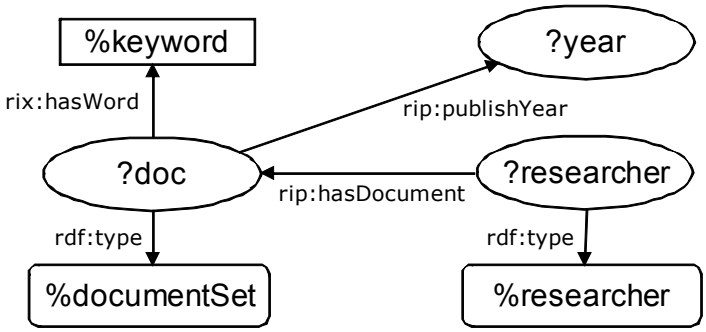


Fig. 8. RDF graph pattern specified in the WHERE clause in Fig. 7.

RDF graphs. Further, the k index, namely the number of documents `?doc` without duplication for each researcher `?r`, is computed by calling the statistical function `ris:countDistinct` in the EVALUATE clause.

```
@prefix rio: <http://omicspace.riken.jp/GRASQL/>
@prefix rip: <http://omicspace.riken.jp/GRASQL/predicate>
@prefix rix: <http://omicspace.riken.jp/GRASQL/procedure>
@prefix ris: <http://omicspace.riken.jp/GRASQL/statistics>
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
@let  %keyword  "type 2 diabetes"
@let  %documentSet  rio:MEDLINE
@let  %geneSet  rio:MouseGene
SELECT ?gene ?p
WHERE  {
    ?gene                rip:hasDocument      ?docGene ;
                         rip:hasDocument      ?docIntersection ;
                         rdf:type             %geneSet .
    ?docKey              EXT:rix:hasWord      %keyword ;
                         rdf:type             %documentSet .
    ?docIntersection     EXT:rix:hasWord      %keyword ;
                         rdf:type             %documentSet .
    ?docGene             rdf:type             %documentSet .
    ?docAll              rdf:type             %documentSet .
}
EVALUATE ?p FOR ?gene      {
     ?p = ris:statisticTest#FisherExactTest(?a,?b,?c,?d) ;
     ?a = count(DISTINCT ?docIntersection) ;
     ?b = count(DISTINCT ?docKey)-?a ;
     ?c = count(DISTINCT ?docGene)-?a ;
     ?d = count(DISTINCT ?docAll)-?a-?b-?c
}
ORDER BY ?p
```

Fig. 9. GRASQL query for a direct search using Fisher's exact test as a method of computing the statistical significance of the intersection `?docIntersection`.

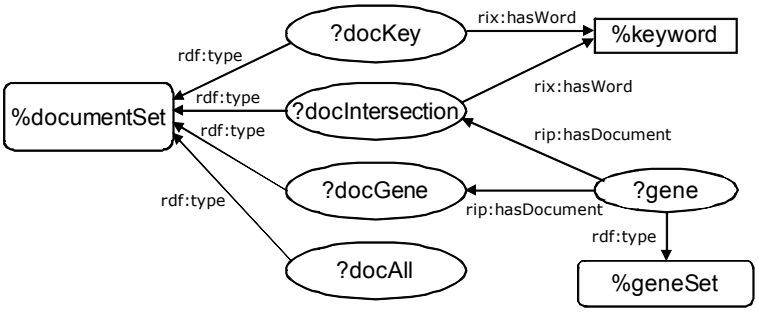


Fig. 10. RDF graph pattern satisfying the condition described in the WHERE clauses shown in Figs. 9 and 12.

3.2 Statistical tests in the PosMed search

Our method discovers entities significantly related to a user's keyword by using the documents associated with the entities. In this study, we use 'entity' (or 'document') to clearly denote that the RDF name is a biomedical entity (or a document). The simplest method for discovering entities is (1) full-text search over documents to find those containing the user's keyword, and then (2) obtaining entities associated with the documents found. This process is notated here as $user's\ keyword \rightarrow document \rightarrow entity$. To compute the significance of the association between each entity and a keyword, we have introduced a statistical test based on the number of shared documents. More concretely, for each entity, the search engine first generates a 2×2 contingency table consisting of the number of documents

a. matching both the keyword and the entity,
b. matching the keyword but not matching the entity,
c. not matching the keyword but matching the entity and
d. matching neither the keyword nor the entity.

Then, the engine applies a statistical test to the contingency table and computes a *P*-value, or the significance of the test. Finally, all resultant entities are ranked by their *P*-values. We call the discovery method described above a direct search, which is described by the query in Fig. 9. The statistical function `ris:statisticalTest#FisherExactTest` computes the *P*-value by constructing a 2×2 contingency table $\begin{pmatrix} a & b \\ c & d \end{pmatrix}$ with its four arguments a, b, c and d, and applies Fisher's exact test to the table. The simple method of evaluating the query shown in Fig. 9 is a sequential evaluation of the WHERE, EVALUATE and ORDER BY clauses in this order. In this method, the WHERE clause is evaluated to obtain all RDF graphs satisfying the condition in the WHERE clause. Figure 10 shows the RDF graph pattern with all variables and constants appearing in the WHERE clause. In practice, since the number of RDF graphs matching the pattern in Fig. 10 may be huge, this simple method of evaluation requires the implementation of an optimisation mechanism to achieve a functional language processor. Figure 11 is a chart that includes a Venn diagram of MEDLINE abstracts; it shows the relationship between each subset of MEDLINE abstracts and other RDF entities. This figure shows the primitive data structure for query searching as a set of relationships between each subset of MEDLINE abstracts and other entities such as `%keyword` and `?gene`, rather than the relationships between each MEDLINE abstract and the other entities. In our example, to compute `?p`, only four subsets—`?docAll`, `?docKey`,

`?docIntersection` and `?docGene` —of MEDLINE abstracts are necessary; they can be obtained by a specialised document search method such as the full-text search technique. Since this approach does not require a huge RDF graph space for computing the statistical significance, it opens up a new possibility for realising a practical GRASQL language processing system. Furthermore, the results of statistical analysis can be stored as a named graph using the CONSTRUCT statement instead of the SELECT statement. Figure 12 shows a CONSTRUCT query that generates RDF graphs with blank nodes as shown in Fig. 11. The generated named graph can be efficiently used as input data in SPARQL as well as in GRASQL. Statistical tests can also be used in a search to indirectly generate the associations between entities and a keyword via entity–entity relationships associated with documents. A typical example of entity–entity relationships is the co-citation frequencies of the entities in documents. The significance of the association between two entities can be computed by a statistical test of the number of documents, similar to a direct search. That is, for each entity–entity relationship, a *P*-value is computed using a 2×2 contingency table that contains the number of documents

a. matching both entities,
b. matching the first entity but not matching the second,
c. not matching the first entity but matching the second and
d. matching neither entity.

The entity–entity relationship can be obtained as a set of RDF triples using the query shown in Fig. 13.

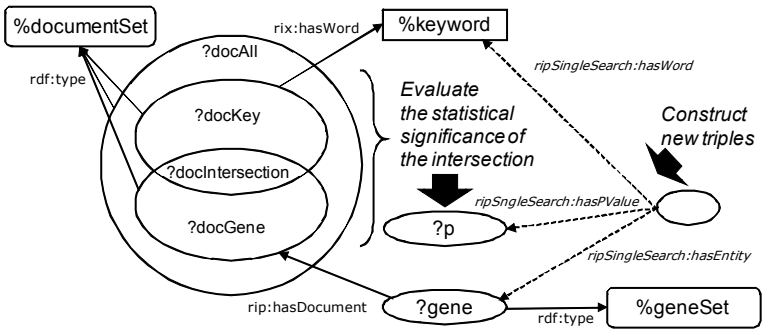


Fig. 11. Statistical diagram showing the relationships among the entities specified by the query in Fig. 12. New RDF triples are constructed by the query's CONSTRUCT statement.

We realise an inference search for the connection $user's\ keyword \rightarrow document \rightarrow entity \rightarrow document \rightarrow entity$ by applying entity–entity relationships to the entities resulting from a single association search. The *P*-value P_d of the associated entity is computed by

$$P_d = 1 - (1 - P_s)(1 - P_r) \tag{1}$$

where P_s is the *P*-value of the first direct search, and P_r is the *P*-value of the second association search of the entity–entity relationship.

Furthermore, several search connections, $user's\ keyword \rightarrow document \rightarrow entity_1 \rightarrow document \rightarrow entity_2$, which reach the same entity $entity_2$ via a different entity $entity_1$, may be obtained. In this case, the *P*-value of the resultant entity $entity_2$ can be computed by

$$P_{entity_2} = \prod_i P_{i,entity_2} \tag{2}$$

where $P_{i,entity_2}$, $(1 \le i \le n)$ are the *P*-values of *n* connections that finally reach $entity_2$. This model is based on the idea that a solution containing several connections may be more important than others. Another method is selecting the best connection by choosing the smallest *P*-value. In this case, the equation,

$$P_{entity_2} = \min_i (P_{i,entity_2}) \tag{3}$$

is applied for computing the *P*-value.

```
@prefix rio:        <http://omicspace.riken.jp/GRASQL/>
@prefix rip:        <http://omicspace.riken.jp/GRASQL/predicate>
@prefix rix:        <http://omicspace.riken.jp/GRASQL/procedure>
@prefix ris:        <http://omicspace.riken.jp/GRASQL/statistics>
@prefix rdf:        <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
@prefix ripSigleSearch:
                    <http://omicspace.riken.jp/GRASQL/singleSearch>
@let  %keyword  "type 2 diabetes"
@let  %documentSet  rio:MEDLINE
@let  %geneSet  rio:MouseGene
CONSTRUCT {
     []  ripSingleSearch:hasEntity      ?gene ;
         ripSingleSearch:hasWord  %keyword ;
         ripSingleSearch:hasPValue      ?p .
}
WHERE {
    ?gene                rip:hasDocument      ?docGene ;
                         rip:hasDocument      ?docIntersection ;
                         rdf:type             %geneSet .
    ?docKey              EXT:rix:hasWord      %keyword ;
                         rdf:type             %documentSet .
    ?docIntersection     EXT:rix:hasWord      %keyword ;
                         rdf:type             %documentSet .
    ?docGene             rdf:type             %documentSet .
    ?docAll              rdf:type             %documentSet .
}
EVALUATE ?p FOR ?gene {
    ?p = ris:statisticTest#FisherExactTest(?a,?b,?c,?d) ;
    ?a = count(DISTINCT ?docIntersection) ;
    ?b = count(DISTINCT ?docKey)-?a ;
    ?c = count(DISTINCT ?docGene)-?a ;
    ?d = count(DISTINCT ?docAll)-?a-?b-?c
}
```

Fig. 12. GRASQL query including a CONSTRUCT statement, which is used to save the results of the statistical analysis described in the WHERE and EVALUATE clauses into a set of RDF graphs. In the CONSTRUCT statement, as in SPARQL, a blank node `[ ]` is used to describe the relationships among `?gene`, `%keyword` and `?p`.

3.3 GRASQL representation of gene prioritisation in PosMed

To describe the semantics of gene prioritisation in PosMed more precisely, we will write the direct search and inference search patterns shown in Figs. 3(A) and 3(B), respectively, in GRASQL.

A GRASQL query for a direct search is written in Fig. 12. For convenience in enumerating examples, we first assume that the named graph http://omicspace.riken.jp/GRASQL/single/Mm/MEDLINE from the direct search obtained by the query in Fig. 11 is generated. We also assume that the named graph http://omicspace.riken.jp/GRASQL/relation/Mm/MEDLINE of entity–entity relationships obtained by the query in Fig. 13 is generated.

Using these two named graphs, we write a query for an inference search of the connection $user's\ keyword \rightarrow document \rightarrow entity \rightarrow document \rightarrow entity$, as shown in Fig. 14.

```
@prefix rio:       <http://omicspace.riken.jp/GRASQL/>
@prefix rip:       <http://omicspace.riken.jp/GRASQL/predicate>
@prefix ris:       <http://omicspace.riken.jp/GRASQL/statistics>
@prefix rdf:       <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
@prefix ripInference:
                   <http://omicspace.riken.jp/GRASQL/inference>
@let  %documentSet  rio:MEDLINE
@let  %geneSet  rio:MouseGene
CONSTRUCT {
     [] ripInference:hasEntity1  ?gene1 ;
        ripInference:hasEntity2  ?gene2 ;
        ripInference:hasPValue   ?p .
}
WHERE {
    ?gene1                rip:hasDocument      ?docGene1 ;
                          rip:hasDocument      ?docIntersection ;
                          rdf:type             %geneSet .
    ?gene2                rip:hasDocument      ?docGene2 ;
                          rip:hasDocument      ?docIntersection ;
                          rdf:type             %geneSet .
    ?docGene1             rdf:type             %documentSet .
    ?docGene2             rdf:type             %documentSet .
    ?docIntersection      rdf:type             %documentSet .
    ?docAll               rdf:type             %documentSet .
}
EVALUATE ?p FOR ?gene1 ?gene2 {
    ?p = ris:statisticTest#FisherExactTest(?a,?b,?c,?d) ;
    ?a = count(DISTINCT ?docIntersection) ;
    ?b = count(DISTINCT ?docKey)-?a ;
    ?c = count(DISTINCT ?docGene)-?a ;
    ?d = count(DISTINCT ?docAll)-?a-?b-?c
}
```

Fig. 13. GRASQL query that builds RDF triples of co-citation relationships of mouse genes from MEDLINE abstracts.

```
@prefix rio:          <http://omicspace.riken.jp/GRASQL/>
@prefix ris:          <http://omicspace.riken.jp/GRASQL/statistics>
@prefix  rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
@prefix  ripSingleSearch:
                      <http://omicspace.riken.jp/GRASQL/singleSearch>
@prefix  ripInference:
                      <http://omicspace.riken.jp/GRASQL/inference>
@let  %keyword        "type 2 diabetes"
@let  %geneSet        rio:MouseGene
SELECT ?gene2 ?gene1 ?p ?pTotal
FROM NAMED <http://omicspace.riken.jp/GRASQL/single/Mm/MEDLINE>
FROM NAMED <http://omicspace.riken.jp/GRASQL/relation/Mm/MEDLINE>
WHERE {
    ?x    ripInference:hasEntity2          ?gene2 ;
          ripInference:hasEntity1          ?gene1 ;
          ripInference:hasPValue           ?pInference .
    ?y    ripSingleSearch:hasEntity        ?gene1 ;
          ripSingleSearch:hasWord %keyword ;
          ripSingleSearch:hasPValue        ?pSingle .
}
EVALUATE ?p FOR ?gene1 ?gene2 {
    ?p = 1-(1-?pSingle)(1-?pInference)
}
EVALUATE ?pTotal FOR ?gene2 {
    ?pTotal = ris:multiPValue(?p)
}
ORDER BY ?pTotal ?p
```

Fig. 14. GRASQL query for inference search for connection `%keyword` → `?gene1` → `?gene2` using named graphs generated by CONSTRUCT statements in advance. In this query, the two EVALUATE clauses are evaluated sequentially in the order of their appearance. In the example, *P*-value `?p` for each pair (`?entity1, ?entity2`) is computed, and then *P*-value `?pTotal` Total for each entity `?entity2` is computed. Finally, by evaluating the ORDER BY clause, the solutions of 4-tuples (`?entity1, ?entity2, ?p, ?pTotal`) are sorted by `?pTotal` and `?p`.

The function `ris:multiPValue` in the second EVALUATE clause is an implementation of Equation 2. Furthermore, `ris:minPValue` is an implementation of Equation 3 that does not appear in this article.

4. Data preparation and implementation

4.1 Data sources

Currently, PosMed employs more than 20 million documents including MEDLINE (title, abstract and MeSH term), genome annotation, phenome information, PPI, co-expression, localisation, disease, drug and metabolite records (Table 1).

4.2 High-accuracy manual curation for generating semantic links from genes to documents

To develop a set of document databases for our original search engine for PosMed, we developed a method of mapping between genes and documents based on an NER (Leser & Hakenberg, 2005) technique that extracts named entities such as genes from a document.

A. PosMed

	No. of documents	Data sources	Data Contents	Reference
MEDLINE	18 295 132	MEDLINE	MEDLINE title, abstract and MeSH term	Coletti & Bleich, 2001
Mouse mutant	12 911	BRMM	Mouse phenotypes	Masuya et al., 2007
OMIM	21 136	OMIM	Genetic disorder descriptions	Amberger et al., 2009
HsPPI	35 731	HsPPI	Protein-protein interaction	Makino & Gojobori, 2007
REACTOME	10 761	REACTOME	Biological pathways	Matthews et al., 2009
Mouse gene record	58 768	MGI	Gene descriptions (annotations)	Blake et al., 2009
Rat gene record	36 634	RGD	Gene descriptions (annotations)	Dwinell et al., 2009
Human gene record	35 362	HGNC	Gene descriptions (annotations)	Wain et al., 2002
Metabolite record	18 045	KNApSAcK	Metabolite descriptions	Shinbo et al., 2006
Drug record	1 015	Original data	Drug descriptions	
Disease record	1 911	Original data	Disease descriptions	
RIKEN researcher record	8 603	Original data	Names of researchers appear as authors in MEDLINE	
Total	18 534 098			

B. PosMed-plus

	No. of documents	Data sources	Data Contents	Reference
MEDLINE	18 295 132	MEDLINE	MEDLINE title, abstract and MeSH term	Coletti & Bleich, 2001
At co-expression	44 082	ATTED-II	Microarray based co-expression prediction	Obayashi et al., 2009
At localisation	8 404	SUBA-2	Experimentally validated subcellular localisation	Heazlewood et al., 2007
At PPI	24 418	AtPID	Protein-protein interaction	Cui et al., 2008
	214	RAPID	RIKEN Arabidopsis Phenome Information DB	Kuromori et al., 2006
At phenotype	1 697	TAIR	Phenotype informations from TAIR	Swarbreck et al., 2008
	1 784	Literature	Manually collected original data	
Rice markers	1 712	RAP-DB	RFLP marker	Harushima et al., 1998
	15 623		SSR marker	McCouch et al., 2002
Homologus genes	1 553 922	Original data	Homologue genes between Arabidopsis and rice	Hanada et al., 2008
Arabidopsis gene record	33 003	TAIR, UniProt	Gene descriptions (annotations)	Swarbreck et al., 2008 ; UniProt Consortium, 2009
Rice gene record	29 389	RAP-DB	Gene descriptions (annotations)	Rice Annotation Project, 2008
Total	20 009 380			

Table 1. Data descriptions for (A) PosMed and (B) PosMed-plus.

Since false-positive relationships may arise from a primitive NER method that simply checks for the appearance of a name in a document, we instead employ a full-text search engine for NER, with logical queries defined as a list of names or words related to a gene concatenated with logical operators such as `AND`, `OR` and `NOT`. Specifically, as a base query we computationally collected all the synonym names for each gene from The Arabidopsis Information Resource (TAIR) and UniProt, connected these synonyms with the logical `OR` operation and added 'Arabidopsis' with the `AND` operation. Using these base queries, we performed a full-text search against a set of documents including MEDLINE title, abstract and MeSH terms (Coletti & Bleich, 2001). To reduce false-positive hits and true-negative hits, we carefully edited these queries manually through trial and error by performing a full-text search for each trial against the document set. For example, to detect all MEDLINE documents for the AT1G03880 (cruciferin B, CRB) gene while eliminating false-positive hits with the homonym 'CRB', which represents 'chloroplast RNA binding', we defined the following query: `('AT1G03880' OR 'CRU2' OR 'CRB' OR 'CRUCIFERIN 2' OR 'CRUCIFERIN B') AND ('Arabidopsis') NOT ('chloroplast RNA binding')`. This curation method is effective for updating with the latest publications. Once we curate a query, the query can be reused to extract gene–document relationships by performing a full-text search against those new document sets.

4.3 Implementation

PosMed was developed as a web-oriented tool based on a client-server model in which users access the system with conventional web browsers. However, we recommend using Microsoft Internet Explorer 8 or later or Firefox 3 or later for Windows, and Safari 4 or later or Firefox 3 or later for Macintosh. The core software component GRASE must execute a search process by very rapidly interpreting a GRASQL query program. To develop GRASE, we employed Apache Lucene, a rapid full-text search engine with a rich query language, for testing the predicate `rix:hasWord`. Since a search process can be executed for each target entity in parallel, we use nine distributed computers to realise a high-throughput search. Therefore, we distributed the data for each entity; i.e. MEDLINE abstracts and mouse gene–mouse gene relationship data associated with each distributed mouse gene and researcher are distributed among the computers to achieve a parallel search.

5. Applications of PosMed

We describe examples illustrating the power of PosMed and PosMed-plus below.

5.1 General usage of PosMed

5.1.1 Search with user-specified keywords and chromosomal intervals

A typical application of PosMed is searching with user-specified keywords and chromosomal intervals suggested by linkage analysis. As an example, we retrieved diabetes- or insulin-related genes in the chromosomal interval from 90 Mbp to 140 Mbp on chromosome 1 in the mouse genome (Fig. 15(A)). In this example, PosMed retrieved candidate genes ranked by the statistical significance between the user's keyword and each gene. Although PosMed found > 470 000 documents, it returned results in 0,865 s. Users can download all the candidate genes together with the associated gene annotations by using the 'download rank list' button in the blue box on the left (Fig. 15(D)). PosMed also supports an expert mode that allows users to select possible search paths and confirm the number of resulting genes for each search path. Clicking on a gene name listed in the gene search result

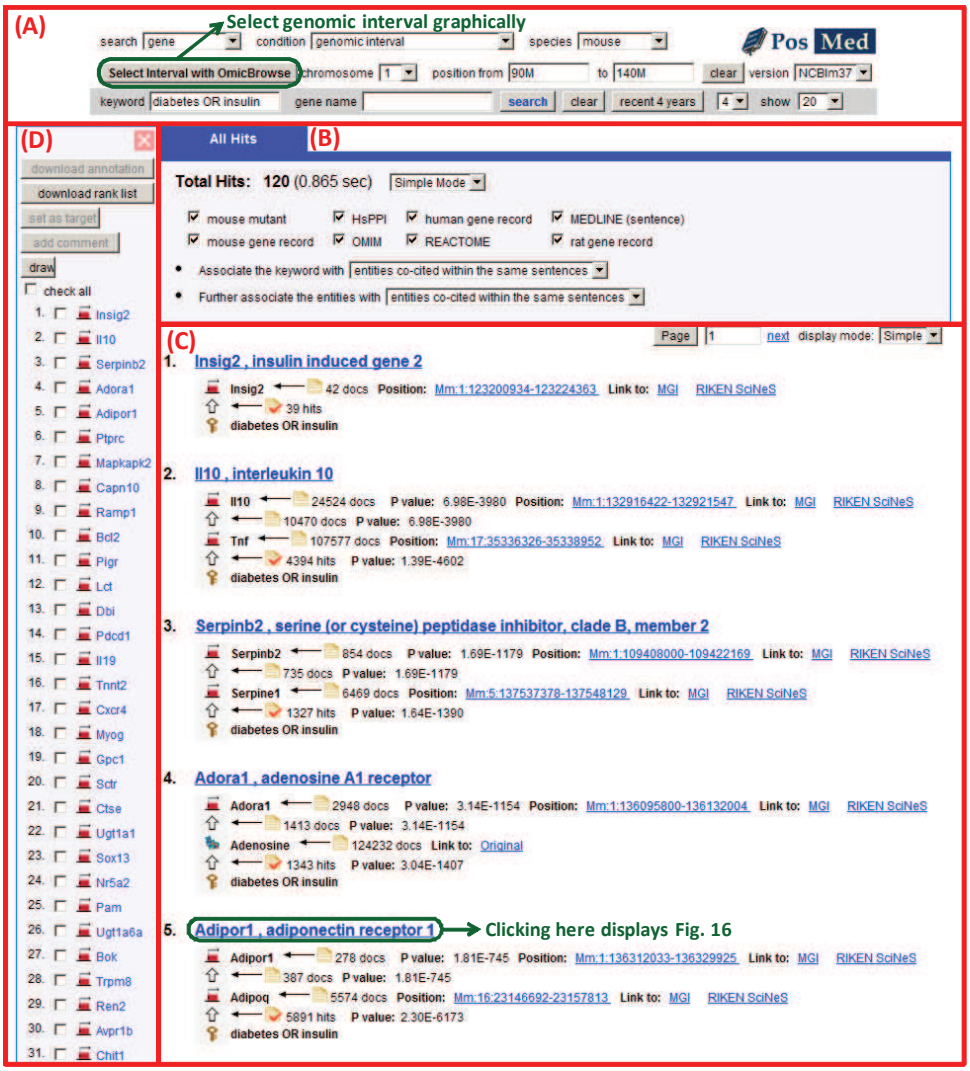


Fig. 15. Example search result for mouse genes against the query keyword 'diabetes or insulin' and the genomic interval between 90 Mbp and 140 Mbp on chromosome 1 in the NCBIm 37 genome. Users can construct queries at the top of the output display (A). To select a genomic interval visually, PosMed cooperates with the Flash-based genomic browser OmicBrowse. The 'All Hits' tab (B) shows a list of selectable document sets to be included in the search. As a default parameter, PosMed sets 'Associate the keyword with entities co-cited within the same sentences'. If the total number of candidate genes is less than 20, PosMed will automatically change this to 'Associate the keyword with entities co-cited within the same document' to show more candidates (B). Search results are ranked in (C). Users can download at most 300 candidate genes and their annotations from (D).

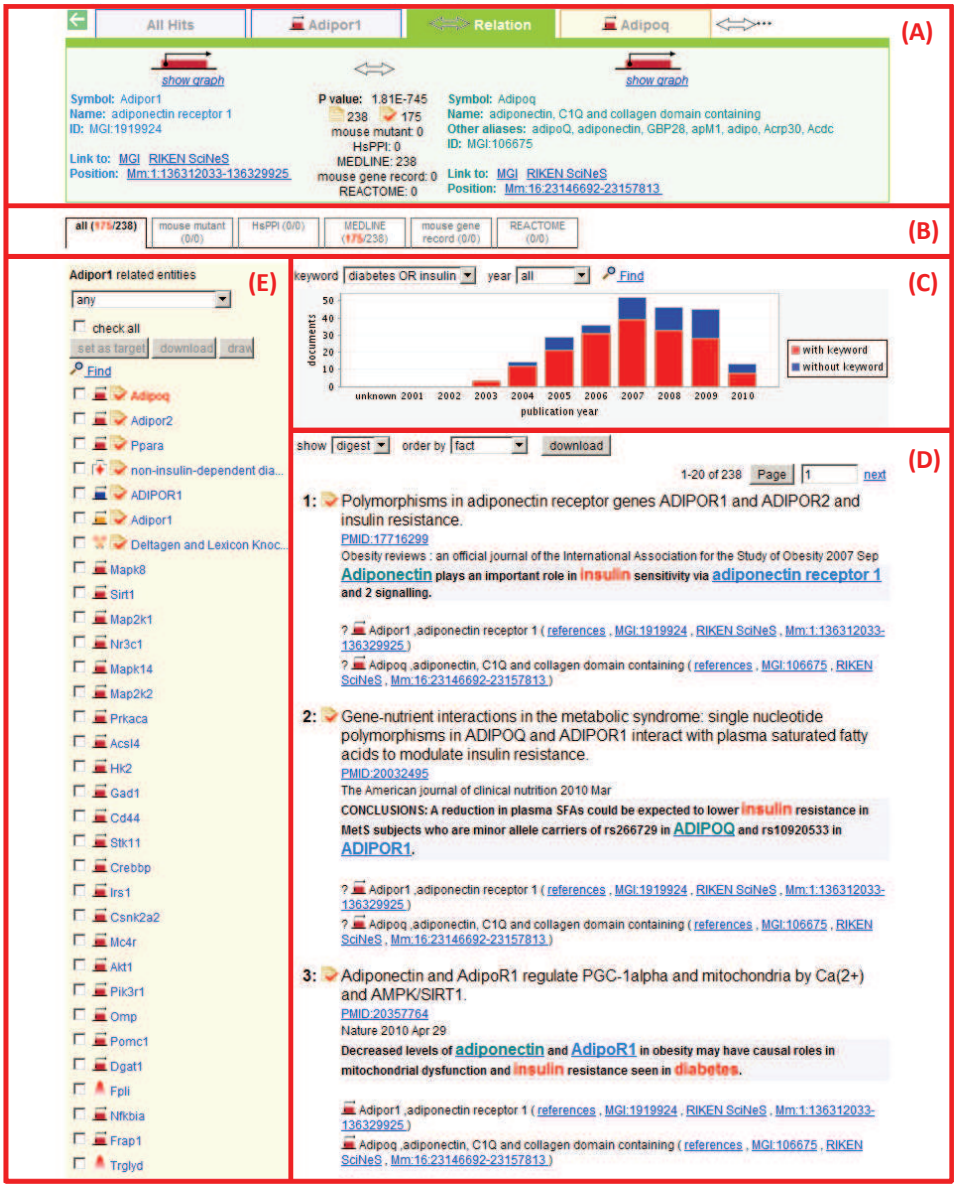


Fig. 16. Detailed document screen in PosMed. This page shows document sets supporting both the *Adipor1* gene ranked fifth in Fig. 15(C) and the *Adipoq* gene. Gene descriptions are shown in (A). Users can select the type of documents from the mouse mutant, HsPPI, MEDLINE mouse gene record or REACTOME in (B). The bar chart represents the number of related documents per year. Red and blue indicate the number of documents with and without a user-specified keyword, respectively. All documents are shown at (D). The *Adipor1*-related genes are listed in (E).

page shown in Fig. 15(B) reveals the supporting evidence for each candidate gene. To confirm the expression pattern of candidate genes with a genome browser, we provide a link to our genome browser OmicBrowse (Matsushima et al., 2009; Toyoda et al., 2007) from the gene location (Fig. 15(C)). OmicBrowse covers genome versions for mouse, human, rat, Arabidopsis and rice, and each genome is mapped to omic-type databases and a total of 344 data sources.

5.1.2 Search with phenotypic keywords

PosMed also allows users to discover genes related to phenotypic keywords. For example, if users search on the keyword 'rumpled leaves' in Arabidopsis, PosMed-plus shows four known cases via the direct search and one new candidate gene via the inference search. For the four known cases, PosMed-plus shows the link to the RIKEN Arabidopsis Phenome Information Database (RAPID), and users can confirm the phenotypes by looking at pictures. PosMed-plus also shows the evidence documents in the inference path to the AT1G51500 candidate gene. In this case, AT1G51500 is retrieved via the AT1G17840 gene, which is one of the four known genes found in the direct search. They are highly connected with co-expression, PPI and co-citation data.

5.1.3 Reference search with gene IDs

It is difficult to retrieve all the appropriate references based on gene names because of the wide variation in synonyms. Moreover, sometimes the same abbreviated names are used for functionally different genes, causing false-positive hits. In PosMed, we carefully extracted these gene–reference relationships manually, as described above. Therefore, users can retrieve the curated results with the gene ID (e.g. MGI code and AGI code) even if the abstracts do not contain the gene ID itself.

5.1.4 Search for omics data

As shown in Fig. 16, PosMed integrates various data such as gene annotations, mouse mutant records and human PPIs. Users can select any document set (the default setting is to search everything) and retrieve the required data, all within the same interface. PosMed links not only to the original databases but also to OmicBrowse, which also assists users in accessing and downloading various omics data.

5.2 *In silico* positional cloning after QTL analysis in rice

To evaluate the efficiency of PosMed with a concrete example, we confirmed whether PosMed (PosMed-plus in this rice example) could successfully retrieve correct genes that have been identified by qualitative trait locus (QTL) analysis. Three examples are described below.

Ren et al. (2005) isolated the *SKC1* gene and through QTL analysis found that it encoded an Na^+-selective transporter. In this example, we need to prioritise candidate genes without the functionally related keyword 'transporter'. Instead of the functional keyword, we retrieved genes with the phenotypic keyword 'salt tolerance' and selected the genomic interval between the markers C955 and E50811 on chromosome 1. PosMed-plus returned the Os01g0307500 (cation transporter family protein) gene with a high ranking. This is because the keyword 'salt tolerance' was mapped to the sodium ion transmembrane transporter gene AT4G10310, and Os01g0307500 was suggested as a homologue of AT4G10310.

Using a no-pollen type of male-sterile mutant (*xs1*), Zuo et al. (2008) revealed that mutant microspores are abnormally condensed and agglomerated to form a deeply stained cluster at the late microspore stage. This halts the microspore vacuolation process, and therefore,

the mutant forms lack functional pollen. This mutation is controlled by a single recessive gene, *VR1* (vacuolation retardation 1), which is located between the molecular markers RM17411 and RM5030 on chromosome 4. We searched for candidate genes with the phenotypic keyword 'sterility' in the suggested chromosome region. PosMed-plus suggested the Os04g0605500 gene (similar to calcium-transporting ATPase) as the homologue of the Arabidopsis calcium-transporting ATPase, AT3G21180. Since Schiøtt et al. (2004) found that mutation of AT3G21180 results in partial male sterility, we conclude that PosMed-plus found an appropriate candidate.

Lastly, Zhang et al. (2008) found a male sterility mutant of anther dehiscence in advance, *add(t)*, between the markers R02004 and RM300 on chromosome 2. In this search, PosMed-plus returned RNA-binding region RNP-1, Os02g0319100 and disease-resistance protein family protein Os02g0301800, with strong homology with Arabidopsis genes. PosMed-plus retrieved the Os02g0319100 gene as a homologue of Arabidopsis mei2-like (AML) protein 5, AT1G29400. As supporting evidence, Kaur et al. (2006) showed that multiple mutants of all the AML genes displayed a sterility phenotype. The other candidate gene, Os02g0301800, was derived via an inference search. First, PosMed-plus retrieved the keyword 'sterility' in a document describing the AT2G26330 gene. Next, AT2G26330 was linked to AT5G43470 as supported by three co-citations. Finally, Os02g0301800 was returned as a homologue of AT5G43470. PosMed-plus originally suggested the Os02g0301800 gene because AT2G26330 is linked to the keyword 'sterility' in a document. However, this document states that AT2G26330 causes aberrant ovule development and female-specific sterility. Since Zhang et al. (2008) focused on male sterility, we conclude that Os02g0319100 is the appropriate candidate.

5.3 Other example results

In RIKEN's large-scale mouse ENU mutagenesis project, PosMed was used to prioritise genes and has contributed to the successful identification of more than 65 responsible genes (Masuya et al., 2007). PosMed is also used by researchers worldwide and has successfully narrowed the candidate genes responsible for a specific function after QTL analysis (Kato et al., 2008; Moritani et al., 2006).

5.4 Further usage

We here introduced PosMed as a web tool for assisting in the prioritisation of candidate genes for positional cloning. Using the search engine GRASE, we also implemented inference-type full-text search functions for metabolites, drugs, mutants, diseases, researchers, document sets and databases. For cross-searching, users can select 'any' for the search items at the top right on the PosMed web page. Since this system can search various omics data, we named it OmicScan. In addition to English, GRASE accepts queries in Japanese and French. More advanced usage of PosMed is explained in the PosMed tutorial available at http://omicspace.riken.jp/tutorial/HowToUseGPS_Eng.pdf.

6. Discussion and conclusion

To use not only well-formed knowledge in RDF but also non-well-formed document data on the Semantic Web, we have introduced statistical concepts into the existing RDF query language SPARQL using a literature mining technique for searching a vast number of documents written in a natural language. The core data structure in our method is that documents are linked with each entity accurately associated by NER with human refinement, namely manual curation. The advantages of this simple structure are as follows.

- Facility of keyword selection: An arbitrary keyword appears in at least one document. Thus, a user can choose a keyword that is not necessarily related to an entity the user wants to find.
- Open-ended extensibility of documents: A new document can be added to the system if it is associated with at least one existing entity. Documents about an entity written from various viewpoints enrich knowledge so that the entity can be linked to the user's keyword.
- Open-ended extensibility of entities: A new entity can be added if at least one document associated with it exists. Therefore, entities of different categories can be introduced, which allows association search among them.
- Open-ended extensibility of semantic knowledge: Existing biomedical data in RDF format can be introduced directly into a GRASQL query.

Thanks to these advantages, PosMed can support various types of heterogeneous omics knowledge.

PosMed has been widely used to prioritise candidate genes after QTL analysis in species including mouse and Arabidopsis and to successfully identify responsible genes. Our approach is novel compared to gene prioritisation systems such as BIOTLA (Hristovski et al., 2005), Manjal (Sehgal & Srinibasan, 2005) and LitLinker (Yetisgen-Yildiz & Pratt, 2006), since PosMed is based on *P*-values computed by Fisher's exact test via tables of numbers of documents and used as correlation scores between a user's keyword and the resulting genes for ranking.

Our future work will include data extension of PosMed with not only well-formed omics knowledge in RDF but also non-well-formed document data on the Semantic Web using the statistical concepts of GRASQL.

7. References

Adie, E.; Adams, R.; Evans, K.; Porteous, D. & Pickard, B. (2006). SUSPECTS: enabling fast and effective prioritization of positional candidates. *Bioinformatics*, Vol. 22, 773-774

Aerts, S.; Lambrechts, D.; Maity, S.; Van Loo, P.; Coessens, B.; De Smet, F.; Tranchevent, L.; De Moor, B.; Marynen, P.; Hassan, B.; Carmeliet, P. & Moreau, Y. (2006). Gene prioritization through genomic data fusion. *Nat. Biotechnol.*, Vol. 24, 537-544

Amberger, J.; Bocchini, C.; Scott, A. & Hamosh, A. (2009). McKusick's Online Mendelian Inheritance in Man (OMIM). *Nucleic Acids Res.*, Vol. 37, D793-D796

Ashburner, M.; Ball, C.A.; Blake, J.A.; Botstein, D.; Butler, H.; Cherry, J.M.; Davis, A.P.; Dolinski, K.; Dwight, S.S.; Eppig, J.T.; Harris, M.A.; Hill, D.P.; Issel-Tarver, L.; Kasarskis, A.; Lewis, S.; Matese, J.C.; Richardson, J.E.; Ringwald, M.; Rubin, G.M. & Sherlock, G. (2000). Gene ontology: tool for the unification of biology. *Nat. Gene.*, Vol. 25, 25-29

Berners-Lee, T.; Hendler, J. & Lassila, O. (2001). The Semantic Web. *Sci. Am.*, Vol. 284, 34-43

Blake, J.; Bult, C.; Eppig, J.; Kadin, J. & Richardson, J. (2009). The Mouse Genome Database genotypes::phenotypes. *Nucleic Acids Res.*, Vol. 37, D712-D719

Coletti, M. & Bleich, H. (2001). Medical subject headings used to search the biomedical literature. *J. Am. Med. Inform. Assoc.*, Vol. 8, 317-323

Cui, J.; Li, P.; Li, G.; Xu, F.; Zhao, C.; Li, Y.; Yang, Z.; Wang, G.; Yu, Q. & Shi, T. (2008). AtPID: Arabidopsis thaliana protein interactome database - an integrative platform for plant systems biology. *Nucleic Acids Res.*, Vol. 36, D999-D1008

Dwinell, M.; Worthey, E.; Shimoyama, M.; Bakir-Gungor, B.; DePons, J.; Laulederkind, S.; Lowry, T.; Nigram, R.; Petri, V.; Smith, J.; Stoddard, A.; Twigger, S.; Jacob, H. & the RGD Team. (2009). The Rat Genome Database 2009: variation, ontologies and pathways. *Nucleic Acids Res.*, Vol. 37, D744-D749

The Gene Ontology Consortium. (2006). The Gene Ontology (GO) project in 2006. *Nucleic Acids Res.*, Vol. 34, D322-D326

GeneSniffer. Available From http://www.genesniffer.org

Hanada, K.; Zou, C.; Lehti-Shiu, M.; Shinozaki, K. & Shiu, S. (2008). Importance of lineage-specific expansion of plant tandem duplicates in the adaptive response to environmental stimuli. *Plant Physiol.*, Vol. 148, 993-1003

Harushima, Y.; Yano, M.; Shomura, A.; Sato, M.; Shimano, T.; Kuboki, Y.; Yamamoto, T.; Lin, S.Y.; Antonio, B.A.; Parco, A.; Kajiya, H.; Huang, N.; Yamamoto, K.; Nagamura, Y.; Kurata, N.; Khush, G.S. & Sasaki, T. (1998). A high-density rice genetic linkage map with 2275 markers using a single F2 population. *Genetics*, Vol. 148, 479-494

Heazlewood, J.; Verboom, R.; Tonti-Filippini, J.; Small, I. & Millar, A. (2007). SUBA: the Arabidopsis subcellular database. *Nucleic Acids Res.*, Vol. 35, D213-D218

Hirsch, J.E. (2005). An Index to Quantify an Individual's Scientific Research Output. *Proc. Natl. Acad . Sci . USA*, Vol. 102, 16569-16572

Hristovski, D.; Peterlin, B.; Mitchell, J.A. & Humphrey, S.M. (2005). Using literature-based discovery to identify disease candidate genes. *Int. J. Med. Inform.*, Vol. 74, 289-298

Kato, N.; Watanabe, Y.; Ohno, Y.; Inoue, T.; Kanno, Y.; Suzuki, H. & Okada, H. (2008). Mapping quantitative trait loci for proteinuria-induced renal collagen deposition. *Kidney Int.*, Vol. 73, 1017-1023

Kaur, J.; Sebastian, J. & Siddiqi, I. (2006). The Arabidopsis-mei2-like genes play a role in meiosis and vegetative growth in Arabidopsis. *Plant Cell*, Vol. 18, 545-559

Kobayashi, N. & Toyoda, T. (2008). Statistical search on the Semantic Web. *Bioinformatics*, Vol. 24, 1002-1010

Köhler, S.; Bauer, S.; Horn, D. & Robinson, P. (2008). Walking the interactome for prioritization of candidate disease genes. *Am. J. Hum. Genet.*, Vol. 82, 949-958

Kuromori, T.; Wada, T.; Kamiya, A.; Yuguchi, M.; Yokouchi, T.; Imura, Y.; Takabe, H.; Sakurai, T.; Akiyama, K.; Hirayama, T.; Okada, K. & Shinozaki, K. (2006). A trial of phenome analysis using 4,000 Ds-insertional mutants in gene-coding regions of Arabidopsis. *Plant J.*, Vol. 47, 640-651

Leser, U. & Hakenberg, J. (2005). What makes a gene name? Named entity recognition in the biomedical leterature. *Brief Bioinform.,* Vol. 6, 357-369

Makino, T. & Gojobori, T. (2007). Evolution of protein-protein interaction network. *Genome Dyn.*, Vol. 3, 13-29

Makita, Y.; Kobayashi, N.; Mochizuki, Y.; Yoshida, Y.; Asano, S.; Heida, N.; Deshpande, M.; Bhatia, R.; Matsushima, A.; Ishii, M.; Kawaguchi, S.; Iida, K.; Hanada, K.; Kuromori, T.; Seki, M.; Shinozaki, K. & Toyoda, T. (2009). PosMed-plus: an intelligent search engine that inferentially integrates cross-species information resources for molecular breeding of plants. *Plant Cell Physiology*, Vol. 50, 1249-1259

Manola, F. & Miller, E. (2004). RDF Primer. World Wide Web Consortium, Recommendation REC-rdf-primer-20040210. Available from http://www.w3.org/TR/2004/REC-rdf-primer-20040210/

Masuya, H.; Yoshikawa, S.; Heida, N.; Toyoda, T.; Wakana, S. & Shiroishi, T. (2007). Phenosite: a web database integrating the mouse phenotyping platform and the experimental procedures in mice. *J. Bioinform. Comput. Biol.*, Vol. 5, 1173-1191

Masuya, H.; Makita, Y.; Kobayashi, N.; Nishikata, K.; Yoshida, Y.; Mochizuki, Y.; Doi K.; Takatsuki, T.; Waki, K.; Tanaka, N.; Ishii, M.; Matsushima, A.; Takahashi, S.; Mizoguchi, R.; Kozaki K.; Furuichi, T.; Kawaji, H.; Wakana, S.; Nakamura, Y.; Yoshiki, A.; Murata, T.; Fukami-Kobayashi, K.; Mohan, S.; Ohara, O.; Hayashizaki, Y.; Obata, Y. & Toyoda, T. (2010). The RIKEN integrated database of mammals. *Nucleic Acids Res.*, (in press)

Matsushima, A.; Kobayashi, N.; Mochizuki, Y.; Ishii, M.; Kawaguchi, S.; Endo, T.A.; Umetsu, R., Makita, Y. & Toyoda, T. (2009). OmicBrowse: a Flash-based high-performance graphics interface for genomic resources. Nucleic Acids Res., Vol. 37, Web Server Issue, W57-W62

Matthews, L.; Gopinath, G.; Gillespie, M.; Caudy, M.; Croft, D.; de Bono, B.; Garapati, P.; Hemish, J.; Hermjakob, H.; Jassal, B.; Kanapin, A.; Lewis, S.; Mahajan, S.; May, B.; Schmidt, E.; Vastrik, I.; Wu, G.; Birney, E.; Stein, L. & D'Eustachio, P. (2009). Reactome knowledgebase of human biological pathways and processes. *Nucleic Acids Res.*, Vol. 37, D619-D622

McCouch, S.; Teytelman, L.; Xu, Y.; Lobos, K.; Clare, K.; Waltam, M.; Fu, B.; Maghirang, R.; Li, Z.; Xing, Y.; Zhang, Q.; Kono, I.; Yano, M.; Fjellstrom, R.; DeClerck, G.; Schneider, D.; Cartinhour, S.; Ware, D. & Stein, L. (2002). Development and mapping of 2240 new SSR markers for rice (*Oryza sativa* L.). *DNA Res.*, Vol. 9, 199-207

Moritani, M.; Togawa, K.; Yaguchi, H.; Fujita, Y.; Yamaguchi, Y.; Inoue, H.; Kamatani, N. & Itakura, M. (2006). Identification of diabetes susceptibility loci in db mice by combined quantitative trait loci analysis and haplotype mapping. *Genomics*, Vol. 88, 719-730

Noruzi, A. (2005). Google Scholar: The New Generation of Citation Indexes. *Libli*, Vol. 55, 170-180

Obayashi, T.; Hayashi, S.; Saeki, M.; Ohta, H. & Kinoshita, K. (2009). ATTED-II provides coexpressed gene networks for Arabidopsis. *Nucleic Acids Res.*, Vol. 37, D987-D991

Prud'Hommeaux, E. & Seaborne, A. (2008). SPARQL Query Language for RDF. World Wide Web Consortium, Recommendation REC-rdf-sparqlquery-20080115. Available from http://www.w3.org/TR/2008/REC-rdf-sparql-query-20080115/

Ren, Z.; Gao, J.; Li, L.; Cai, X.; Huang, W.; Chao, D.; Zhu, M.; Wang, Z.; Luan, S. & Lin, H. (2005). A rice quantitative trait locus for salt tolerance encodes a sodium transporter. *Nat. Genet.*, Vol. 37, 1141-1146

Rice Annotation Project (2008). The Rice Annotation Project Database (RAPDB): 2008 update. *Nucleic Acids Res.*, Vol. 36, D1028-D1033

Schiött, M.; Romanowsky, S.; Baekgaard, L.; Jakobsen, M.; Palmgren, M. & Harper, J. (2004). A plant plasma membrane Ca^{2+} pump is required for normal pollen tube growth and fertilization. *Proc. Natl. Acad . Sci . USA*, Vol. 101, 9502-9507

Seelow, D.; Schwarz, J. & Schuelke, M. (2008). GeneDistiller - distilling candidate genes from linkage intervals. *PLoS ONE*, Vol. 3, e3874, 537-544

Sehgal, A.K. & Srinivasan, P. (2005). Manjal - A Text Mining System for MEDLINE. *Proceedings of the 28th Annual International ACM SIGIR.*, Salvador, Brazil, 680

Shinbo, Y.; Nakamura, Y.; Altaf-Ul-Amin, Md.; Asahi, H.; Kurokawa, K.; Arita, M.; Saito, K.; Ohta, D.; Shibata, D. & Kanaya, S. (2006). KNApSAcK: A Comprehensive Species-Metabolite Relationship Database, In: *Plant Metabolomics*, Biotechnology in Agriculture and Forestry, Vol. 57, Saito, K.; Dixon, R.A. & Willmitzer, L., 165-184, Springer Verlag, Berlin

Swarbreck, D.; Wilks, C.; Lamesch, P.; Berardini, T.Z.; Garcia-Hernandez, M.; Foerster, H.; Li, D.; Meyer, T.; Muller, R.; Ploetz, L.; Radenbaugh, A.; Singh, S.; Swing, V.; Tissier, C.; Zhang, P. & Huala, E. (2008). The Arabidopsis Information Resource (TAIR): gene structure and function annotation. *Nucleic Acids Res.*, Vol. 36, D1009-D1014

Thornblad, T.; Elliott, K.; Jowett, J. & Visscher, P. (2007). Prioritization of positional candidate genes using multiple web-based software tools. *Twin Res. Hum. Genet.*, Vol. 10, 861-870

Toyoda, T.; Mochizuki, Y.; Player, K.; Heida, N.; Kobayashi, N. & Sakaki, Y. (2007). OmicBrowse: a browser of multidimensional omics annotations. *Bioinformatics*, Vol. 23, 524-526

UniProt Consortium (2009). The Universal Protein Resource (UniProt) 2009. *Nucleic Acids Res.*, Vol. 37, D169-D174

Van Driel, M.; Cuelenaere, K.; Kemmeren, P.; Leunissen, J.; Brunner, H. and Vriend, G. (2005). GeneSeeker: extraction and integration of human disease-related information from web-based genetic databases. *Nucleic Acids Res.*, Vol. 33, W758-W761

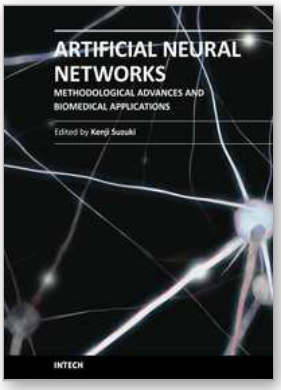
Wain, H.; Lush, M.; Ducluzeau, F. & Povey, S. (2002). Genew: the human gene nomenclature database. *Nucleic Acids Res.*, Vol. 30, 169-171

Yetisgen-Yildiz, M. & Pratt, W. (2006). Using statistical and knowledge-based approaches for literature-based discovery. *J. Biomed. Inform.*, Vol. 39, 600-611

Yoshida, Y.; Makita, Y.; Heida, N.; Asano, S.; Matsushima, A.; Ishii, M.; Mochizuki, Y.; Masuya, H.; Wakana, S.; Kobayashi, N. & Toyoda, T. (2009). PosMed (Positional Medline): prioritizing genes with an artificial neural network comprising medical documents to accelerate positional cloning. *Nucleic Acids Res.*, Vol. 37, Web Server issue, W147-W152

Zhang, Y.; Li, Y.; Zhang, J.; Shen, F.; Huang, Y. & Wu, Z. (2008). Characterization and mapping of a new male sterility mutant of anther advanced dehiscence (t) in rice. *J. Genet. Genomics*, Vol. 35, 177-182

Zuo, L.; Li, S.; Chu, M.; Wang, S.; Deng, Q.; Ding, L.; Zhang, J.; Wen, Y.; Zheng A. & Li, P. (2008). Phenotypic characterization, genetic analysis, and molecular mapping of a new mutant gene for male sterility in rice. *Genome*, Vol. 51, 303-308



Artificial Neural Networks - Methodological Advances and Biomedical Applications

Edited by Prof. Kenji Suzuki

ISBN 978-953-307-243-2
Hard cover, 362 pages
Publisher InTech
Published online 11, April, 2011
Published in print edition April, 2011

Artificial neural networks may probably be the single most successful technology in the last two decades which has been widely used in a large variety of applications in various areas. The purpose of this book is to provide recent advances of artificial neural networks in biomedical applications. The book begins with fundamentals of artificial neural networks, which cover an introduction, design, and optimization. Advanced architectures for biomedical applications, which offer improved performance and desirable properties, follow. Parts continue with biological applications such as gene, plant biology, and stem cell, medical applications such as skin diseases, sclerosis, anesthesia, and physiotherapy, and clinical and other applications such as clinical outcome, telecare, and pre-med student failure prediction. Thus, this book will be a fundamental source of recent advances and applications of artificial neural networks in biomedical areas. The target audience includes professors and students in engineering and medical schools, researchers and engineers in biomedical industries, medical doctors, and healthcare professionals.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Norio Kobayashi and Tetsuro Toyoda (2011). Prioritising Genes with an Artificial Neural Network Comprising Medical Documents to Accelerate Positional Cloning in Biological Research, Artificial Neural Networks - Methodological Advances and Biomedical Applications, Prof. Kenji Suzuki (Ed.), ISBN: 978-953-307-243-2, InTech, Available from: http://www.intechopen.com/books/artificial-neural-networks-methodological-advances-and-biomedical-applications/prioritising-genes-with-an-artificial-neural-network-comprising-medical-documents-to-accelerate-posi

INTECH
open science | open minds

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821